



(19) 대한민국특허청(KR)
(12) 등록특허공보(B1)

(45) 공고일자 2014년08월18일
(11) 등록번호 10-1431198
(24) 등록일자 2014년08월11일

(51) 국제특허분류(Int. Cl.)
G06F 15/16 (2006.01) G06F 13/14 (2006.01)
(21) 출원번호 10-2013-0019742
(22) 출원일자 2013년02월25일
심사청구일자 2013년02월25일
(56) 선행기술조사문헌
KR101055845 B1
KR1020090032821 A

(73) 특허권자
인하대학교 산학협력단
인천광역시 남구 인화로 100, 인하대학교 (용현동)
(72) 발명자
송민석
서울 서초구 남부순환로 2311-12, 107동 402호 (방배동, 래미안방배아트힐)
이영주
인천 남동구 백범로227번길 69, B동 402호 (만수동, 삼용아파트)
김의석
서울 강서구 공항대로 351-21, 105동 801호 (등촌동, 미주진로아파트)
(74) 대리인
특허법인엠에이피에스

전체 청구항 수 : 총 7 항

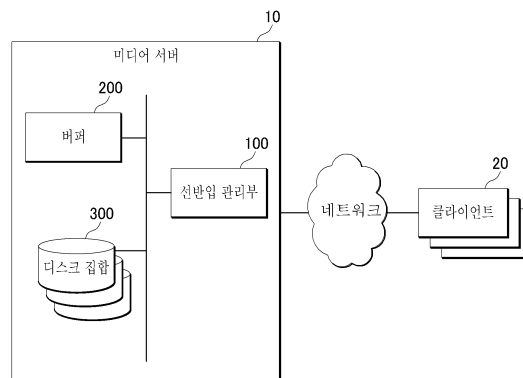
심사관 : 이석주

(54) 발명의 명칭 미디어 서버 및 미디어 데이터 선반입 방법

(57) 요약

본 발명은 하나 이상의 디스크를 포함하는 디스크 집합; 상기 디스크 집합에 저장된 미디어 데이터가 임시 저장되는 버퍼; 및 상기 미디어 데이터를 상기 버퍼에 선반입(prefetch)시키는 선반입 관리부;를 포함하되, 상기 선반입 관리부는 상기 클라이언트가 스트리밍 서비스를 받는 단위 시간인 라운드(round)의 크기를 미리 설정하고, 상기 미디어 데이터를 상기 버퍼에 선반입하는 라운드 및 선반입없이 상기 버퍼에 선반입되어 있는 데이터를 상기 클라이언트에게 제공하는 라운드의 집합인 선반입 윈도우(window)를 동적으로 설정하는 미디어 서버를 제공한다.

대표도 - 도1



이 발명을 지원한 국가연구개발사업

과제고유번호	1415122511
부처명	지식경제부
연구사업명	디지털콘텐츠원천기술개발
연구과제명	상황대응형 분산 트랜스코딩기술을 이용한 저전력 고성능 멀티미디어 콘텐츠 관리기술 개
발	
기 여 율	1/1
주관기관	인하대학교산학협력단
연구기간	2012.06.01 ~ 2013.05.31

특허청구의 범위

청구항 1

미디어 서버에 있어서,

하나 이상의 디스크를 포함하는 디스크 집합;

상기 디스크 집합에 저장된 미디어 데이터가 임시 저장되는 버퍼; 및

상기 미디어 데이터를 상기 버퍼에 선반입(prefetch)시키는 선반입 관리부;를 포함하되,

상기 선반입 관리부는

클라이언트가 스트리밍 서비스를 받는 단위 시간인 라운드(round)의 크기를 미리 설정하고,

상기 미디어 데이터를 상기 버퍼에 선반입하는 라운드 및 선반입없이 상기 버퍼에 선반입되어 있는 데이터를 상기 클라이언트에게 제공하는 라운드의 집합인 선반입 윈도우(window)를 동적으로 설정하는 미디어 서버.

청구항 2

제 1 항에 있어서,

상기 선반입 윈도우의 크기는 상기 디스크 집합이 탐색(seek) 동작에 소모하는 에너지 및 상기 버퍼 공간 접근에 대한 최대 요구가 최소가 되도록 결정되는 미디어 서버.

청구항 3

제 1 항에 있어서,

상기 선반입 윈도우의 크기는 최대 선반입 윈도우 크기의 약수(divisor) 중에서 선택되는 미디어 서버.

청구항 4

제 1 항에 있어서,

상기 선반입 윈도우의 시작 라운드보다 먼저 도착한 클라이언트의 스트리밍 요구는 상기 선반입 윈도우가 시작될 때까지 매 라운드마다 선반입 및 스트리밍이 이루어지는 미디어 서버.

청구항 5

미디어 서버에서의 미디어 데이터 선반입 방법에 있어서,

(a) 클라이언트가 스트리밍 서비스를 받는 단위 시간인 라운드(round)의 크기를 미리 설정하는 단계;

(b) 미디어 데이터를 버퍼에 선반입하는 라운드 및 선반입없이 상기 버퍼에 선반입되어 있는 데이터를 상기 클라이언트에게 제공하는 라운드의 집합인 선반입 윈도우(window)를 동적으로 설정하는 단계; 및

(c) 상기 선반입 윈도우에 따라 상기 클라이언트에게 미디어 데이터를 제공하는 단계;를 포함하는 선반입 방법.

청구항 6

제 5 항에 있어서,

상기 (b) 단계는

최대 선반입 윈도우 크기를 선택하는 단계;

최대 선반입 윈도우 크기의 약수를 구하여 후보 선반입 윈도우 크기 목록을 산출하는 단계; 및

선반입 윈도우 크기에 따른 버퍼 공간 접근에 대한 최대 요구의 변화 및 디스크 소모 에너지 변화를 고려하여 선반입 윈도우 크기를 결정하는 단계;를 포함하는 선반입 방법.

