



(19) 대한민국특허청(KR)

(12) 등록특허공보(B1)

(45) 공고일자 2015년07월09일

(11) 등록번호 10-1535530

(24) 등록일자 2015년07월03일

(51) 국제특허분류(Int. Cl.)

G06F 15/16 (2006.01) G06F 21/60 (2013.01)

(21) 출원번호 10-2013-0066954

(22) 출원일자 2013년06월12일

심사청구일자 2013년07월23일

(65) 공개번호 10-2014-0144858

(43) 공개일자 2014년12월22일

(56) 선행기술조사문헌

JP4945715 B*

JP2007122336 A

JP2008140049 A

JP2006331411 A

*는 심사관에 의하여 인용된 문헌

(73) 특허권자

서울대학교산학협력단

서울특별시 관악구 관악로 1 (신림동)

(72) 발명자

박영훈

서울 영등포구 당산로 214, 405동 2001호 (당산동5가, 삼성래미안4차아파트)

서승우

서울 관악구 관악로 1, 뉴미디어통신공동연구소 132동 405호 (신림동, 서울대학교)

(74) 대리인

김영철, 김 순 영

전체 청구항 수 : 총 20 항

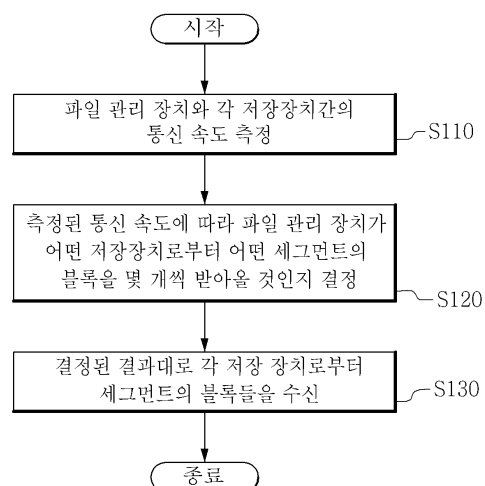
심사관 : 김병균

(54) 발명의 명칭 일정 수 이상의 파일 조각으로 원본 파일을 복원하는 파일 관리 장치 및 그것의 파일 관리 방법

(57) 요약

본 발명은 복수의 저장 장치들로부터 원본 파일을 복원하는 파일 관리 장치에 있어서, 원본 파일을 복원하는 데 소요되는 시간 및 통신량을 최소화하는 기술을 제공한다. 본 발명이 파일 관리 방법은 복수의 저장 장치들 중 파일 조각을 수신할 저장 장치들을 선택하는 단계, 파일 관리 장치와 선택된 저장 장치들 사이의 통신 속도들에 따라, 선택된 저장 장치들로부터 수신할 파일 조각들을 나타내는 스케줄링 정보를 결정하는 단계, 결정된 스케줄링 정보를 참조하여, 상기 선택된 저장 장치들 각각으로부터 파일 조각들을 수신하는 단계 및 수신된 파일 조각들로부터 복원 파일을 생성하는 단계를 포함한다.

대표도 - 도11



명세서

청구범위

청구항 1

복수의 저장 장치들 중 파일 조각의 송신을 요청할 저장 장치들을 선택하는 단계;

상기 파일 조각의 송신을 요청할 저장 장치들을 선택한 이후, 파일 수신 장치와 상기 선택된 저장 장치들 사이의 통신 속도들에 따라, 상기 선택된 저장 장치들 각각에 저장된 파일 조각 중 수신할 부분의 크기를 결정함으로써 상기 선택된 저장 장치들로부터 수신할 각각의 파일 조각에 어떤 세그먼트들이 포함되는지를 나타내는 스케줄링 정보를 결정하는 단계;

상기 결정된 스케줄링 정보를 참조하여, 상기 선택된 저장 장치들 각각으로부터 저장된 파일 조각의 일부 또는 전부를 수신하는 단계; 및

상기 수신된 파일 조각들로부터 복원 파일을 생성하는 단계를 포함하되,

상기 복원 파일은 nC_{k-1} 개의 세그먼트들로 구성되고,

상기 복수의 저장 장치들 각각은 상기 nC_{k-1} 개의 세그먼트들 중 적어도 일부의 복수의 세그먼트들을 세그먼트 단위로 저장하고,

상기 수신된 파일 조각들 각각은 상기 복수의 저장 장치들 중 대응하는 저장 장치에 저장된 상기 적어도 일부의 복수의 세그먼트들을 전부 또는 부분적으로 포함하고,

상기 n 은 상기 복수의 저장 장치들의 개수이고,

상기 k 는 선택된 저장 장치들의 개수인, 파일 관리 방법.

청구항 2

제 1 항에 있어서,

상기 복수의 저장 장치들 각각에 저장된 파일 조각은 상기 nC_{k-1} 개의 세그먼트들 중 $n-1C_{k-1}$ 개의 세그먼트들을 각각 포함하는, 파일 관리 방법.

청구항 3

제 1 항에 있어서,

상기 수신된 파일 조각들은 상기 복원 파일을 구성하는 상기 nC_{k-1} 개의 세그먼트들을 모두 포함하는, 파일 관리 방법.

청구항 4

제 1 항에 있어서,

각각의 상기 세그먼트는 적어도 하나의 블록을 포함하며,

상기 수신된 파일 조각들 각각은 서로 중복되는 블록을 포함하지 않는, 파일 관리 방법.

청구항 5

제 1 항에 있어서,

상기 스케줄링 정보를 결정하는 단계는,

상기 통신 속도들을 참조하여, 상기 nC_{k-1} 개의 세그먼트들을 송신할 저장 장치 간의 우선 순위를 결정하는 단계;
및

상기 결정된 우선 순위에 따라, 상기 선택된 저장 장치들에 상기 nC_{k-1} 개의 세그먼트들 각각을 배타적으로 송신하는 일을 할당하는 단계를 포함하는, 파일 관리 방법.

청구항 6

제 5 항에 있어서,

상기 스케줄링 정보를 결정하는 단계는,

상기 선택된 저장 장치들 중 적어도 하나의 저장 장치에 대해서, 상기 하나의 저장 장치에 송신이 할당된 블록들의 개수가 상기 하나의 저장 장치와 대응하는 통신 속도에 비해 과다인지 또는 과소인지 결정하는 단계; 및

상기 과다인지 또는 과소인지 결정한 결과에 따라, 상기 선택된 저장 장치들에 송신이 할당된 블록들 중 적어도 일부를 송신하는 일을 재할당하는 단계를 더 포함하는, 파일 관리 방법.

