

N-VOD 서버에서 VCR 서비스를 위한 효율적인 버퍼 및 디스크 스케줄링

(An Efficient Buffering and Disc Scheduling for VCR -Like N-VOD Services)

최성욱(Seong-Wook Choi)¹⁾

요약

멀티미디어 스트림은 일반적으로 용량이 크고, 서로 다른 미디어간의 동기화가 필요하며, 실시간으로 재생되어야 한다는 특징이 있다. 그러므로 N-VOD 서버에 관계된 연구는, 궁극적으로 디스크 대역폭이나 버퍼의 크기 등 서버의 주어진 자원 한계 아래에서 얼마만큼 사용자의 수를 최대화 하느냐에 주된 관심이 되고 있다. 본 논문에서는 효율적인 멀티미디어 서비스를 위하여 서버의 자원을 관리하여 효율적으로 서비스를 할 수 있는 VCR 조작이나 버퍼 및 디스크 관리정책을 제안한다. 시뮬레이션 해본결과 전통적인 방식보다 버퍼의 활용과 서비스 처리시간에서 약 18% 정도 향상된 성능을 보였는데, 이는 서비스 사용자의 수를 증가 시키는 문제와 밀접한 관련이 있다.

Abstract

Intensive studies have been made in the area of N-VOD server. Multimedia files in the N-VOD sever are characterized with the large volume of data, the requirements of synchronization and real-time playback of streams. The basic goal of the study is to find an efficient mechanism to allow maximum number of users under the limited resources such as Buffer size and disk bandwidth. we propose a efficient VCR-LIKE operations and buffer, disk scheduling policy for multimedia services with management of VOD sever resources. Simulation results show that the rate of buffer usage and service time of proposed scheme are about 18% performance improved than that of traditional methods. This implies that our method can allow much more users for given resources.

접수완료 : 2006. 5. 15.

심사완료 : 2006. 6. 15.

1)정회원 : 시립 인천전문대학

※ 본 연구는 시립 인천 전문대학의 2005년도 연구지원비에 의한 것임.

1. 서론

일괄 수용 서비스(Batched admission service)는 서버 자원의 한계를 극복하고 사용자 서비스의 수용능력을 증대하기 위하여 연구되었다. 이 방식은 미리 정의된 시간 간격 동안에 모아진 사용자의 동일한 비디오의 요구를 한번의 I/O 수행으로 처리하게 되므로 그에 해당하는 만큼 디스크의 부하를 감소시킬 수 있다. 일괄 수용 서비스 정책 중에서 대표적인 것으로는 배칭(Batching)[1,3,4] 스케줄링이 있는데, 이는 동일한 비디오를 요구하는 사용자의 서비스 개시 시간을 일정한 시간 간격(Time interval)동안 모아서 함께 처리하는 방식이다. 일괄 수용 서비스를 위한 배칭 스케줄링은 사용자의 입장에서는 서비스를 요청하고 난 후 일정 시간의 지연을 감수해야 하는 불편함이 있다. 배칭 스케줄링에서 초기 서비스 지연을 방지하기 위하여 재생을 조정의 개념을 추가한 adaptive piggybacking 정책[2,5] 연구되었으나, 서비스 재생율의 변동에 의한 QOS의 저하가 발생할 수 있으며 영상 및 음성 정보를 재 동기화하기 위한 부가적인 정책이 필요하여 메커니즘 복잡해 질 수 있다[6,7]. 패칭(Patching)[8,9,10]은 서비스지연을 해소할 수 있는 또 하나의 서비스 방식이다. 이 방식은 나중에 서비스를 요청하는 사용자는 현재 서비스 중인 스트림을 일단 자신의 버퍼에 저장하게하고 앞부분은 별도의 채널을 활용하여 패칭시켜 서비스하는 방식인데, 패칭용 통신채널과 버퍼가 추가로 필요하다. 그림1은 서버 계층, 프록시 계층, 사용자계층의 세 단계로 구성된 일반적인 VOD 시스템 구조이다. 서버계층에서의 스케줄링은 우선, 멀티캐스팅을 위하여 사용자들을 그룹화하기 위한 서비스 스케줄링과 버퍼 공유 스케줄링, 디스크 스케줄링으로 나눌 수 있다. 그 아래, 프록시 계층에서의 프록시 배치 전략은 복수개의 프록시를 사용하여 효과적으로 계층화[13]하는 방식과 네트워크 분산 형태의 협

력 프록시[12]가 있다. 그 외에 클러스터링 서버 방식[15]과 오버레이 형태의 연합 서버[14]도 제안되고 있다.