청구항 7

제 5 항에 있어서,

상기 (c) 단계는

상기 선반입 윈도우의 시작 라운드보다 먼저 도착한 클라이언트의 스트리밍 요구에 대해 상기 선반입 윈도우가 시작될 때까지 매 라운드마다 선반입 및 스트리밍하는 단계;를 더 포함하는 선반입 방법.

명세서

기술분야

[0001] 본 발명은 미디어 서버 및 미디어 데이터 선반입 방법에 관한 것이다.

배경기술

[0002] 최근의 멀티미디어(multimedia)와 네트워크 기술에 있어서의 발전은 전자 도서관, 주문형 교육(education-on-demand), 원격 교육(distance learning) 과 UCC(user-created content) 와 같은 응용들에 대해 주문형 비디오(video-on-demand) 서비스로의 즉각적인 이용을 가능하게 하였다. VoD 응용의 다양성과 인기는 데이터 센터 내의 저장 장치와 대역폭의 수요를 상당히 증가시켰다. 한 예로, 유튜브(YouTube)에서는 매일 40억 개 이상의 비디오들이 시청되고 있다. 이와 같은 수요는 데이터 센터에서의 상당한 에너지 소비는 야기하며, 그 결과 데이터 센터의 운영에 있어서의 경제적인 문제 등과 같은 큰 영향을 미치게 한다. 한 예로, 30,000ft²의 면적을 가지는 중규모의 데이터 센터는 일반적으로 15MW의 전력을 필요로 하는데 이는 연간 1300만 달러에 해당하는 금액이다. 설상가상으로 서버에서 소모되는 전력은 많은 열을 발생시키며, 이를 해결하기 위해 값비싼 냉각 시스템을 설계하게 되는데 이 또한 큰 전력을 소비한다.

[0003] 비디오 서버는 일반적으로 서버의 구성 요소 중 가장 많은 에너지를 소비하는 요소 중 하나인 디스크들의 배열로 구성되며, 디스크의 소비 전력이 데이터 센터 총 소비 전력의 27%를 차지한다고 이야기한다.

[0004] 디스크는 다양한 전력 모드(mode)를 지원하는데, 활성 모드(active mode)에서는 디스크의 헤드(head)는 데이터를 읽거나 쓰게 한다. 탐색 모드(seek mode)에서 데이터를 찾기 위해 헤드는 플래터(platter)위에서 움직이게 된다. 유휴 모드(idle mode)에서는 전속력으로 디스크가 회전하지만, 데이터 요청에 대한 처리를 하지 않는다. 대기 모드(standby mode)에서는 디스크의 회전이 완전히 멈추게 되며, 다른 모드들에 비해 가장 적은 에너지를 사용한다.

[0005] 이에 따라 서버에서의 디스크 전력 관리를 위한 기존의 연구들 대부분은 디스크가 대기 모드, 즉, 저전력 모드에 있는 기간을 확장하는 것이었다. 그러나 비디오 서버의 디스크는 끊임없이 데이터를 요청 받으므로, 대기 모드 시간을 늘리기 쉽지 않다.

[0006] 한편, 탐색 모드에서는 디스크 헤드의 기계적 움직임 때문에 각 전력 모드 중 더 많은 전력을 요구한다. 30기의 디스크를 측정 한 결과 탐색 모드에서의 평균 전력은 10.8W 인 반면에, 활성 모드와 유휴 모드에서는 각각 9.3W, 7.1W 의 평균 전력 소비를 보였으며 탐색 연산은 가장 많은 시간이 소요되었다. 따라서 전체 전력에서 탐색 모드가 주요한 기여임을 알 수 있다.

[0007] 따라서 대기 모드가 아닐 때의 디스크 전력 소모를 최대한 줄일 수 있도록 디스크의 탐색 동작 횟수를 최대한 줄이는 것이 필요하다.

[0008] 이때, 미디어 서버의 디스크 배열은 서로 다른 종류의 디스크들로 구성될 수 있으므로, 디스크 소모 에너지 관리에 있어서도 이를 고려해야 한다. 예를 들어, 서버에서 디스크 오류가 발생하여 디스크 배열의 일부 디스크를 새로운 디스크로 교체해야 할 경우, 기존의 다른 디스크보다 저장 용량이 좀더 크고 소모 전력이 좀더 작은 디스크로 교체될 가능성이 높을 것이다. 따라서 디스크의 탐색 모드를 줄이는 방법은 디스크 집합을 구성하는 각 디스크의 전력 특성이 서로 다른 경우에도 적용할 수 있어야 한다.

[0009] 디스크 소비 에너지 감소 문제와 관련하여 한국등록특허 제10-1142413호("동영상 재생 시 디스크 소모 에너지 감소를 위한 데이터 선반입 방법")에는 디스크를 저전력 모드로 전환하여 디스크 소비 에너지를 줄이는 구성이 개시되어 있다.

[0010] 또한, 한국공개특허 제2010-0127322호("동영상 재생에서 디스크 에너지 소모 감소를 위한 적응형 데이터 선 반

입 기법")도 디스크를 저전력 모드로 전환하여 디스크 소비 에너지를 줄이는 구성을 개시하고 있다.

발명의 내용

해결하려는 과제

- [0011] 본 발명은 기술한 문제를 해결하기 위한 것으로서, 그 목적은 미디어 서버의 디스크 소비 전력을 최소화할 수 있는 미디어 데이터 선반입 방법을 제공하는 것이다.