청구항 7

제 6 항에 있어서,

상기 적어도 일부를 송신하는 일을 재할당하는 단계는,

상기 하나의 저장 장치에 송신이 할당된 블록들의 개수가 상기 하나의 저장 장치와 대응하는 통신 속도에 비해 과다이면, 상기 하나의 저장 장치에 송신이 할당된 블록들의 적어도 일부를 송신하는 일을 상기 선택된 저장 장치들 중 다른 저장 장치에 재할당하는 단계를 포함하는, 파일 관리 방법.

청구항 8

제 6 항에 있어서,

상기 적어도 일부를 송신하는 일을 재할당하는 단계는,

상기 하나의 저장 장치에 송신이 할당된 블록들의 개수가 상기 하나의 저장 장치와 대응하는 통신 속도에 비해 과소이면, 상기 선택된 저장 장치들 중 다른 저장 장치에 송신이 할당된 블록들의 적어도 일부를 송신하는 일을 상기 하나의 저장 장치에 재할당하는 단계를 포함하는, 파일 관리 방법.

청구항 9

제 6 항에 있어서,

상기 적어도 일부를 송신하는 일을 재할당하는 단계는,

상기 결정된 우선 순위를 참조하여, 상기 선택된 저장 장치들에 송신이 할당된 블록들 중 적어도 일부를 송신하는 일을 재할당하는, 파일 관리 방법.

청구항 10

제 9 항에 있어서,

상기 적어도 일부를 송신하는 일을 재할당하는 단계는,

상기 선택된 저장 장치에 송신이 할당된 세그먼트들이 상기 선택된 저장 장치들 중 얼마나 많은 저장 장치들에 공통적으로 저장되어 있는지를 나타내는 함수를 참조하여 상기 적어도 일부를 송신하는 일을 재할당하는, 파일 관리 방법.

청구항 11

제 6 항에 있어서,

상기 적어도 일부를 송신하는 일을 재할당하는 단계는,

상기 선택된 저장장치 각각이 상기 파일 수신 장치와의 통신 속도가 빠를수록 더 많은 수의 블록들을 송신하도록, 상기 적어도 일부를 송신하는 일을 재할당하는, 파일 관리 방법.

청구항 12

제 1 항에 있어서,

상기 복원 파일은, 상기 선택된 저장 장치들의 개수가 소정의 수 이상일 때만 성공적으로 생성되는, 파일 관리 방법.

청구항 13

복수의 저장 장치들로부터 파일 조각들을 수신하는 파일 수신 장치에 있어서,

상기 복수의 저장 장치들 중 선택된 저장 장치들과 상기 파일 수신 장치 사이의 통신 속도들에 따라, 상기 선택된 저장 장치들 각각에 저장된 파일 조각 중 수신할 부분의 크기를 결정함으로써 상기 선택된 저장 장치들로부터 수신할 각각의 파일 조각에 어떤 세그먼트들이 포함되는지를 나타내는 스케줄링 정보를 결정하는 스케줄러;

상기 선택된 저장 장치들과의 통신을 수행하거나 상기 선택된 저장 장치들과의 통신을 위한 인터페이스를 제공하는 통신부; 및

상기 스케줄링 정보를 참조하여 상기 통신부를 통해 상기 선택된 저장 장치들 각각으로부터 저장된 파일 조각의 일부 또는 전부를 수신하고, 상기 수신된 파일 조각들로부터 복원 파일을 생성하도록 상기 파일 수신 장치를 제어하는 컨트롤러를 포함하되,

상기 복원 파일은 nC_{k-1} 개의 세그먼트들로 구성되고,

상기 복수의 저장 장치들 각각은 상기 nC_{k-1} 개의 세그먼트들 중 $n-1C_{k-1}$ 개의 세그먼트들을 세그먼트 단위로 저장하고,

상기 수신된 파일 조각들 각각은 상기 복수의 저장 장치들 중 대응하는 저장 장치에 저장된 상기 $n-1C_{k-1}$ 개의 세그먼트들을 전부 또는 부분적으로 포함하고,

상기 n 은 상기 복수의 저장 장치들의 개수이고,

상기 k 는 선택된 저장 장치들의 개수인, 파일 수신 장치.

청구항 14

제 13 항에 있어서,

상기 컨트롤러는 상기 선택된 저장 장치들을 상기 복수의 저장 장치들 중에서 선택하는, 파일 수신 장치.

청구항 15

제 13 항에 있어서,

상기 스케줄러는,

상기 통신 속도들을 참조하여 상기 nC_{k-1} 개의 세그먼트들을 송신할 저장 장치 간의 우선 순위를 결정하고, 상기 결정된 우선 순위에 따라 상기 선택된 저장 장치들에 상기 nC_{k-1} 개의 세그먼트들 각각을 배타적으로 송신하는 일을 할당하는, 파일 수신 장치.

청구항 16

제 15 항에 있어서,

각각의 상기 세그먼트는 적어도 하나의 블록을 포함하며,

상기 스케줄러는,

상기 선택된 저장 장치들 중 적어도 하나의 저장 장치에 대해서, 상기 하나의 저장 장치에 송신이 할당된 블록들의 개수가 상기 하나의 저장 장치와 대응하는 통신 속도에 비해 과다인지 또는 과소인지 결정하고, 상기 과다인지 또는 과소인지 결정한 결과에 따라 상기 선택된 저장 장치들에 송신이 할당된 블록들 중 적어도 일부를 송신하는 일을 재할당하는, 파일 수신 장치.

청구항 17

제 16 항에 있어서,

상기 스케줄러는,

상기 우선 순위를 참조하여, 상기 선택된 저장 장치들에 송신이 할당된 블록들 중 적어도 일부를 송신하는 일을 재할당하는, 파일 수신 장치.

청구항 18

제 16 항에 있어서,

상기 스케줄러는,

상기 선택된 저장 장치 각각이 상기 통신 속도들 중 대응되는 통신 속도가 빠를수록 더 많은 수의 블록들을 송신하도록, 상기 적어도 일부를 송신하는 일을 재할당하는, 파일 수신 장치.

청구항 19

제 13 항에 있어서,

상기 복원 파일은, 상기 선택된 저장 장치들의 개수가 소정의 수 이상일 때만 성공적으로 생성되는, 파일 수신 장치.