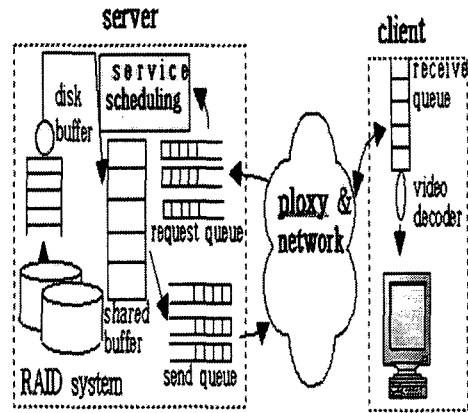


그림 34 VOD 시스템 개요
Fig. 1 VOD system overview

본 논문에서는 멀티캐스트 환경의 Near-VOD 서비스에서 VCR 동작을 효율적으로 지원하기 위한 디스크 스케줄링과 이에 따른 버퍼 관리 정책을 제안한다. 2장에서는 본 논문에서 다룬, VOD 서비스의 스케줄링에 관한 사례들을 살펴보고, 3장에서는 본 논문에서 제안한 파티션 디스크 및 버퍼 스케줄링 개념에 대하여 논의한다. 4장에서는 본 논문에서 제안한 스케줄링 정책을 시뮬레이션하고, 기존의 방식과 비교 분석한다. 끝으로 5장에서는 결론과 함께 향후 연구 방향에 대하여 간략히 기술한다.

2. 관련연구

VOD 서비스는 환경에 따라 True Video-On-Demand(T-VOD), Pay-Per-View(PPV), Quasi Video-On-Demand(Q-VOD), NearVideo-On-Demand(N-VOD), Staggered Video-On-Demand(S-VOD)등 여러 유형의 서비스가 존재하며, 그 방식의 특징은 다음과 같다. PPV ser

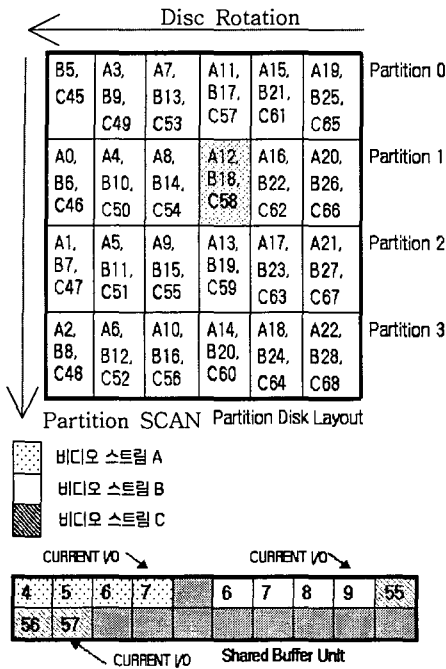
v-ice는 기존 유선방송에서의 PPV 서비스와 유사한 서비스로서, 사용자가 특정 프로그램에 대하여 계약하고 지불함으로써 제공받게 되는 영상서비스이다. Q-VOD service는 사용자들은 우선 사용 서비스에 대한 관심에 따라 사용 그룹들로 분류되고, 사용자의 VOD 서비스에 대한 대화기능은 그룹간의 상태이전으로 교환함으로써 대화형 서비스의 가장 기본적 특성만을 나타낼 수 있는 서비스이다. N-VOD service는 동일한 비디오 프로그램을 일정한 시간 간격(약 15분)으로 전송하고, 사용자는 원하는 프로그램이 전송되는 특정시간에 셋탑 장치를 통하여 시청하거나 또는 비디오 프로그램을 VCR과 같은 녹화기에 기록하였다가, down-loading 후에 재생하여 시청하는 서비스를 의미한다. 이 방식은 매우 경제적인 방식이기는 하지만, 실시간 대화형 기능을 제공하지 않으므로 VCR에서와 같은 정지, 되감기, 빨리 감기 등을 즉시 수행할 수 없다. S-VOD service는 동일한 비디오 프로그램을 다수의 전송 채널에 시간차(약5분 단위)를 두고 전송함으로써, 사용자가 채널간의 교환을 실시하여 대화형 서비스의 빨리 감기 기능과 되감기 기능을 간접적으로 실현하는 서비스를 의미한다. 때로는 이러한 서비스 방식을 넓은 의미에서 N-VOD라 부르기도 한다. T-VOD service는 사용자가 비디오 프로그램 서비스의 세션(session) 시연에 대한 완전한 제어 능력을 갖는다. 사용자는 원하는 시간에 즉시 주문에 따른 비디오 프로그램을 제어하여 취향에 맞는 영상 서비스를 제공받을 수 있다. 이러한 방식은 완전한 대화형 기능 서비스의 제공을 목적으로 한다. 그러나 특정 비디오 프로그램에 대한 주문이 집중되는 경우에 프로그램 공급 장치, 즉 비디오 서버와 전송망에서 공급과 제어에 따른 신호의 폭주현상이 발생하여 대화형 기능의 저하를 가져올 수 있다. 멀티캐스트 서비스 환경 하에서 VCR 동작을 지원하는 연구로는 우선 VCR 검색 등을 필요로 하는 사용자가 현재의 그룹에서 탈퇴하여 원하는 서비스 위치에 근접한 사용자