과제의 해결 수단

- [0012] 상기와 같은 목적을 달성하기 위한 본 발명의 제 1 측면에 따른 미디어 서버는 하나 이상의 디스크를 포함하는 디스크 집합; 상기 디스크 집합에 저장된 미디어 데이터가 임시 저장되는 버퍼; 및 상기 미디어 데이터를 상기 버퍼에 선반입(prefetch)시키는 선반입 관리부;를 포함하되, 상기 선반입 관리부는 클라이언트가 스트리밍 서비스를 받는 단위 시간인 라운드(round)의 크기를 미리 설정하고, 상기 미디어 데이터를 상기 버퍼에 선반입하는 라운드 및 선반입없이 상기 버퍼에 선반입되어 있는 데이터를 상기 클라이언트에게 제공하는 라운드의 집합인 선반입 윈도우(window)를 동적으로 설정하는 것을 특징으로 한다.
- [0013] 상기와 같은 목적을 달성하기 위한 본 발명의 제 2 측면에 따른 미디어 서버에서의 미디어 데이터 선반입 방법은 (a) 클라이언트가 스트리밍 서비스를 받는 단위 시간인 라운드(round)의 크기를 기설정하는 단계; (b) 미디어 데이터를 버퍼에 선반입하는 라운드 및 선반입없이 상기 버퍼에 선반입되어 있는 데이터를 상기 클라이언트에게 제공하는 라운드의 집합인 선반입 윈도우(window)를 동적으로 설정하는 단계; 및 (c) 상기 선반입 윈도우에 따라 상기 클라이언트에게 미디어 데이터를 제공하는 단계;를 포함하는 것을 특징으로 한다.

발명의 효과

- [0014] 본 발명은 미디어 서버의 디스크 소비 전력을 최소화할 수 있는 미디어 데이터 선반입 방법을 제공할 수 있다.
- [0015] 특히 본 발명은 디스크의 탐색(seek) 동작에 소모되는 에너지를 줄일 수 있다.
- [0016] 또한 본 발명은 디스크 집합이 서로 다른 종류의 디스크들로 구성된 미디어 서버 환경에도 적용할 수 있다. 예를 들어 각 디스크의 전력 특성이 동일하지 않아도 된다.

도면의 간단한 설명

- [0017] 도 1은 본 발명의 일실시예에 따른 미디어 서버의 구조를 도시하고 있다.
- 도 2는 본 발명의 일실시예에 따른 선반입 윈도우를 도시하고 있다.
- 도 3은 본 발명에 쓰인 기호의 의미를 도시하고 있다.
- 도 4는 본 발명의 일실시예에 따른 미디어 데이터 선반입 방법을 위한 수식을 도시하고 있다.
- 도 5은 본 발명의 일실시예에 따른 미디어 데이터 선반입 방법을 위한 알고리즘을 도시하고 있다.
- 도 6은 본 발명의 일실시예에 따른 미디어 데이터 선반입 방법의 흐름을 도시하고 있다.

발명을 실시하기 위한 구체적인 내용

- [0018] 아래에서는 첨부한 도면을 참조하여 본 발명이 속하는 기술 분야에서 통상의 지식을 가진 자가 용이하게 실시할 수 있도록 본 발명의 실시예를 상세히 설명한다. 그러나 본 발명은 여러 가지 상이한 형태로 구현될 수 있으며 여기에서 설명하는 실시예에 한정되지 않는다. 그리고 도면에서 본 발명을 명확하게 설명하기 위해서 설명과 관계없는 부분은 생략하였으며, 명세서 전체를 통하여 유사한 부분에 대해서는 유사한 도면 부호를 붙였다.
- [0019] 명세서 전체에서, 어떤 부분이 다른 부분과 "연결"되어 있다고 할 때, 이는 "직접적으로 연결"되어 있는 경우뿐 아니라, 그 중간에 다른 소자를 사이에 두고 "전기적으로 연결"되어 있는 경우도 포함한다. 또한 어떤 부분이 어떤 구성요소를 "포함"한다고 할 때, 이는 특별히 반대되는 기재가 없는 한 다른 구성요소를 제외하는 것이 아니라 다른 구성요소를 더 포함할 수 있는 것을 의미한다.
- [0020] 도 1은 본 발명의 일실시예에 따른 미디어 서버의 구조를 도시하고 있다.