청구항 20

복수의 저장 장치들 중 파일 조각의 송신을 요청할 저장 장치들을 선택하는 단계, 상기 파일 조각의 송신을 요

청할 저장 장치들을 선택한 이후, 파일 수신 장치와 상기 선택된 저장 장치들 사이의 통신 속도들에 따라, 상기 선택된 저장 장치들 각각에 저장된 파일 조각 중 수신할 부분의 크기를 결정함으로써 상기 선택된 저장 장치들로부터 수신할 각각의 파일 조각에 어떤 세그먼트들이 포함되는지를 나타내는 스케줄링 정보를 결정하는 단계, 상기 결정된 스케줄링 정보를 참조하여, 상기 선택된 저장 장치들 각각으로부터 저장된 파일 조각의 일부 또는 전부를 수신하는 단계 및 상기 수신된 파일 조각들로부터 복원 파일을 생성하는 단계를 포함하되, 상기 복원 파일은 nC_{k-1} 개의 세그먼트들로 구성되고, 상기 복수의 저장 장치들 각각은 상기 nC_{k-1} 개의 세그먼트들 중 적어도 일부의 복수의 세그먼트들을 세그먼트 단위로 저장하고, 상기 수신된 파일 조각들 각각은 상기 복수의 저장 장치들 중 대응하는 저장 장치에 저장된 상기 적어도 일부의 복수의 세그먼트들을 전부 또는 부분적으로 포함하고, 상기 n 은 상기 복수의 저장 장치들의 개수이고, 상기 k 는 선택된 저장 장치들의 개수인, 파일 관리 방법을 실행하기 위한 컴퓨터 프로그램을 기록하는 컴퓨터 판독 가능한 기록 매체.

발명의 설명

기술 분야

본 발명은 복수의 저장 장치에 분산 저장된 파일 조각들로부터 원본 파일을 복원하는 파일 관리 장치 및 그것의 파일 관리 방법에 관한 것으로서, 더욱 구체적으로는 복수의 저장 장치의 통신 속도를 고려하여 파일 조각들을 다운로드함으로써 원본 파일의 복원 시간을 최소화하는 파일 관리 장치 및 그것의 파일 관리 방법에 관한 것이다.

배경 기술

비밀 정보를 보관하는 경우, 비밀 정보가 분실되거나, 파괴될 리스크는 항상 존재한다. 이와 동시에 비밀정보가 도난될 수 있는 리스크 역시 존재한다. 분실 또는 파괴의 리스크는 비밀 정보를 복수 곳의 장소에 보관해 둬으로써 줄일 수 있지만, 이 경우 도난의 리스크가 증가하는 단점이 있다. 이들 리스크를 함께 해결하는 방법의 하나로서 비밀 분산법(Secret Sharing)이 제시되었다.

비밀 분산법은 비밀 정보 MSK로부터 복수의 쉼어 정보 SH(1), ..., SH(N)를 생성하고, 이들을 복수의 분산 관리 장치 PA(1), ..., PA(N)에 분산하여 관리시키고, 이들 분산 정보 SH(1), ..., SH(N) 중 소정 수 이상의 정보를 얻을 수 있었던 경우에만, 비밀 정보 MSK를 복원할 수 있는 방식이다.

비밀 분산법은 저장되는 값의 기밀성, 가용성, 무결성을 모두 보장해줄 수 있다는 장점 덕분에 많은 분산 저장 장치에 적용되어 왔다. 저장된 파일(F)을 작은 단위인 값(F[1], F[2], ..., F[s], 단, s는 파일을 이루는 값의 개수)으로 쪼갬 뒤, 각각의 값 F[i]들을 비밀 분산법을 이용하여 복수의 쉼어정보 f[i,1], f[i,2], ..., f[i,n] (단, n은 저장 장치의 개수)을 만들어 내고, 쉼어정보를 $f(s) = f[1,m] || f[2,m] || \dots || f[s,m]$ (단, $m=1,2,\dots,n$) 과 같이 연결하여 n개의 파일 조각 f(1), f(2), ..., f(n)을 만들어 낸다.

기존 비밀 분산법이 적용된 분산 저장 장치는 일정 수 미만의 파일 조각으로 전체 파일의 내용을 알 수 없으므로 저장된 파일의 기밀성을 제공할 수 있었고, 일정 수 미만의 파일 조각이 없어지거나 손상되더라도 나머지 파일 조각으로 파일을 복원할 수 있기 때문에 가용성을 보장해줄 수 있었다.

하지만, 최근 들어 저장되는 파일의 크기가 커지면서 연산량이 많은 기존 비밀 분산법을 직접적으로 적용하기에는 무리가 있으며, 또한 파일 조각의 크기가 원래의 파일의 크기와 동일하므로 저장 공간 및 통신 비용의 낭비가 커질 가능성이 있다.

이러한 문제점을 해결하기 위해 선행 특허 한국특허출원 제 10-2013-0424811 호가 제안되었다. 하지만, 이 선행 특허를 분산 저장 시스템에 적용할 경우, 여러 분산된 저장 장치로부터 중복된 파일 블록들이 전송될 수 있으며, 이 경우 파일 분산 관리 장치가 파일을 복원하기 위해 수신하는 데이터의 양이 매우 커질 수 있는 문제점이 있었다.

선행기술문헌

특허문헌

(특허문헌 0001) 한국특허출원 제 10-2013-0424811 호

발명의 내용

해결하려는 과제

본 발명의 목적은 분산 저장된 파일 조각들로부터 원본 파일을 복원할 때, 저장 장치와의 통신 속도를 고려함으로써 복원 시간을 최소화하는 파일 분산 관리 장치 및 방법을 제공하는 데 있다.

본 발명의 다른 목적은 분산 저장된 파일 조각들로부터 원본 파일을 복원할 때 중복되는 블록이 수신되지 않도록 함으로써, 복원 시간 및 필요 통신량을 최소화하고 통신 비용을 절감시키는 파일 분산 관리 장치 및 방법을 제공하는 데 있다.

본 발명의 다른 목적은 분산 저장된 파일 조각들 중 일정 수 이상의 파일 조각을 획득할 때만 원본 파일이 복원되도록 함으로써, 파일 분산 관리 장치 및 방법의 보안성과 기밀성을 향상시키는 데 있다.

과제의 해결 수단

본 발명의 파일 관리 방법은 복수의 저장 장치들 중 파일 조각을 수신할 저장 장치들을 선택하는 단계; 파일 관리 장치와 상기 선택된 저장 장치들 사이의 통신 속도들에 따라, 상기 선택된 저장 장치들로부터 수신할 파일 조각들을 나타내는 스케줄링 정보를 결정하는 단계; 상기 결정된 스케줄링 정보를 참조하여, 상기 선택된 저장 장치들 각각으로부터 파일 조각들을 수신하는 단계; 및 상기 수신된 파일 조각들로부터 복원 파일을 생성하는 단계를 포함하되, 상기 복원 파일은 nC_{k-1} 개의 세그먼트들로 구성되고, 상기 복수의 저장 장치들 각각은 상기 nC_{k-1} 개의 세그먼트들 중 적어도 일부의 세그먼트들을 세그먼트 단위로 저장하고, 상기 수신된 파일 조각들 각각은 상기 복수의 저장 장치들 중 대응하는 저장 장치에 저장된 상기 적어도 일부의 세그먼트들을 전부 또는 부분적으로 포함하고, 상기 n 은 상기 복수의 저장 장치들의 개수이고, 상기 k 는 선택된 저장 장치들의 개수이다.