그룹으로 점프(JUMP) 하는 방식이 있는데, 이 방식은 패칭(Patching)서비스 환경에서 적용할 수가 있으나 채널의 확보 문제와 원하는 위치에 사용자 그룹이 존재하지 않을 경우 정확한 서비스가 어렵다[8]. 클라이언트 PC의 버퍼를 활용한 방식은 사용자에 의한 VCR 조작을 독립적으로 행할 수가 있다는 장점이 있으며, 서버에서 클라이언트까지에 이르는 서비스 라인의 자원 점유를 줄이는 효과가 있을 수가 있지만, 재생률 및 VCR 검색 시간에 따른 버퍼의 부족 문제, 검색 단위를 초과하였을 경우에 다른 멀티 캐스팅 그룹으로의 전환에 따른 자원 및 채널의 점유 문제 등이 발생할 수가 있다[9]. 또한 VCR 서비스 실행 시에 검색 위치와 가장 가까운 멀티캐스트 그룹으로 이동할 수 있는 SAM(Split and Merge Protocol) 방식도 제안되었다[10].

3. 대화형 VOD 시스템

3.1 개요

본 논문에서 제안한 스케줄링은 디스크 스케줄링을 감안한 버퍼 공유 스케줄링과 효율적인 VCR 조작이라 할 수가 있다. 버퍼 공유 스케줄링은 기존 방식(IC)에 비하여 버퍼 사용을 줄이고, 디스크 스케줄링은 기존 연구들이 VCR -LIKE 동작을 실행하기 위하여 사용되던 버퍼의 사용을 억제하고 검색시간을 단축하여 서버의 서비스 자원을 절약하기 위하여 제안되었다. 우선 그림 4의 파티션 디스크[11] 레이아웃의 특징은 단일 디스크 표면에 n 개의 원형 부분들 ($1, 2, 3, \dots, n$)인 파티션으로 나누고 각 파티션은 여러 개의 트랙들로 구성되며, 각 트랙은 고정된 블록으로 나누어진다.



[그림4] 파티션 레이아웃 및 버퍼 공유 스케줄링

[Fig.4] Partition Layout and Buffer Sharing Scheduling

각 스트림은 일련의 연속된 블록으로 표현되며, 스트림의 연속된 블록은 각각 다른 파티션에 기록되기 때문에 헤드가 특정 파티션 내의 스트림의 시작 지점을 만나면 동일 방향으로 (디스크 안쪽에서 바깥쪽으로 혹은 그 반대 방향으로) 디스크 표면을 읽어 간다. 예를 들어 비디오 A의 스트림 블록을 A0에서 A22까지를 파티션 디스크에 저장하였다면, 파티션 1에는 블록 0, 4, 8, 16, 20

파티션 2에는 1, 5, 9, 13, 17, 21 파티션 3에는 2, 6, 10, 14, 18, 22 파티션 0에는 3, 7, 11, 15, 19가 저장되어진다. 그 외 각 파티션에는 비디오 A 뿐만 아니라, 동기화를 위한 복합미디어 및 타 비디오 스트림(B, C, 등)들도 비디오 A와 동일한 방식으로 저장된다. 비디오 A의 스트림 블록 0에서 22까지를 읽기 위해서 헤드는

디스크 안쪽에서부터 바깥쪽으로 움직이며 파티션 1, 2, 3, 0 순으로 읽어간다. 만일 파티션 3에 위치한 스트림 블록 A2를 읽을 차례에서 파티션 3에 속해있는 또 다른 비디오 B, C의 스트림 블록(B16, C64)에 대한 읽기 요청이 있다면, 디스크의 회전에 의하여 짧은 시간 동안 한꺼번에 읽어 들일 수 있다. 상호 동기화가 필요한 복합 미디어를 서비스 할 경우에 실시간 디스크 스케줄링 기법간의 평균 검색 시간을 다음과 같이 나타낼 수가 있다. 우선 마감 시간 우선 주사(SCAN-EDF) 방식이나, 집단 일소(GSS) 방식은 임의 블록주사(RANDOM BLOCK SCAN) 방식과 동일한 검색 패턴을 나타내므로 식(1)로 표시할 수가 있으며, 디스크 파티션 방식은 식(2)와 같다.