- [0021] 미디어 서버(10)는 네트워크로 연결된 하나 이상의 클라이언트(20)에게 비디오 스트리밍(streaming) 서비스를 제공한다.
- [0022] 이를 위해서 미디어 서버(10)는 하나 이상의 디스크로 구성된 디스크 집합(300) 및 디스크 집합(300)에 저장된 미디어 데이터가 클라이언트에게 스트리밍될 때 임시 저장되는 버퍼(200)를 포함한다. 디스크 집합(300)에는 미디어 데이터가 저장되며, 디스크 집합(300)을 구성하는 디스크는 다른 종류의 디스크일 수 있다. 예를 들어, 각 디스크는 서로 다른 전력 특성을 가지고 있을 수 있다.
- [0023] 하나의 파일에 속한 데이터는 디스크 집합(300)의 도처에 산재해 저장되거나 인접하게 저장될 수 있을 것이다. 그러나 산재된 데이터 블록을 읽는 것은 많은 탐색을 요구한다. 따라서 본 발명의 일실시예에 따른 미디어 서버(10)는 한 번의 탐색으로 각각의 파일에 접근할 수 있도록, 각 미디어 파일을 디스크에 저장할 때 되도록 인접한 위치에 저장되도록 한다.
- [0024] 본 발명의 일실시예에 따른 미디어 서버(10)는 디스크 집합(300)에 저장되어 있는 미디어 데이터를 클라이언트에게 원활하게 스트리밍할 수 있도록 버퍼(200)에 선반입(prefetch)하는 본 발명의 일실시예에 따른 선반입 관리부(100)를 포함한다.
- [0025] 이때, 본 발명의 일실시예에 따른 선반입 관리부(100)는 디스크 집합(300)이 소모하는 에너지를 최대한 줄일 수 있는 선반입 시점을 결정한다. 특히 본 발명의 일실시예에 따른 선반입 관리부(100)는 디스크 집합(300)의 탐색(seek) 동작의 횟수를 줄이는 것을 목표로 한다.
- [0026] 디스크의 탐색 동작의 비용을 줄이기 위해, 본 발명의 일실시예에 따른 선반입 관리부(100)는 먼저, 디스크에서의 검색 횟수와 전력 소모간의 관계를 검사하고, 선반입 연산이 디스크 전력 요구에 어떠한 영향을 미치는지 분석한다. 그리고 버퍼(200)의 크기를 늘리지 않으면서도 전반적인 에너지 소비를 최소화하기 위해, 선반입되는 데이터의 양과 그 빈도를 결정한다.
- [0027] 즉, 본 발명의 일실시예에 따른 선반입 관리부(100)는 탐색 횟수를 줄이기 위해 비디오 스트림들의 특정한 선택에 의해 즉시 요청되는 데이터를 한 번의 탐색 연산으로 버퍼로 선반입될 수 있게 한다.
- [0028] 미디어 스트리밍은 주기적인 특성을 가지고 있으므로, 본 발명의 일실시예에 따른 선반입 관리부(100)는 라운드(round) 기반의 스케줄링을 이용한다. 라운드는 단위 시간이다. 즉, 본 발명에서는 시간을 일정한 크기의 간격으로 나눈 것을 라운드라고 지칭하고 R로 표기하기로 한다.
- [0029] 클라이언트들은 각 라운드에 한번 서비스를 받는다. 예를 들어, 라운드의 길이가 2초일 때 1.5Mbps로 스트림을 서비스하기 위해서 본 발명의 일실시예에 따른 미디어 서버(10)는 매 라운드마다 3Mbit의 데이터를 버퍼(200)로 반입해야 한다. 즉, 라운드는 각 클라이언트가 스트리밍 서비스를 받는 단위 시간이라고 할 수 있다.
- [0030] 디스크의 데이터 전송률이 비디오 스트림의 재생률보다 상당히 높기 때문에, 비디오 서버의 각 디스크들은 다양한 스트림으로 데이터를 제공할 수 있다. 디스크의 대역폭 이용은 한 라운드 동안 모든 스트림들을 탐색하는 전체 서비스 시간 비용으로 정의될 수 있다.
- [0031] 따라서, 어떤 라운드에 버퍼(200)에 데이터를 다음 라운드들에서도 사용할 수 있을 만큼 최대한 많이 선반입하고, 다음 라운드에서는 데이터를 반입 없이 이미 버퍼(200)에 반입되어 있는 데이터를 사용하여 스트리밍하는 것이 가능하다. 이렇게 하면 디스크 집합(300)에 저장되어 있는 미디어 파일에 대한 접근 횟수가 줄어들므로, 디스크 집합(300)의 탐색 횟수를 줄일 수 있을 것이다.
- [0032] 즉, 본 발명의 일실시예에 따른 선반입 관리부(100)는 디스크 탐색 횟수를 줄이기 위해 선반입 윈도우(window)라는 개념을 사용한다. 선반입 윈도우는 미디어 데이터를 한번의 선반입으로 스트리밍할 수 있는 라운드들의 수이다. 선반입 윈도우를 구성하는 하나의 라운드 동안 버퍼(200)를 되도록 많이 채워놓고, 선반입 윈도우의 나머지 라운드들에서는 버퍼(200)를 비우기만 하는 것이다. 편의상 이후 기술에서 버퍼(200)를 채워놓는 라운드를 선반입 라운드, 버퍼(200)를 비우기만 하는 라운드를 나머지 라운드라고 하기로 한다. 또한 선반입 윈도우가 포함하는 라운드의 수를 선반입 윈도우의 크기라고 지칭하기로 한다.
- [0033] 예를 들어, 선반입 윈도우 크기가 4 라면 데이터는 1개의 라운드 동안 4라운드 동안 스트리밍할 데이터를 모두 선반입하고, 나머지 3개의 라운드 동안은 버퍼(200)에 저장되어 있는 데이터를 스트리밍하기만 한다. 즉, 버퍼(200)가 다음 3라운드 동안 필요한 데이터를 이미 포함하고 있게 된다.
- [0034] 반면, 윈도우 크기가 1이라면 어떠한 데이터도 선반입되지 않음을 말한다. 이 경우에는 매 라운드마다 데이터가

버퍼(200)로 반입된다.