실시 예로서, 상기 복수의 저장 장치들 각각은 상기 nC_{k-1} 개의 세그먼트들 중 $n-1C_{k-1}$ 개의 세그먼트들을 각각 포함한다.

실시 예로서, 상기 수신된 파일 조각들은 상기 복원 파일을 구성하는 상기 nC_{k-1} 개의 세그먼트들을 모두 포함하고, 상기 수신된 파일 조각들은 상기 복수의 저장 장치들에 저장된 상기 n 개의 파일 조각들 중 상기 k 개의 임의의 파일 조각들이다.

실시 예로서, 상기 n 개의 파일 조각들 중 상기 k 개보다 적은 수의 임의의 파일 조각들은 상기 nC_{k-1} 개의 세그먼트들 중 적어도 일부를 불포함한다.

실시 예로서, 상기 수신된 파일 조각들 각각은 서로 중복되는 블록을 포함하지 않는다.

실시 예로서, 상기 스케줄링 정보를 결정하는 단계는, 상기 통신 속도들을 참조하여, 상기 nC_{k-1} 개의 세그먼트들의 할당을 위한 상기 수신할 파일들의 우선 순위를 결정하는 단계; 및 상기 결정된 우선 순위에 따라, 상기 nC_{k-1} 개의 세그먼트들 각각을 상기 수신할 파일 조각들에 배타적으로 할당하는 단계를 포함한다.

실시 예로서, 상기 스케줄링 정보를 결정하는 단계는, 상기 수신할 파일 조각들 중 적어도 하나의 파일 조각에 대해서, 상기 하나의 파일 조각에 할당된 블록들의 개수가 상기 하나의 파일 조각과 대응하는 통신 속도에 비해 과다인지 또는 과소인지 결정하는 단계; 및 상기 과다인지 또는 과소인지 결정된 결과에 따라, 상기 수신할 파일 조각들에 포함된 블록들 중 적어도 일부를 재할당하는 단계를 더 포함한다.

실시 예로서, 상기 적어도 일부를 재할당하는 단계는, 상기 하나의 파일 조각에 할당된 블록들의 개수가 상기 하나의 파일 조각과 대응하는 통신 속도에 비해 과다이면, 상기 하나의 파일 조각에 포함된 블록들의 적어도 일부를 상기 수신할 파일 조각들 중 다른 파일 조각에 재할당하는 단계를 포함한다.

실시 예로서, 상기 적어도 일부를 재할당하는 단계는, 상기 하나의 파일 조각에 할당된 블록들의 개수가 상기 하나의 파일 조각과 대응하는 통신 속도에 비해 과소이면, 상기 수신할 파일 조각들 중 다른 파일 조각에 포함된 블록들의 적어도 일부를 상기 하나의 파일 조각에 재할당하는 단계를 포함한다.

실시 예로서, 상기 적어도 일부를 재할당하는 단계는 상기 수신할 파일 조각들에 포함된 세그먼트들이 상기 수신할 파일 조각들 중 얼마나 많은 파일 조각들에 공통적으로 저장되어 있는지를 나타내는 인기도 함수를 참조하여, 상기 수신할 파일 조각들에 포함된 블록들 중 적어도 일부를 재할당한다.

실시 예로서, 상기 인기도 함수는

$$P(w) = \sum_{j=1}^k \delta(w, p_j)$$

$$\text{단, } \delta(w, p_j) = \begin{cases} r_{p_j} - X_{p_j} & , r_{p_j} > X_{p_j}, w \in S_{p_j} \\ 0 & , \text{otherwise} \end{cases}$$

의 수학적식으로 정의되고, 상기 w 는 상기 수신할 파일 조각들에 포함된 상기 세그먼트들의 번호를 나타내는 값이고, 상기 p_j 는 상기 수신할 파일 조각들을 제공하는 저장 장치들의 번호를 나타내는 값이고, 상기 r_{p_j} 는 상기 선택된 저장 장치들의 통신 속도에 비례하도록 상기 수신할 파일 조각에 포함된 블록들의 전체 개수를 나눈 값들 중 p_j 번째 저장 장치에 대응되는 값이고, 상기 X_{p_j} 는 상기 수신할 파일 조각들 중 상기 p_j 번째 저장 장치로부터 수신할 파일 조각에 할당된 블록들의 개수이고, 상기 S_{p_j} 는 상기 p_j 번째 저장 장치에 대응되는 파일 조각을 구성하는 세그먼트들을 지시하는 번호들의 집합이다.

실시 예로서, 상기 적어도 일부를 재할당하는 단계는 상기 수신할 파일 조각들 각각이 상기 통신 속도들 중 대응되는 통신 속도가 빠를수록 더 많은 수의 블록들을 포함하도록, 상기 수신할 파일 조각들에 포함된 블록들 중 적어도 일부를 재할당한다.

실시 예로서, 상기 복원 파일은, 상기 선택된 저장 장치들의 개수가 소정의 수 이상일 때만 성공적으로 생성된다.

본 발명에 따른 파일 관리 장치는 복수의 저장 장치들 중 선택된 저장 장치들과 상기 파일 관리 장치 사이의 통신 속도들에 따라, 상기 선택된 저장 장치들로부터 수신할 파일 조각들을 나타내는 스케줄링 정보를 결정하는 스케줄러; 상기 선택된 저장 장치들과의 통신을 수행하거나 상기 선택된 저장 장치들과의 통신을 위한 인터페이스를 제공하는 통신부; 및 상기 스케줄링 정보를 참조하여 상기 통신부를 통해 상기 선택된 저장 장치들 각각으로부터 파일 조각들을 수신하고, 상기 수신된 파일 조각들로부터 복원 파일을 생성하도록 상기 파일 관리 장치를 제어하는 컨트롤러를 포함하되, 상기 복원 파일은 nC_{k-1} 개의 세그먼트들로 구성되고, 상기 복수의 저장 장치들 각각은 상기 nC_{k-1} 개의 세그먼트들 중 $n-1C_{k-1}$ 개의 세그먼트들을 세그먼트 단위로 저장하고, 상기 수신된 파일 조각들 각각은 상기 복수의 저장 장치들 중 대응하는 저장 장치에 저장된 상기 적어도 일부의 세그먼트들을 전부 또는 부분적으로 포함하고, 상기 n 은 상기 복수의 저장 장치들의 개수이고, 상기 k 는 선택된 저장 장치들의 개수이다.

실시 예로서, 상기 컨트롤러는 상기 선택된 저장 장치들을 상기 복수의 저장 장치들 중에서 선택한다.