$$disk_{ser} = B_n \{ S(B_l / T_r + R_x + SE) + 2SE \} \quad (1)$$

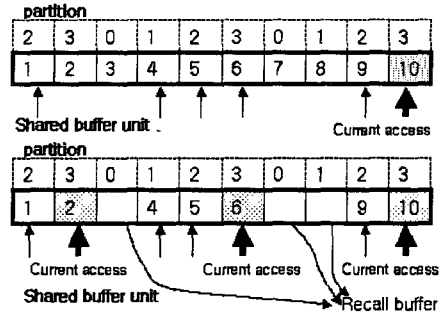
$$disk_{ser} = B_n \{ S(B_l / T_r + R_x + SE) + SE/P \} \quad (2)$$

여기서 c 는 이동 실린더(cylinder)개수 이고, SE 는 평균 디스크 조사시간, a 는 헤드 안정 계수, b 는 가속 안정 계수이다. B_n 은 블록 수, S 는 스트림 수, B_l 은 블록 크기, T_r 은 최대 전송률, R_x 은 디스크 최대 회전 속도, P 는 파티션 개수이다. 디스크 조사시간(SE)에서 마감 시간 우선 방식 및 집단 일소 방식은 단거리 탐색이나 장거리 탐색이며, 디스크 분할 기법은 동기화 스트림이 동일 트랙 상에 존재함으로써 조사시간이 거의 없거나, 최단거리 탐색으로 행할 수 있다. 또한 평균 조사시간은 장거리 탐색일 경우 $SE = a + bc$, 단거리 탐색일 경우 $SE = a + b\sqrt{c}$ 이며, 최단거리 탐색일 경우 약 2ms가 소요된다. 공유버퍼 유닛에는 현재 서비스 중인 비디오 A의 스트림 블록 4, 5, 6, 7과 비디오 B의 스트림 블록 6, 7, 8, 9 그리고 비디오 C의 스트림 블록 55, 56, 57이 저장되어 있으며, 현재 비디오 A의 7번, 비디오 B의 9번, 비디오 C의 57번 스트림 블록이 각각 서비스 중이다. 그리고 4개의 파티션으로 구분되어진 디스크에는 파티션 0에서 3까지 세 가지(A, B, C)의 비디오 스트림 블록들이 중

방향 순서대로 나누어 저장되어 있는 모습을 나타내고 있다.

3.2 버퍼공유(Buffer Sharing)

공유 버퍼 유닛은 비디오 스트림과 같이 연속 매체를 서비스할 때 하나의 사용자에게 서비스한 버퍼의 내용을 이후에 계속되는 또 다른 사용자에게 재사용할 수 있게 하기 위하여, 한번의 디스크 읽기에 의해서 앞서 서비스하고 있는 사용자의 비디오 스트림을 서비스 후에도 버퍼에 계속 남겨 두어 뒤따라오는 사용자가 이를 공유하여 사용할 수 있게 하는 방식이다. 그러므로 버퍼의 사용 비용을 무시한다면, 일반적으로 버퍼 유닛의 사이즈는 클수록 서비스에 유리하다. 본 논문에서 제안한 파티션 디스크를 이용한 공유버퍼 정책은 보다 적은 수의 버퍼를 사용하여 비디오 스트림을 서비스할 수 있다. 예를 들어 그림 4의 디스크 파티션에서 보는 바와 같이, 비디오 C에서 스트림 C45, C49, C53, C57, C61, C65는 한 번의 스캔으로 함께 읽어드릴 수가 있기 때문에 특별한 경우가 아니라면, C45에서 C65 까지 모두 공유 버퍼에 저장 할 필요는 없다. 현재 헤드가 파티션 0에 위치해 있고, 비디오 스트림 A7, B9, C57을 읽기 중이라면, 이들은 함께 읽혀서 공유버퍼 유닛에 새로 저장되어 진다. 비디오 A의 4, 5, 6, 7과 비디오 B의 6, 7, 8, 9 비디오 C의 55, 56, 57은 선두 사용자 그룹들이 사용한 후에 버퍼에 남겨두면 뒤에 오는 사용자 그룹에서 이를 재사용함으로써 신속한 서비스가 가능하고 디스크 액세스를 감소시킬 수가 있다. 그림5에는 공유 버퍼 스케줄링을 나타내었다. 2개의 표 중에서 위에 것은 인터벌 윈도우의 크기가 10인 인터벌 케싱에서의 버퍼 공유 유닛이다. 현재 비디오 스트림 블록 10을 읽어 서비스하고 있다면, 나머지 비디오 스트림 블록 9, 6, 5, 4, 1은 다른 사용자들이 공유하고 있는 상태이다.