- [0035] 이러한 선반입 윈도우의 크기 및 선반입 라운드 시점은 디스크 소모 에너지를 최대한 줄일 수 있으면서도 최대 버퍼 요구가 최소가 될 수 있도록 동적으로 결정된다.
- [0036] 자세한 내용을 이하 도면들을 참조하여 살펴본다..
- [0037] 도 2은 본 발명의 일실시예에 따른 선반입 윈도우를 도시하고 있다.
- [0038] 도면은 시간을 단위 시간인 라운드들로 나누어 도시하고 있다. 도면의 동그라미는 선반입 동작이 이루어지고 있음을 의미한다. 선반입은 디스크 집합(300)에 접근하여 디스크에 저장되어 있는 미디어 데이터를 버퍼(200)로 반입하는 동작이므로, 이러한 선반입 동작이 이루어지는 라운드는 디스크의 탐색 동작을 필요로 한다.
- [0039] 라운드 R은 선반입이 시작되는 라운드이다. 도시되어 있는 실시예에서처럼 선반입 윈도우의 크기가 4일 때, 다음 3개의 라운드, 즉, R1, R2, R3에서는 선반입이 이루어지지 않고, 스트리밍 동작만 이루어진다. 따라서 다음 선반입 라운드는 R4가 된다.
- [0040] 도시되어 있는 바와 같이, 클라이언트 i가 R-5 라운드에 도착하였다면, 즉, R-5 라운드에서 클라이언트 i가 본 발명의 일실시예에 따른 미디어 서버(10)에 미디어 스트리밍 서비스를 요청하였다면, 본 발명의 일실시예에 따른 선반입 관리부(100)는 선반입이 시작되는 라운드인 라운드 R까지는 선반입 윈도우를 적용하지 않는다. 따라서, R-4, R-3, R-2, R-1 라운드에서는 클라이언트 i가 요청한 미디어 데이터에 대해 선반입, 즉, 디스크 탐색과 스트리밍이 모두 이루어진다.
- [0041] 이때, 선반입 윈도우의 크기 4는 최대 선반입 윈도우 크기인 8의 약수(divisor)이다. 즉, 본 발명의 일실시예에 따른 선반입 관리부(100)는 최대 선반입 윈도우 크기에 상한을 두며, 허용되는 윈도우 크기는 최대 선반입 윈도우 크기의 약수로 되도록 제한한다. 예를 들어, 최대 선반입 윈도우 크기가 8일 때 1, 2, 4, 8은 선반입 윈도우 크기로 선택될 수 있지만, 3은 8의 약수가 아니므로 선반입 윈도우 크기로 선택될 수 없다.
- [0042] 자세한 내용을 도 3 및 도 4를 통해 살펴본다. 도 3는 본 발명에 쓰인 기호의 의미를, 도 4는 본 발명의 일실시예에 따른 미디어 데이터 선반입 방법을 위한 수식을 도시하고 있다.
- [0043] 편의상 도면에서 위 첨자 및 아래 첨자를 사용하여 도시된 기호를 이후 기술에서는 첨자 없이 또는 $_$ 를 사용하여 표기한다. 예를 들어, 최대 선반입 윈도우 크기를 N_p 라고 표기한다.
- [0044] 최대 선반입 윈도우 크기를 N_p 로 설정하면, j번째 윈도우 크기는 N_p 의 j번째 약수인 V_j 가 되며, N_v 는 약수들의 총 개수를 의미한다. 라운드 R에서 새로운 N_p 의 라운드들이 연속으로 시작하므로, 윈도우 크기가 V_j 로 설정되면, 선반입 연산은 라운드 R에서 시작하여 매 V_j 라운드마다 활성화 된다.
- [0045] 예를 들어, 도 2의 실시예에서 $N_p = 8$ 이므로, 선택 가능한 윈도우 크기의 집합은 $\{V_1=1, V_2=2, V_3=4, V_8=8\}$ 이다. 윈도우 크기가 $V_3=4$ 로 결정되었으므로, 도 2에 도시되어 있는 바와 같이, 선반입은 라운드 R에서 시작하여 매 4라운드마다 활성화된다.
- [0046] 버퍼 요구가 최고조에 달하는 것을 방지하기 위해 선반입 연산이 라운드들에 걸쳐 고르게 할당될 필요가 있으므로, 가능한 멀리 떨어지게 하기 위해 각 윈도우 크기 인덱스(index) j에 대해 가능한 선반입 라운드들의 배열을 구성해야 한다. $N_{j,e}$ 는 이 배열에서의 원소들의 개수를 의미한다.
- [0047] 예를 들어, 윈도우 크기는 4 이고 $N_p = 8$ 이라면, 가능한 선반입 라운드들의 목록의 배열은 $\{ \{R, R+4\}, \{R+1, R+5\}, \{R+2, R+6\}, \{R+3, R+7\} \}$ 이 된다. 라운드 번호에 따른 오름차순으로 정렬된 배열에서 k 번째 원소를 $I_{j,k}$ 로 표현한다.
- [0048] 버퍼의 크기는 윈도우 크기에 종속적인 선반입을 필요로 한다. 선반입 라운드가 선택되면, 매 N_p 라운드마다 동일한 패턴의 선반입 연산이 반복되게 된다. 그러므로 라운드 R 과 $(R+N_p-1)$ 사이에서 오직 버퍼 요구만을 검사할 필요가 있는데, $C_i(j,k)$ 를 클라이언트 i에 대해 윈도우 크기 인덱스 j 와 선반입 라운드 목록 인덱스 k 의 경우라고 하자. 그렇다면 $B_i(j,k)(n)$ 은 n 번째 라운드 동안의 버퍼 요구가 된다. ($n=R, \dots, R+N_p-1$)
- [0049] 선반입이 발생하는 경우 해당 라운드동안의 버퍼 요구는 최대화 되었다가 다음 선반입 연산이 일어날 때 까지 라운드별로 점차 감소하게 된다. 이를 반영하기 위해, 클라이언트 i에 대해 n번째 라운드에서 선반입이 발생하는 그 다음 라운드까지의 라운드들의 수를 $D_i(j,k)(n)$ 으로 나타낼 수 있다. $C_i(j,k)$ 에 대한 버퍼 요구인 $B_i(j,k)(n)$ 은 도 4의 수식 1과 같이 계산될 수 있다.