실시 예로서, 상기 스케줄러는 상기 통신 속도들을 참조하여 상기 nC_{k-1} 개의 세그먼트들의 할당을 위한 상기 수신할 파일들의 우선 순위를 결정하고, 상기 결정된 우선 순위에 따라 상기 nC_{k-1} 개의 세그먼트들 각각을 상기 수신할 파일 조각들에 배타적으로 할당한다.

실시 예로서, 상기 스케줄러는 상기 수신할 파일 조각들 중 적어도 하나의 파일 조각에 대해서, 상기 하나의 파일 조각에 할당된 블록들의 개수가 상기 하나의 파일 조각과 대응하는 통신 속도에 비해 과다인지 또는 과소인지 결정하고, 상기 과다인지 또는 과소인지 결정한 결과에 따라 상기 수신할 파일 조각들에 포함된 블록들 중 적어도 일부를 재할당한다.

실시 예로서, 상기 스케줄러는 상기 수신할 파일 조각들에 포함된 세그먼트들이 상기 수신할 파일 조각들 중 얼마나 많은 파일 조각들에 공통적으로 저장되어 있는지를 나타내는 인기도 함수를 참조하여, 상기 수신할 파일 조각들에 포함된 블록들 중 적어도 일부를 재할당한다.

실시 예로서, 상기 스케줄러는 상기 수신할 파일 조각들 각각이 상기 통신 속도들 중 대응되는 통신 속도가 빠를수록 더 많은 수의 블록들을 포함하도록, 상기 수신할 파일 조각들에 포함된 블록들 중 적어도 일부를 재할당

한다.

실시 예로서, 상기 복원 파일은, 상기 선택된 저장 장치들의 개수가 소정의 수 이상일 때만 성공적으로 생성된다.

본 발명에 따른 컴퓨터 판독 가능한 기록 매체는 복수의 저장 장치들 중 파일 조각을 수신할 저장 장치들을 선택하는 단계, 파일 관리 장치와 상기 선택된 저장 장치들 사이의 통신 속도들에 따라, 상기 선택된 저장 장치들로부터 수신할 파일 조각들을 나타내는 스케줄링 정보를 결정하는 단계, 상기 결정된 스케줄링 정보를 참조하여, 상기 선택된 저장 장치들 각각으로부터 파일 조각들을 수신하는 단계 및 상기 수신된 파일 조각들로부터 복원 파일을 생성하는 단계를 포함하되, 상기 복원 파일은 nC_{k-1} 개의 세그먼트들로 구성되고, 상기 복수의 저장 장치들 각각은 상기 nC_{k-1} 개의 세그먼트들 중 적어도 일부의 세그먼트들을 세그먼트 단위로 저장하고, 상기 수신된 파일 조각들 각각은 상기 복수의 저장 장치들 중 대응하는 저장 장치에 저장된 상기 적어도 일부의 세그먼트들을 전부 또는 부분적으로 포함하고, 상기 n 은 상기 복수의 저장 장치들의 개수이고, 상기 k 는 선택된 저장 장치들의 개수인, 파일 관리 방법을 실행하기 위한 컴퓨터 프로그램을 기록한다.

발명의 효과

본 발명에 따르면 분산 저장된 파일 조각들로부터 원본 파일을 복원할 때, 저장 장치와의 통신 속도를 고려함으로써 복원 시간이 최소화될 수 있다.

또한, 분산 저장된 파일 조각들로부터 원본 파일을 복원할 때 중복되는 블록이 수신되지 않도록 함으로써, 복원 시간 및 필요 통신량이 최소화되고 통신 비용이 절감될 수 있다.

또한, 분산 저장된 파일 조각들 중 일정 수 이상의 파일 조각을 획득할 때만 원본 파일이 복원되도록 함으로써, 파일 분산 관리 장치 및 방법의 보안성과 기밀성이 향상될 수 있다.

도면의 간단한 설명

도 1은 본 발명의 실시 예에 따른, 파일 관리 시스템을 나타내는 블록도이다.

도 2는 본 발명의 실시 예에 따른, 파일 관리 시스템이 저장 장치들에 저장된 파일 조각들로부터 원본 파일을 복원하는 방법을 예시적으로 나타내는 개념도이다.

도 3은 도 1에 도시된 파일 관리 장치를 구체적으로 나타내는 블록도이다.

도 4는 본 발명의 실시 예에 따른, $k=4$, $n=6$ 인 경우의 재배열된 파일(F')을 예시적으로 나타내는 도면이다.

도 5 및 도 6은 도 4에 도시된 재배열된 파일(F')에 대응되는 DT 및 PT를 나타내는 행렬도이다.

도 7은 도 4 내지 도 6의 실시 예에 있어서, 각 저장 장치들에 저장되는 파일 조각들을 나타내는 도면이다.

도 8 내지 도 10은 본 발명의 실시 예에 따른, 스케줄링 테이블(ST)을 결정하는 방법을 순차적으로 나타내는 행렬도이다.

도 11은 본 발명의 실시 예에 따른, 파일 관리 방법을 나타내는 순서도이다.

도 12는 본 발명의 실시 예에 따른, 파일 관리 장치가 스케줄링 테이블(ST)을 참조하여 저장 장치들로부터 파일 조각들을 전송받아 원본 파일을 복원하는 방법을 나타내는 개념도이다.

발명을 실시하기 위한 구체적인 내용

후술하는 본 발명에 대한 상세한 설명은, 본 발명이 실시될 수 있는 특정 실시예를 예시로서 도시하는 첨부 도면을 참조한다. 이들 실시예는 당업자가 본 발명을 실시할 수 있기에 충분하도록 상세히 설명된다. 본 발명의 다양한 실시예는 서로 다르지만 상호 배타적일 필요는 없음이 이해되어야 한다. 예를 들어, 여기에 기재되어 있는 특정 형상, 구조 및 특성은 일 실시예에 관련하여 본 발명의 정신 및 범위를 벗어나지 않으면서 다른 실시예로 구현될 수 있다. 또한, 각각의 개시된 실시예 내의 개별 구성요소의 위치 또는 배치는 본 발명의 정신 및 범위를 벗어나지 않으면서 변경될 수 있음이 이해되어야 한다. 따라서, 후술하는 상세한 설명은 한정적인 의미로서 취하려는 것이 아니며, 본 발명의 범위는, 적절하게 설명된다면, 그 청구항들이 주장하는 것과 균등한 모든 범위와 더불어 첨부된 청구항에 의해서만 한정된다. 도면에서 유사한 참조부호는 여러 측면에 걸쳐서 동일하게

나 유사한 기능을 지칭한다.