[그림5] 공유버퍼스케줄링

[Fig.5] shared_buffer_unit scheduling

그 아래의 표는 버퍼 공유 스케줄링이다. 비디오 스트림 블록 2, 6, 10은 순서적으로는 떨어져 있지만, 모두 동일한 파티션3에 속해 있으므로 한꺼번에 읽어 올수가 있기 때문에 총 7개의 버퍼가 사용되고 나머지 3개는 회수할 수가 있다. 만약 i 번째 공유 버퍼에 포함되어 있는 스트림 블록에 대한 서비스 요청을 S_i 라 하면, 버퍼 B_i 에 의해서 절약될 수 있는 I/O 량인 $E[Ni]$ 은 (3)의 식으로 표현 될 수 있다.

$$E[Ni] = \sum S_i - 1 \quad (3)$$

3.3 일시 정지(Pause)

일시정지 기능은 일차적으로 클라이언트 버퍼를 활용하여 선입선출(FIFO)방법을 사용하여 서비스한다. 사용자가 일시정지 버튼을 클릭하면 재생은 일시 정지되지만, 비디오 스트림은 클라이언트 버퍼의 허용치만큼 계속 수신된다. 일시 정지 시간에 버퍼링되는 스트림의 크기가 클라이언트 버퍼의 크기보다 적을 경우에는 클라이언트로부터의 재생 개시 요청이 오면 버퍼링된 비디오 스트림을 즉시 재생함으로써 계속하여 서비스를 진행할 수가 있다. 그러나 일시 정지 시간이 길어 클라이언트의 버퍼가 부족할 경우는 일단 스트림의 전송을 중지하고, 서비스 재개시 요청이 오면, 해당 스트림이 저장된 파티션의 서비스가 시작될 경우 다

른 스트림과 함께 서비스할 수가 있다. 예를 들어 그림 4에서 파티션0의 B5를 서비스 중에 일시 정지하였고, 서비스 재 개시 때에는 파티션2의 B23을 서비스 해주고 있다고 하면, 이후 B24를 서비스하고 B25를 서비스 해주기 위하여 검색 위치가 파티션0로 오게 되면 이때 B5와 B24를 함께 읽어 서비스 해줄 수가 있다. 서비스 재 개시 요청 후 최대 대기시간을 P_{Wait_Time} 라하고 파티션의 평균 검색시간을 Psv , 현재 파티션의 수를 Pn 이라하면 식(4)와 같이 표현 할 수가 있다.

$$P_{Wait_Time} = Psv * (Pn - 1) \quad (4)$$

그리고 기존의 배칭(인터벌 혹은 패칭) 윈도우의 크기에 따라 클라이언트 버퍼의 사이즈가 결정되는 방식과 비교한 식이 식(5)와 식(6)에 나타내었다. 본 논문에서 제안된 스케줄링은 윈도우의 크기와 독립적이며 파티션의 개수에 비례한다고 할 수가 있다. 윈도우의 크기를 $SV_{invssec}$, $pl_{sec}i$ 를 i 번째 비디오 스트림의 재생율, i 번째 비디오 스트림의 블록의 크기를 Bl_i 라 하면 기존 방식에 의한 클라이언트 버퍼의 요구량을 식(5)에 그리고 본 논문에서 제안한 효율적인 클라이언트 버퍼의 요구량을 식(6)에 나타내었다.

$$CBs \geq Bl_i * SV_{invssec} * pl_{sec}i \quad (5)$$

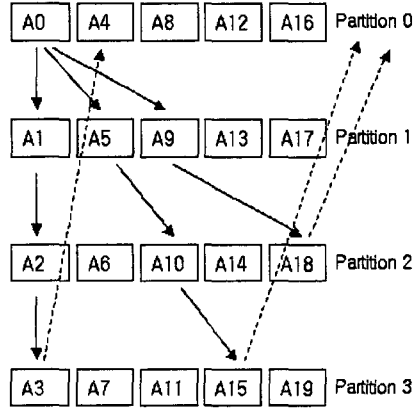
$$CBs \geq Bl_i * pn \quad (6)$$

3.4 고속 재생(Fast Forward)

파티션 디스크를 활용한 고속 재생의 경우 정속 재생의 검색 시간으로 n 배속재생이 가능하다. 그림 6에서 보면 A0를 서비스한 후 다음 A1를 검색하는 시간이나 5배속인 A5를 검색하는 시간이 같기 때문이다. 파티션 디스크에서 가능한 고속 재생 배수 FF_{sp} 는 파티션 수를 Pn , K 를 차수라 하면 식 7와 같이 정의할 수가 있다.

$$FF_{sp} = k * Pn + 1 \quad (7)$$

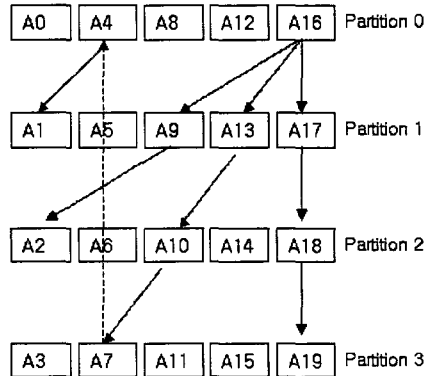
즉 파티션의 수가 4개이면 정속(A0, A1, A2, A3, A4 ...)외에 5배속(A0, A5, A10 ...)이나 9 배속(A0, A9, A18 ...)배속으로 재생 할 수가 있다.