- [0050] 에너지는 윈도우 크기에 종속적인 선반입에 의해 절약될 수 있다. $E_{saving_i,j}$ 를 클라이언트 i 에 대해 윈도우 사이즈 인덱스 j 와 윈도우 사이즈 인덱스가 1일 때 비교한 에너지 감소라 하면, N_p 를 V_j 로 나눈 것 만큼의 탐색이 N_p 라운드 동안 필요하게 되고, $E_{saving_i,j}$ 는 수식 2와 같이 계산될 수 있다.
- [0051] 각 클라이언트 i 에 대해 전체적인 디스크 에너지 소비를 최소화하는 윈도우 크기 인덱스와 선반입 라운드 목록 인덱스로 구성된 쌍을 찾기 위해 W_i 를 선택된 윈도우 크기 인덱스로, S_i 는 선택된 선반입 라운드 목록 인덱스라 하면, n 번째 라운드 동안의 버퍼 요구는 버퍼 상한 B_{max} 를 초과하지 않으며, W_i 와 S_i 를 결정하는 선반입 결정 문제(Prefetching Determination Problem)은 수식 3과 같이 나타낼 수 있다.
- [0052] 각 클라이언트 i 와 윈도우 크기 인덱스 j 에 대한 매개변수 $Q_{i,j}$ 는 수식 4와 같이 나타낼 수 있다. 수식 4의 분자는 절약된 에너지를 분모는 윈도우 사이즈 인덱스 j 와 1 간의 최대 버퍼 요구의 차이를 나타낸다.
- [0053] 선반입 결정 문제는 다수선택 다차원 knapsack 문제의 특별한 경우를 포함한다. 이는 NP-hard 문제이므로 도 5과 같은 휴리스틱(heuristic) 알고리즘을 제안한다.
- [0054] 도 5이 도시하고 있는 본 발명의 일실시예에 따른 미디어 데이터 선반입 방법을 위한 알고리즘을 간략히 설명하면, 다음 두 단계로 되어 있다.
- [0055] 1) 초기화: 모든 W_i 와 S_i 는 1로 초기화 되며, 최소 버퍼를 산출하지만 최대 에너지 소비를 갖는다.
- [0056] 2) 선택: (1) 선택된 클라이언트에 대한 윈도우 크기는 에너지 소비를 줄이기 위해 증가한다. (2) 선반입 라운드 집합은 라운드 전반에 걸쳐 버퍼 요구가 균형 잡힐 수 있도록 선택되게 된다.
- [0057] 이를 위해 높은 $Q_{i,j}$ 은 주어진 버퍼 크기에 대해 더 많은 에너지를 절약할 수 있으므로 먼저 최대 $Q_{i,j}$ 값을 갖는 클라이언트 i 를 선택한다. 그리고 클라이언트 i 의 윈도우 크기 인덱스를 j 까지 증가시킨다. 그 다음으로 선택된 윈도우 크기 인덱스 j 에 대해 선반입 라운드 집합은 가용한 집합 $I_{j,k}$ 내에서 선택되어야 한다. ($k = 1, \dots, \dots$)
- [0058] 라운드들에 걸쳐 버퍼 부하의 균형을 잡기 위해 $I_{j,k}$ 가 선택되었을 때 각 라운드 동안 최대 버퍼 요구를 계산하고, 최소 버퍼 요구를 초래하는 $I_{j,k}$ 값을 선택한다. 이 연산은 버퍼 요구를 만족하는 한 윈도우 크기를 증가시키면서 반복되게 된다.
- [0059] 도 5의 알고리즘은 클라이언트가 요청하거나 비디오 스트림이 닫힐 때 디스크에 적재되는 것에 있어서의 변화를 수용하기 위해 실행된다. 예를 들어, 전술한 바와 같이, 클라이언트 i 가 R-5 라운드에서 비디오 스트림을 요청하면 R-1 번째 라운드까지 N_p 라운드들의 다음 시퀀스의 시작인 R번째 라운드에서 새로운 윈도우 크기와 선반입 라운드로 시작하도록 구성하기 위해 W_i 는 1로 설정된다. W_i 는 R번째 라운드에서는 알고리즘 실행에 의해 획득한 값인 4로 설정된다.
- [0060] n 번째 라운드 동안 디스크들의 각 모드에 대한 에너지 소비를 결정하기 위해 디스크 m 에 대한 seek 에너지는 수식 5로 표현될 수 있으며, I_{wi}, s_i 를 만족하는 n 라운드 동안에 클라이언트 i 에 대한 읽기 시간은 이며, 디스크 m 에 대한 회전 에너지는 수식 6으로 표현될 수 있다. 한 라운드에서 유휴 시간은 T_R 로부터 읽기 시간과 탐색 시간을 뺀으로써 구할 수 있으며, 디스크 m 에 대한 유휴 모드에서의 에너지는 수식 7과 같이 표현될 수 있다.
- [0061] 따라서 n 번째 라운드 동안의 전체 에너지 소비는 수식 8과 같이 나타낼 수 있다.
- [0062] 도 6은 본 발명의 일실시예에 따른 미디어 데이터 선반입 방법의 흐름을 도시하고 있다.
- [0063] 최대 선반입 윈도우 크기(N_p)를 선택한다 (S110). 도 2의 실시예에서 $N_p=8$ 이다.
- [0064] 최대 선반입 윈도우 크기의 약수를 구하여 후보 선반입 윈도우 크기 목록을 산출한다(S120). 도 2의 실시예에서 후보 선반입 윈도우 크기 목록은 {1, 2, 4, 8}이다.
- [0065] 선반입 윈도우 크기에 따른 선반입 발생 라운드들간 최대 버퍼 요구 변화를 산출한다(S130). 즉, 후보 선반입 윈도우 크기 목록에 대해, 최대 버퍼 요구를 선반입 윈도우 크기가 1일 때와 비교한다.
- [0066] 선반입 윈도우 크기에 따른 에너지 절약을 산출한다(S140). 즉, 후보 선반입 윈도우 크기 목록에 대해, 디스크 탐색 감소에 의해 절약되는 에너지를 선반입 윈도우 크기가 1일 때와 비교한다.
- [0067] 상기 단계들(S130, S140)에서 산출한 에너지 절약 및 최대 버퍼 요구 변화를 고려하여 선반입 윈도우 크기를 결

정한다(S150).