한편, 본 발명의 선행 특허 한국특허출원 제 10-2013-0424811 호는 여기서 그 전체가 참조로서 인용될 수 있다. 또한, 파일의 분산 저장 및 복원에 있어서, 본 발명에서 설명되지 않은 것들에 대해서는 한국특허출원 제 10-2013-0424811 호의 내용이 차용될 수 있다. 예를 들어, 본 발명에 따른 저장 장치들에 저장된 파일 조각들은 한국특허출원 제 10-2013-0424811 호에 기재된 방법에 의해 생성된 것일 수 있다.

도 1은 본 발명의 일 실시 예에 따른, 파일 관리 시스템(1000)을 나타내는 도면이다. 파일 관리 시스템(1000)은 파일 관리 장치(1100) 및 n 개의 저장 장치들(1200)로 구성된다.

파일 관리 시스템(1000)은 원본 파일로부터 생성한 복수의 파일 조각들을 n 개의 저장 장치들(1200)에 분산하여 저장한다. 그리고, 파일 관리 시스템(1000)이 분산된 파일 조각들로부터 원본 파일을 복원하기 위해서는, 저장 장치들(1200) 중 임의의 k 개(k 는 0보다 큰 n 이하의 정수) 저장 장치들에 저장된 k 개의 파일 조각들을 획득하여야 한다. 즉, 파일 관리 시스템(1000)이 k 개의 파일 조각들을 획득하는 경우 원본 파일이 완전히 복원될 수 있지만, k 개보다 적은 수의 파일 조각들만을 획득하는 경우 원본 파일은 완전히 복원되지 않는다. 이와 관련된 분산 및 복원 원리에 관한 구체적인 설명은 한국특허출원 제 10-2013-0424811 호에 상세히 기재되어 있다.

저장 장치들(1200)은 n 개의 저장 장치들(1200_1, 1200_2, ..., 1200_ n)을 포함할 수 있다. 저장 장치들(1200) 각각은 원본 파일로부터 생성된 파일 조각을 저장한다. 실시 예로서, 저장 장치들(1200)에는 한국특허출원 제 10-2013-0016390 호에 기재된 방법에 의해 생성된 파일 조각들이 분산 저장될 수 있다. 예를 들어, 원본 파일(또는, 후술할 재배열된 파일)은 nC_{k-1} 개의 세그먼트들로 구성되고, n 개의 저장 장치들(1200_1, 1200_2, ..., 1200_ n) 각각은 상기 nC_{k-1} 개의 세그먼트들 중 적어도 일부의 세그먼트들(예를 들어, $n-1C_{k-1}$ 개의 세그먼트들)을 세그먼트 단위로 저장할 수 있다. 여기서, k 는 원본 파일을 복원하기 위해 필요한 최소한의 저장 장치들의 수이다.

그리고, 저장 장치들(1200)은 파일 관리 장치(1100)의 제어 또는 파일 관리 장치(1100)로부터의 요청에 응답하여 저장된 파일 조각의 일부 또는 전부를 파일 관리 장치(1100)에 제공한다. 실시 예로서, 파일 관리 장치(1100)는 저장 장치들(1200) 각각의 통신 속도를 고려하여, 저장 장치들(1200)로부터 수신할 파일 조각들을 결정하는 스케줄링 동작을 수행할 수 있고, 저장 장치들(1200) 각각은 스케줄링 동작의 결과에 따라 저장된 파일 조각의 전부 또는 일부를 파일 관리 장치(1100)에 전송할 수 있다.

저장 장치들(1200)은 특별한 기능이 없더라도 각각의 파일 조각을 분리하여 저장할 수 있는 장치로서 구성되면 충분하다. 도 1에서, 저장 장치들(1200)은 물리적으로 분리된 형태로 표현되어 있지만, 이에 한정되는 것은 아니며, 저장 장치들(1200)은 1개의 물리적 장치에서 논리적으로 분리된 형태로도 가능할 수 있다.

일 실시예에서, 저장 장치들(1200) 각각은 파일 관리 장치(1100)와의 통신을 위한 별도의 통신부 또는 통신 인터페이스를 구비할 수 있다.

파일 관리 장치(1100)는 저장 장치들로부터 파일 조각들을 수신하고, 수신된 파일 조각들을 이용하여 원본 파일을 복원한다. 파일 관리 장치(1100)에 대한 더욱 상세한 설명은 아래의 도 3을 참조하여 후술된다.

도 2는 본 발명의 실시 예에 따른, 파일 관리 시스템이 저장 장치들에 저장된 파일 조각들로부터 원본 파일을 복원하는 방법을 예시적으로 나타내는 개념도이다.

도 2를 참조하면, 파일 관리 시스템(2000)은 파일 관리 장치(2100) 및 복수의 저장 장치들(2201, 2202, 2203, 2204)을 포함한다. 파일 관리 장치(2100) 및 복수의 저장 장치들(2201, 2202, 2203, 2204)에 관한 구체적인 내용은 도 1의 파일 관리 장치(1100) 및 저장 장치들(1200)과 실질적으로 동일하다. 파일 관리 시스템(2000)에서 원본 파일(100)은 6개의 세그먼트(a, b, c, d, e, f)로 구성되고, 각 세그먼트들(a, b, c, d, e, f)은 4개의 저장 장치들(2201, 2202, 2203, 2204)에 분산 저장된다. 이때, 4개의 저장 장치들(2201, 2202, 2203, 2204) 중 임의의 3개의 저장 장치들(예를 들어, 2201, 2202, 2204)만으로 원본 파일의 복원이 가능하도록, 각 세그먼트들(a, b, c, d, e, f)은 하나 이상의 저장 장치들에 중복하여 저장될 수 있다.

예를 들어, 제 1 저장 장치(2201)에는 세그먼트(a, b, c)를 포함하는 파일 조각(110)이 저장되고, 제 2 저장 장치(2202)에는 세그먼트(a, d, e)를 포함하는 파일 조각(120)이 저장되고, 제 3 저장 장치(2203)에는 세그먼트(b, d, f)를 포함하는 파일 조각(130)이 저장되고, 제 4 저장 장치(2204)에는 세그먼트(c, e, f)를 포함하는 파일 조각(140)이 저장된다고 가정한다. 이때, 4개의 저장 장치들(2201, 2202, 2203, 2204) 중 어떤 3개의 저장 장치들을 선택하여도 선택된 저장 장치들에는 원본 파일의 모든 세그먼트들(a, b, c, d, e, f)이 포함된다. 따

라서, 파일 관리 시스템(2000)은 임의의 3 개의 저장 장치들(예를 들어, 2201, 2202, 2204)에 저장된 파일 조각들을 수신하여 원본 파일을 완전히 복원할 수 있다.