[그림6] 고속 재생 스케줄링
[Fig.6] Fast Forward Scheduling

3.5 고속 역재생(Fast Rewind)

파티션 디스크를 활용한 고속 역 재생의 경우도 정속 재생의 검색 시간으로 n 배속의 역 재생이 가능하다.



[그림 7] 고속 역 재생 스케줄링
[Fig.7] Fast Rewind Scheduling

그림 7에서 보면 A16를 서비스한 후 다음

A17를 검색하는 시간이나 -3배속인 A13를 검색하는 시간이 같기 때문이다. 파티션 디스크에서 가능한 고속 역 재생 배수 FR_{sp} 는 파티션 수를 Pn , K 를 차수라 하면 식 8과 같이 정의할 수가 있다.

$$FR_{sp} = k * Pn - 1 \quad (8)$$

즉 파티션의 수가 4개이면 -3배속(A16, A13, A10, A7 ...) 이나 -7배속(A16, A9, A2 ...) 배속으로 재생 할 수가 있다.

4. 시뮬레이션 및 분석

본 논문에서 제안한 버퍼 및 디스크 관리 기법은 주요관심은 서버의 효율적인 자원의 활용면에서 버퍼 사용율과 VCR 조작에 따른 검색시간의 단축에 있다. 시뮬레이션 자료 및 기본 파라메타의 내용은 표1과 표2에 각각 나타내었으며, 기타 케이스별 파라메타는 시뮬레이션 결과 분석에서 언급하였다. 시뮬레이션에 사용할 자료의 비디오 요구패턴은 Zif Distribution 을 기본으로 하였고, 서비스 도착율은 λ 는 1sec당 한 개의 평균 도착율을 갖는 Poisson Distribution으로 하였다. 초당 처리 프레임의 수 F_s 는 30으로 하였다. 본 논문에서 제안한 파티션 방식의 버퍼공유 정책의 성능을 분석하기 위하여, 기존 SCAN-EDF 방식을 사용하는 인터벌 캐싱(IC)과 비교하여 보았다. 그림8은 이 두 정책간의 서비스 요구당 버퍼 사용율을 비교해본 것이다. 버퍼의 단위는 1Giga Byte로 하였다. case2는 본 논문의 버퍼 공유 캐싱 기법이며, case1은 인터벌 캐싱 정책이다. 그림8에서 초기에 case2의 버퍼 점유가 다소 증가하는 것은 초기의 대기시간 단축을 위하여 파티션 수만큼 비디오 스트림을 선반입 해오기 때문인데, 서비스 해 갈수록 파티션 디스크와 연동 스케줄링을 함으로써 case1에 비하여 버퍼 점유가 점점 떨어지는 것을 알 수가 있다. 한편 case1과 case2의 평균 버퍼 사용률은

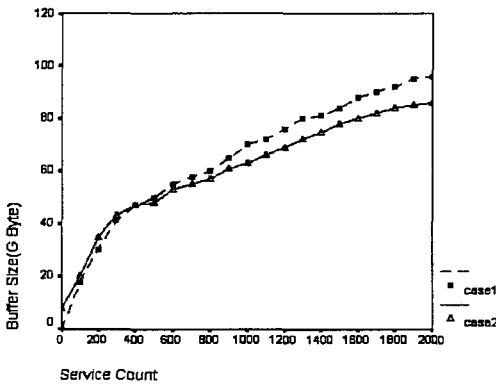
64.19 GByte와 50.33 GByte이며 표준편차는 26.13과 19.36으로 case2가 버퍼 사용률이나 변동률 면에서 효율적임을 알 수가 있다. 그림 9는 VCR 조작에 대한 클라이언트의 버퍼 사용률을 알아보기 위하여, VCR 조작이 증가함에 따른 해당 클라이언트 버퍼의 사용률을 모두 합한 값을 나타내었다. VCR 조작은 일시정지, 고속 재생 및 역 재생을 동일한 건수로 총 120건을 시뮬레이션 하였으며, 총 버퍼 사용률은 40개의 클라이언트 버퍼 사용률을 합계하여 나타낸 것이다. VCR 조작에 따른 클라이언트의 총 버퍼 사용률은 제안한 방식(case2)에서 더 효율적으로 나타나고 있다. 클라이언트 당 평균 버퍼 사용률은 case1이 1.22 GByte와 case2가 0.82 GByte으로 클라이언트별 평균 버퍼 사용률에서 유리함을 보이고 있다. 그림10과 그림11에서는 VCR 조작 요구 수에 따른 디스크 검색시간을 나타내었다. 우선 그림10에서 검색시간의 증가율을 보면 본 논문에서 제안한 Partition 방식이 SCAN 방식에 비하여 검색시간에서 유리함을 보이고 있으며, 총 VCR 조작 요구에 대한 평균 검색 시간에서도 각각 2626(msec),2227(msec)으로 Partition 방식이 효율적임을 알 수가 있다.