[0068] 본 발명의 일 실시예는 컴퓨터에 의해 실행되는 프로그램 모듈과 같은 컴퓨터에 의해 실행가능한 명령어를 포함하는 기록 매체의 형태로도 구현될 수 있다. 컴퓨터 판독 가능 매체는 컴퓨터에 의해 액세스될 수 있는 임의의 가용 매체일 수 있고, 휘발성 및 비휘발성 매체, 분리형 및 비분리형 매체를 모두 포함한다. 또한, 컴퓨터 판독 가능 매체는 컴퓨터 저장 매체 및 통신 매체를 모두 포함할 수 있다. 컴퓨터 저장 매체는 컴퓨터 판독가능 명령어, 데이터 구조, 프로그램 모듈 또는 기타 데이터와 같은 정보의 저장을 위한 임의의 방법 또는 기술로 구현된 휘발성 및 비휘발성, 분리형 및 비분리형 매체를 모두 포함한다. 통신 매체는 전형적으로 컴퓨터 판독가능 명령어, 데이터 구조, 프로그램 모듈, 또는 반송파와 같은 변조된 데이터 신호의 기타 데이터, 또는 기타 전송 매커니즘을 포함하며, 임의의 정보 전달 매체를 포함한다.

[0069] 본 발명의 방법 및 시스템은 특정 실시예와 관련하여 설명되었지만, 그것들의 구성 요소 또는 동작의 일부 또는 전부는 범용 하드웨어 아키텍처를 갖는 컴퓨터 시스템을 사용하여 구현될 수 있다. 본 발명의 일 실시예의 하나 이상의 구성 요소 또는 동작을 실시하기 위하여 사용될 수 있는 컴퓨터 시스템 아키텍처의 일례를 설명하면, 하드웨어 시스템은 프로세서, 캐쉬, 메모리 및 상술한 기능에 관련된 하나 이상의 소프트웨어 어플리케이션 및 드라이버를 포함할 수 있다.

[0070] 전술한 본 발명의 설명은 예시를 위한 것이며, 본 발명이 속하는 기술분야의 통상의 지식을 가진 자는 본 발명의 기술적 사상이나 필수적인 특징을 변경하지 않고서 다른 구체적인 형태로 쉽게 변형이 가능하다는 것을 이해할 수 있을 것이다. 그러므로 이상에서 기술한 실시예들은 모든 면에서 예시적인 것이며 한정적이 아닌 것으로 이해해야만 한다. 예를 들어, 단일형으로 설명되어 있는 각 구성 요소는 분산되어 실시될 수도 있으며, 마찬가지로 분산된 것으로 설명되어 있는 구성 요소들도 결합된 형태로 실시될 수 있다.

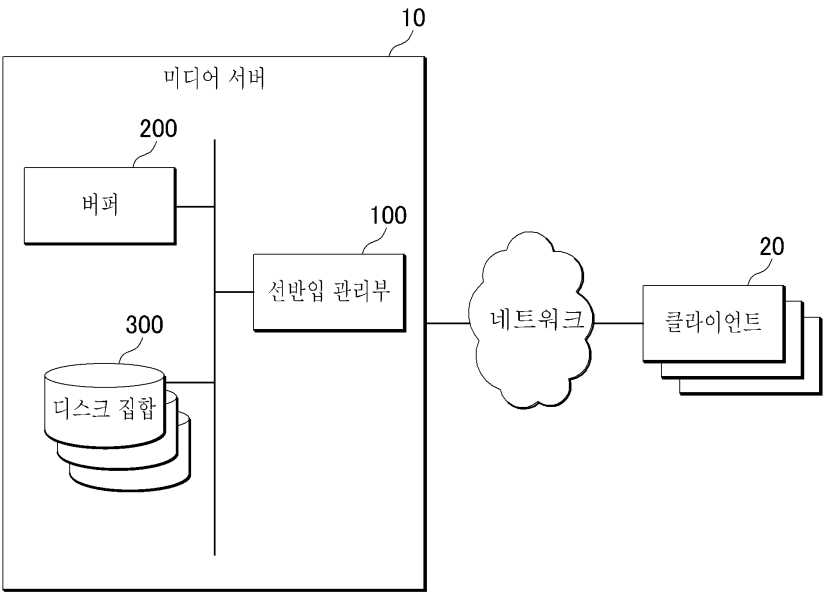
[0071] 본 발명의 범위는 상기 상세한 설명보다는 후술하는 특허청구범위에 의하여 나타내어지며, 특허청구범위의 의미 및 범위 그리고 그 균등 개념으로부터 도출되는 모든 변경 또는 변형된 형태가 본 발명의 범위에 포함되는 것으로 해석되어야 한다.

부호의 설명

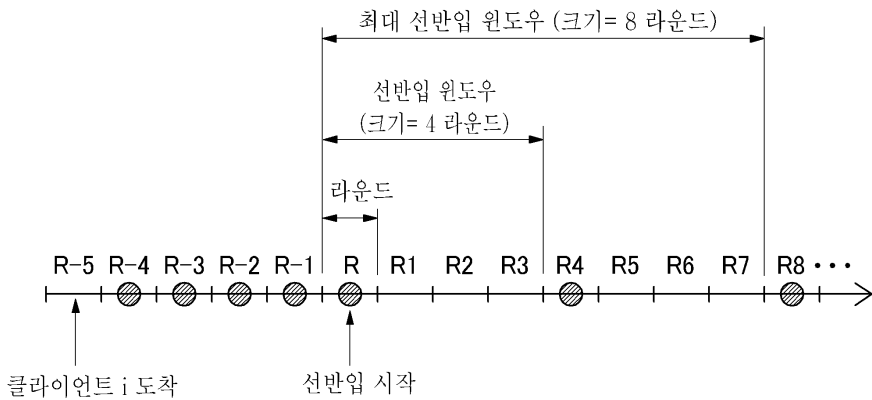
[0072] 10: 미디어 서버
20: 클라이언트
100: 선반입 관리부
200: 버퍼
300: 디스크 집합