한편, 본 발명에서 파일 관리 시스템(2000)은 분산되어 저장된 파일 조각들을 수신하여 원본 파일을 복원한다. 이때, 파일 관리 장치(2100)는 원본 파일을 복원하는데 소요되는 시간 및 통신 비용을 최소화하기 위해, 저장 장치들(2201, 2202, 2203, 2204)과의 통신 속도 및 각 파일 조각들(110, 120, 130, 140)의 중복성을 고려하여 파일 조각들을 수신한다.

여기서, 파일 관리 장치(2100)가 원본 파일을 복원하기 위해 어떤 저장 장치로부터 수신하는 파일 조각은 해당 저장 장치에 저장된 원본 파일의 파일 조각의 일부 또는 전부일 수 있다. 즉, 파일 관리 장치(2100)는 어떤 저장 장치에 저장된 파일 조각의 일부만을 수신하여 원본 파일을 복원하는데 사용할 수 있다. 따라서, 저장 장치에 저장된 파일 조각과 그 저장 장치로부터 파일 관리 장치(2100)가 수신하는 파일 조각은 서로 다를 수 있다.

구체적으로, 원본 파일의 복원이 가능한 일정 수(여기서는, 3개)의 저장 장치들에 저장된 전체 파일 조각들(110, 120, 140)을 수신하면, 중복된 세그먼트들(예를 들어, a, c, e)이 수신될 수 있고 그에 따른 통신 비용도 증가하게 된다. 이에, 본 발명의 파일 관리 장치(2100)는 중복되는 세그먼트들이 수신되지 않도록, 파일 조각들을 스케줄링한다. 여기서, 파일 조각을 스케줄링하는 것의 의미는, 어떤 저장 장치로부터 수신할 파일 조각에 어떤 세그먼트(또는 세그먼트의 일부)들이 포함되도록 할지 결정하는 것을 의미하며, 파일 관리 장치(2100)는 스케줄링의 결과에 따라 적어도 일부의 세그먼트들로 구성된 파일 조각들을 각 저장 장치들로부터 수신한다. 실시 예로서, 스케줄링의 결과에 따라 구성된 파일 조각들은 파일 조각에 포함된 적어도 일부의 세그먼트들의 주소 또는 크기 정보가 포함될 수 있다.

또한, k개(여기서는, 3개)의 저장 장치들로부터 별다른 조치없이 파일 조각들을 수신하는 경우, 파일 관리 장치(2100)는 통신 속도가 낮은 저장 장치로부터 크기가 큰 파일 조각을 수신하고, 통신 속도가 높은 저장 장치로부터 크기가 작은 파일 조각을 수신하게 될 수 있다. 이러한 경우, 파일 관리 장치(2100)에서 전체 파일 조각들을 수신하는데 걸리는 시간이 매우 커질 수 있다. 따라서, 본 발명의 파일 관리 장치(2100)는 각 저장 장치와의 통신 속도에 따라, 통신 속도가 빠른 저장 장치에 대해서는 상대적으로 크기가 큰 파일 조각을 스케줄링하고, 통신 속도가 느린 저장 장치에 대해서는 상대적으로 크기가 작은 파일 조각을 스케줄링한다.

도 2를 참조하여 구체적인 예를 들면, 파일 관리 시스템(2000)은 원본 파일을 복원하기 위해 4개의 저장 장치들(2201, 2202, 2203, 2204) 중 임의의 3개의 저장 장치들(2201, 2202, 2204)을 선택한다. 그리고, 파일 관리 장치(2100)가 파일 조각을 수신하는 시간이 최소화되도록 하기 위해, 파일 관리 시스템(2000)은 각 저장 장치들(2201, 2202, 2204)의 통신 속도에 비례하여 선택된 각 저장 장치들(2201, 2202, 2204)로부터 전송되는 파일 조각들의 크기를 제어한다.

가령, 파일 관리 장치(2100)와의 통신 속도에 있어서, 제 1 저장 장치(2201)가 가장 빠르고, 제 2 저장 장치(2202)가 다음으로 빠르고, 제 4 저장 장치(2204)가 가장 느리다고 가정한다. 이때, 파일 관리 시스템(3000)은 파일 관리 장치(2100)가 수신(또는, 다운로드)하는 파일 조각들의 크기가 대응되는 저장 장치들의 통신 속도에 비례하도록 제어된다. 즉, 제 1 저장 장치(2201)로부터는 가장 큰 크기의 파일 조각(111)이 수신되고, 제 2 저장 장치(2202)로부터는 다음 크기의 파일 조각(121)이 수신되고, 제 4 저장 장치(2204)로부터는 가장 작은 크기의 파일 조각(141)이 수신된다. 이처럼, 수신되는 파일 조각의 크기가 대응되는 통신 속도에 비례하도록 함으로써, 파일 관리 장치(2100)가 파일 조각들(111, 121, 141)을 수신하는 데 걸리는 시간이 감소될 수 있다.

또한, 파일 관리 시스템(2000)은 각 저장 장치들(2201, 2202, 2203)로부터 수신되는 파일 조각들(111, 121, 141)이 서로 중복되는 부분을 포함하지 않도록, 수신되는 파일 조각들(111, 121, 141)을 결정한다. 예를 들어, 제 1 파일 조각(111)에 세그먼트(b)가 포함되는 경우, 다른 파일 조각들(121, 141)에는 세그먼트(b)가 포함되지 않도록 파일 조각들(111, 121, 141)을 스케줄링한다. 또는, 제 1 파일 조각(121)에 세그먼트(a) 중 일부(a')가 포함되는 경우, 다른 파일 조각들(111, 141)에는 세그먼트(a) 중 일부(a')를 제외한 나머지 일부(a'')만이 포함되도록 파일 조각들(111, 121, 141)을 결정한다. 위와 같은 구성에 따르면, 파일 관리 장치(2100)는 중복되는 블록을 수신하지 않으므로, 파일 관리 장치(2100)가 파일 조각들(111, 121, 141)을 수신하는 데 사용되는 통신량 및 시간이 더욱 감소될 수 있다.

도 3은 도 1에 도시된 파일 관리 장치(1100)를 구체적으로 나타내는 블록도이다. 파일 관리 장치(1100)는 통신부(1110), 컨트롤러(1120), 스케줄러(1130) 및 저장부(1140)를 포함할 수 있으나, 이에 제한되는 것은 아니다.

통신부(1110)는 파일 관리 장치(1100)와 외부 객체와의 통신을 수행하는 역할을 한다. 예를 들어, 파일 관리 장

치(1100)는 통신부(1110)를 통해 저장 장치들(1200, 도 1 참조)과 통신할 수 있다. 통신부(1110)는 외부와 네트워크 또는 통신이 가능한 모든 통신 수단을 포함할 수 있으며, 그러한 통신 수단에 수반되는 통신 인터페이스를 구비할 수 있다. 통신부(1110)가 이용하는 통신 인터페이스 또는 통신 방법은 유선 통신, 무선 통신, 3G, 4G, 또는 그 밖의 다양한 통신 인터페이스 또는 통신 방법을 포함할 수 있다.