[표 1] 디스크 파라메타 테이블
[Table.1] Disc parameters table

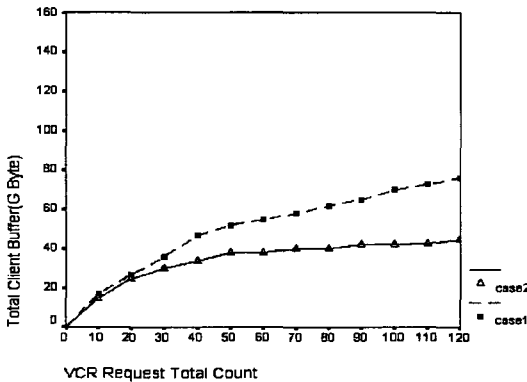
파라메타	값
Sector size	512 Byte
Cylinders	1,962
Tracks per cylinder	19
Data sector per track	72
Number of zones	1
Track skew	8 sectors
Revolution speed	4,002 RPM
Controller reads	2.2 ms
Overhead writes	2.2 ms
sort (ms)	$3.24 + 0.400 \sqrt{c}$
Seek time long (ms)	$8.00 + 0.008c$
boundary	$c = 383$
maximum transfer rate	10M BPS

[표 2] 서비스 파라메타 테이블
[Table.2] Service parameter table

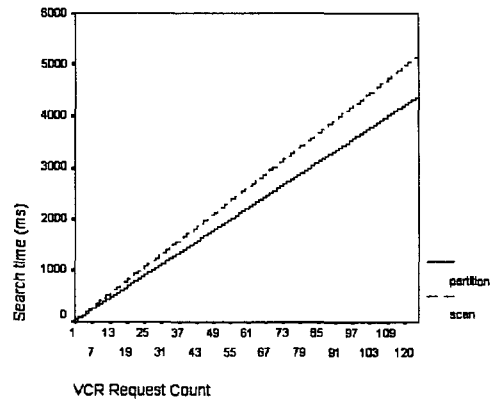
구분	내용	값
비디오 요구패턴	Zif Distribution	
λ (포아송 분포)	서비스 도착율	1/sec
V_no	비디오 수	100개
V_play_time	비디오 상영 시간	4000sec
Batch interval	배칭 간격	180sec
Service count	서비스 요구 수	2000



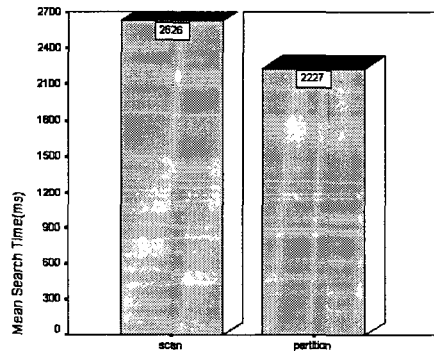
[그림 8] 버퍼 사용률(1)
[Fig. 8] Buffer Utilization(1)



[그림 9] 버퍼 사용률(2)
[Fig. 9] Buffer Utilization(2)



[그림 10] 검색 시간
[Fig. 10] Search Time



[그림 11] 평균 검색 시간
[Fig. 11] Average Search Time

5. 결론

Near-VOD에서 서버 자원의 한계를 극복하고 사용자 서비스의 수용능력을 증대시키기 위한 방식은 사용자의 일괄 수용 서비스, 인터벌 캐싱, 패칭 및 버퍼 공유 정책 등이 있다. 특히 버퍼 공유는 최근 메모리 가격의 하락에 의하여 자주 제안되고 있는 방식이다. 이 방식은 파일과 데이터베이스의 효과적인 정보 처리를 위하여 응용되었으며, 인터넷 웹상의 페이지 객체를 서비스하기 위한 방식으로 서버나 플록시 등에 활용되고 있다. 그러나 이 정책들은