도면

도면1



도면2



도면3

Symbols	Meaning
T^R	Round length (s)
N^d, N^c	Number of disks and clients
T_m^{seek}, r_m	Seek time (s) and transfer rate (MB/s) of disk m
P_m^{seek}	Seek power of disk m (W)
P_m^{active}	Active power of disk m (W)
P_m^{idle}	Idle power of disk m (W)
b_i, d_i	Bit-rate (Mb/s) and index of disk serving client i
N^p	Maximum size of prefetching window (rounds)
V_j	Window size with index j (rounds)
N^v	Number of divisors of N^p (This also means the number of window size indices.)
N_j^c	Number of prefetching round indices for window size with index j
$I_{j,k}$	k^{th} list in an array of possible prefetching rounds when the window size index is j
$C_{i,(j,k)}$	Case in which the window size index j and the index of lists of prefetching rounds k are selected for client i
R	Round index at which a new sequence of N^p rounds begins
W_i	Selected window size index for client i
S_i	Selected prefetching round index for client i
$D_{i,(j,k)}(n)$	Number of rounds between round n , and the next round during which prefetching for client i occurs
$B_{i,(j,k)}(n)$	Buffer requirement for $C_{i,(j,k)}$ during round n
$E_{i,j}^{\text{saving}}$	Energy saved by $C_{i,(j,k)}$ (J)
B^{max}	Maximum buffer size (MB)
$E(n)$	Disk energy consumption during the round n (J)
$E_m^{\text{idle}}(n)$	Idle energy used during round n by disk m (J)
$E_m^{\text{active}}(n)$	Active energy used during round n by disk m (J)
$E_m^{\text{seek}}(n)$	Seek energy used during round n by disk m (J)

도면4

$$(수식 1) \quad B_{i,(j,k)}(n) = (1 + D_{i,(j,k)}(n))b_i T^R.$$

$$(수식 2) \quad E_{i,j}^{\text{saving}} = (N^p - \frac{N^p}{v_j}) T_{d_i}^{\text{seek}} (P_{d_i}^{\text{seek}} - P_{d_i}^{\text{idle}}).$$

$$(수식 3) \quad \begin{aligned} & \text{maximize} \quad \sum_{i=1}^{N^c} E_{i,W_i}^{\text{saving}} \\ & \text{s.t.} \quad B_{i,(W_i,S_i)}(n) \leq B^{\text{max}}, (\forall n = R, \dots, R + N^p - 1). \end{aligned}$$

$$(수식 4) \quad Q_{i,j} = \frac{E_{i,j}^{\text{saving}}}{\max_{n=R, \dots, R+N^p-1} B_{i,(j,1)}(n) - \max_{n=R, \dots, R+N^p-1} B_{i,(1,1)}(n)}.$$

$$(수식 5) \quad E_m^{\text{seek}}(n) = \sum_{\forall i \quad d_i \in m} \sum_{\forall n \in I_{W_i, S_i}} T_m^{\text{seek}} P_m^{\text{seek}}.$$

$$(수식 6) \quad E_m^{\text{active}}(n) = \sum_{\forall i \quad d_i \in m} \sum_{\forall n \in I_{W_i, S_i}} \frac{V_{W_i} b_i T^R}{r_m} P_m^{\text{active}}.$$

$$(수식 7) \quad E_m^{\text{idle}}(n) = P_m^{\text{idle}} (T^R - \sum_{\forall i, d_i \in m} \sum_{\forall n \in I_{W_i, S_i}} (T_m^{\text{seek}} + \frac{V_{W_i} b_i T^R}{r_m})).$$

$$(수식 8) \quad E(n) = \sum_{m=1}^{N^d} (E_m^{\text{seek}}(n) + E_m^{\text{active}}(n) + E_m^{\text{idle}}(n)).$$

도면5

```

1: A: array of all values of  $Q_{i,j}$ ;
2:  $W_i$ : index of window size for client  $i$ ;
3:  $b(n)$ : buffer requirement during round  $n$ ;
4:  $S_i$ : index of prefetching rounds for client  $i$ ;
5:  $I^c$ : temporary index for client;
6:  $I^w$ : temporary index for window size;
7:  $I^p$ : temporary index for prefetching rounds;
8:  $T_k$ : temporary variable for prefetching round index of  $k$ ;
9:  $W_i \leftarrow 1, (i = 1, \dots, N^c)$ ;
10:  $b(n) \leftarrow \sum_{i=1}^{N^c} B_{i,(1,1)}(n), (n = 1, \dots, N^p)$ ;
11: while  $A \neq \phi$  do
12:   Find client index of  $I^c$  and window index of  $I^w$  that has the
   highest value of  $Q_{I^c, I^w} \in A$  and removes  $Q_{I^c, I^w}$  from  $A$ ;
13:   if  $W_{I^c} < I^w$  then
14:     for  $k = 1$  to  $k = N_{I^w}^c$  do
15:        $T_k \leftarrow \max_{n=1, \dots, N^p} (b(n) + B_{I^c, (I^w, k)}(n))$ ;
16:     end for
17:     Find the index  $I^p$  of prefetching rounds that corresponds to
     the smallest value of  $T_{I^p}$ ;
18:     if  $T_{I^p} \leq B^{\max}$  then
19:       for  $n = R$  to  $R + N^p - 1$  do
20:          $b(n) \leftarrow b(n) + B_{I^c, (I^w, I^p)}(n) - B_{I^c, (W_{I^c}, S_{I^c})}(n)$ ;
21:       end for
22:        $W_{I^c} \leftarrow I^w$ ;
23:        $S_{I^c} \leftarrow I^p$ ;
24:     end if
25:   end if
26: end while

```

도면6