대용량 연속 비디오 스트림을 서비스하기에는 무리가 있으며, 비디오 스트림을 위한 최근 몇몇의 버퍼 공유 정책도 버퍼의 한계에 효과적으로 대처하기에 어려운 점이 있다. 이를위하여 효율적인 디스크 검색 스케줄링과 연계하는 방식이 필요하다. 멀티캐스트 서비스 환경 하에서 VCR 동작을 지원하는 연구로 우선 VCR 검색 등을 필요로 하는 사용자가 현재의 그룹에서 탈퇴하여 원하는 서비스 위치에 근접한 사용자그룹으로 점프(JUMP) 하는 방식, 클라이언트 PC의 버퍼를 활용한 방식, VCR 서비스 실행 시에 검색 위치와 가장 가까운 멀티캐스트 그룹으로 이동 할 수 있는 SAM(Split and Merge Protocol) 방식 등이 있는데 이들은 채널의 확보 문제와 원하는 위치에 사용자 그룹이 존재하지 않을 경우 정확한 서비스가 어렵거나, 재생률 및 VCR 검색 시간에 따른 버퍼의 부족 문제, 검색 단위를 초과하였을 경우에 다른 멀티 캐스팅 그룹으로의 전환에 따른 자원 및 채널의 점유 문제 등이 발생할 수가 있다. 본 논문에서 제안한 버퍼 공유 정책은 파티션 디스크를 활용하여 현재 서비스와 공유 버퍼의 상태를 파악하여 효과적으로 스트림을 공유하여 서비스해 줌으로써, 버퍼의 점유와 실제 사용도(히트율) 면에서 유리하며 VCR 조작 서비스도 서버나 클라이언트의 버퍼 사용률이나 검색 효율 면에서 약 18% 정도의 감소 효과를 보여 주었다. 네트워크의 성능이 높아지고 있는 지금, 이 방식은 비디오 서버 뿐만 아니라, 플록시의 캐싱 스케줄링에서도 활용할 수가 있을 것이다.

참고 문헌

- [1] Yee-Hsiang Chang, David Cdggins, "An Open-Systems Approach to Video on Demand". IEEE Communications Magazine, pp 68-80, 1994
- [2] Ohanian TA, "Digital nonlinear editing: new approaches to editing film and video. Focal Press, Boston, Mass, 1993
- [3] Martin Arlitt, Rich Friedrich, and Tai Jin Hewlett-packard Laboratories, 1501 page Mill Road, 1998
- [4] Leana Golubchik, John C.S. Lui, "Adaptive piggybacking: a novel technique for data sharing in video-on-demand storage servers". Multimedia Systems, pp 140-15, 1996
- [5] Sreenivas Gollapudi, Aidong Zhang, "Buffer model and management in distributed multimedia systems", Multimedia Systems, pp 206-218, 1998
- [6]A. L. N. Reddy, J. C. Wyllie, "I/O Issues in a Multimedia System," IEEE COMPUTER R, pp.69-74, March 1994.
- [7]C. Ruemmer and J. Wilkes, "An Introduction to Disk Drive Modeling," IEEE Computer, Vol.27. No. 3. pp.17-28, March 1994.
- [8] K. A. Hua, Y. Chai, S. Sheu, "Pat-ching: A Multicast Technique for True Video-on-Demand Service", *Proc. of ACM Multimedia 1998*, pp. 191~200, Bristol, England, September 1998.
- [9] S.Sheu, Kien. A. Hua, and W.Tavana-pong, "Chaining : A Generalized Batching Technique for Video-On-Demand," IN pROC. of the Int'l Conf. Multimedia Computing and System, Ottawa, Ontario, Canada, pp. 110-117, June, 1977
- [10] Wanjiun Liao and Victor. O.K.Li, "The Split and Merge Protocol for Interactive Video-On-demand", IEEE Multimedia, pp.51-62, 1977
- [11] s.u choi, s.k park, "Storing Technique of Multiple Streams on Disk with a Fixed Single Zone". ITC-C, SCC, 1996.
- [12] S.Acharya and B.Smith, Middleman: "A video caching proxy server" In Proc. of The 10th International Workshop on Network and Operating System Support for Digital Audio and Video IEEE NOSSDAV, 2000.

[13] Duc.A.Tran, Kien.A.Hua, and Simon Sheu, "A new caching architecture for efficient Video-on-Demand services on the internet", Proceedings of IEEE on Applications and the Internet, pp.172-181, January 2003

[14] J.Jannotti, D.Gifford, K.Johnson, .Kaashok, and Jr.J.O'Toole, "Overcast:-Reliable multicasting with an overlay network", in Proc. of 4st Symposium on Operating Systems and Implementation-(OSDI), pp=197-212, October 2000

[15] R. Rejaie, D. Estrin, and M. Handley, "Quality Adaptation for Congestion Controlled Video Playback over the Internet," in To appear in Proc. of ACM SIGCOMM '99, Cambridge, Sept. 1999



1983년 광운 대학교 전자계산과(이학사)

1987년 경희대학교 대학원 전자공학과(공학 석사)

2001년 아주대학교 대학원 컴퓨터공학과(공학 박사)

사)

1992년~현재 시립 인천 전문대학 컴퓨터정보과 교수

관심 분야: 멀티미디어 및 VOD 시스템
소프트웨어 디자인
이동 컴퓨팅 시스템 등