실시 예로서, 통신부(1110)는 파일 관리 장치(1100)와 외부 객체 사이의 통신 상태 정보를 관리하는 통신 상태 관리자(1110)를 포함할 수 있다. 여기서, 통신 상태 정보는 파일 관리 장치(1100)와 외부 객체 사이의 채널 상태 또는 통신 속도를 나타내는 정보를 포함할 수 있다. 또한, 통신 상태 관리자(1111)는 단순히 통신 상태 정보를 관리 및 저장하는 것 이외에, 외부 객체와의 채널 상태 또는 통신 속도를 검출하여 새로운 통신 상태 정보를 생성하거나 기존의 통신 상태 정보를 갱신할 수도 있다.

여기서, 통신 상태 관리자(1111)는 통신부(1110)에 포함되는 것으로 설명되었으나, 이에 한정되는 것은 아니며, 통신 상태 관리자(1111)는 파일 관리 장치(1100)의 어느 위치에도 배치될 수 있고, 심지어는 파일 관리 장치(1100)와 분리된 별도의 모듈로서 배치될 수도 있다.

컨트롤러(1120)는 파일 관리 장치(1100) 및 파일 관리 장치를 구성하는 다른 모듈들(1110, 1130, 1140)의 전반적인 동작을 제어한다. 예를 들어, 컨트롤러(1120)는 n 개 저장 장치들(1200) 중 원본 파일 복원을 위한 파일 조각들을 수신할 k 개 저장 장치들을 선택할 수 있다. 그리고, 컨트롤러(1120)는 파일 관리 장치(1100)의 구동에 필요한 연산들을 수행하는 중앙 처리 장치로서 기능할 수 있다.

스케줄러(1130)는 파일 관리 장치(1100)가 저장 장치들(1200)로부터 수신할 세그먼트들 또는 파일 조각들의 목록을 생성 및 관리한다. 그리고, 생성된 파일 목록들에 따라 저장 장치들(1200)로부터 적어도 일부의 세그먼트 또는 파일 조각들이 수신될 수 있도록 필요한 정보를 컨트롤러(1120) 또는 통신부(1110)에 제공한다. 스케줄러(1130)가 파일 조각 목록을 생성 및 관리하는 구체적인 방법은 도 3 이하에서 더욱 상세히 설명될 것이다.

저장부(1140)는 파일 관리 장치(1100)에 수신된 데이터 또는 파일 관리 장치(1100)에서 처리한 데이터를 저장한다. 예를 들어, 저장부(1140)는 통신 상태 관리자(1111), 컨트롤러(1120) 또는 스케줄러(1130)의 기능을 수행하는 과정에서 얻어진 데이터들을 저장할 수 있다.

도 4 이하에서는 본 발명에 따라 수신할 파일 조각 목록들을 스케줄링하고, 스케줄링된 파일 조각 목록에 따라 파일 조각들을 수신하는 방법이 설명될 것이다.

이후의 설명을 위해, 몇 가지 용어들이 아래와 같이 정의된다.

용어 정의

n : 파일 관리 시스템(1000)이 포함하는 저장 장치의 수

k : 파일 관리 시스템(1000)이 원본 파일을 성공적으로 복원하기 위해 필요한 최소의 저장 장치의 수. 또는, 원본 파일을 성공적으로 복원하기 위해 필요한 최소의 파일 조각의 수. 단, k 는 0보다 크고 n 이하인 정수이다.

$M_1, M_2, M_3, \dots, M_n$: n 개의 저장 장치들

F : 원본 파일

F' : 원본 파일에 더미 영역을 부가한 수정 파일

F'' : 블록 단위로 파일(원본 파일 또는 수정 파일)의 순서를 뒤섞는 재배열을 수행한 재배열된 파일

s : 원본 파일(F)을 구성하는 블록의 개수

b : 파일(원본 파일 또는 수정 파일)을 구성하는 블록의 크기(byte)

e : 파일(원본 파일 또는 수정 파일)의 재배열을 수행하기 위한 재배열 키

한편, 여기서 수정 파일(F')은 원본 파일(F)의 크기가 블록 크기(b)의 정수 배가 아닌 경우, 원본 파일(F)에 더미 영역을 부가하여 전체 크기를 블록 크기(b)의 정수 배로 맞춘 파일이고, 재배열된 파일(F'')은 수정 파일(F') 또는 원본 파일(F)의 블록 순서를 재배열 키(e)를 이용하여 임의적으로 재배열한 파일이다.

$B'[i]$: 수정 파일(F')의 i 번째 블록(즉, F' 는 $B'[1] \parallel B'[2] \parallel \dots \parallel B'[s]$ 로 나타내어진다). 이때, ‘ $\parallel$ ’의 의미는 병렬 연결을 의미한다. 예를 들어, $X = A \parallel B$ 는 A 옆에 B 가 위치한 형태로 X 가 구성됨을 의미함.

$B[i]$: 재배열된 파일(F'')의 i 번째 블록 (즉, F'' 는 $B'[1] \parallel B'[2] \parallel \dots \parallel B'[s]$ 로 나타내어진다)

$S[1], S[2], \dots, S[nC_{k-1}]$: 재배열된 파일(F'')로부터 분할되는 세그먼트들(즉, F'' 는 $S[1] \parallel S[2] \parallel \dots \parallel S[nC_{k-1}]$ 로 나타내어질 수 있고, 각 세그먼트들은 복수의 블록으로 구성될 수 있다).

$f_1, f_2, \dots, f_n$: n 개의 저장 장치들($M_1, M_2, \dots, M_n$)에 각각 저장된 n 개의 파일 조각

$p_1, p_2, \dots, p_k$: n 개의 저장 장치들 중 원본 파일 복원을 위해 선택된 k 개 저장 장치들의 번호

$r_{p1}, r_{p2}, \dots, r_{pk}$: 원본 파일 복원을 위해 선택된 k 개의 저장 장치($M_1, M_2, \dots, M_k$)와 파일 관리 장치(1100) 사이의 통신 속도 비. 단, $r_{p1} \geq r_{p2} \geq \dots \geq r_{pk}$ 이고, $r_1 + r_2 + \dots + r_k = s$ 이다.

DT : Distribution Table ($n \times nC_{k-1}$ 행렬로 구성됨), $DT = \begin{pmatrix} DT_1 \\ \vdots \\ DT_n \end{pmatrix}$ 이며, $DT_i = (DT_{(i,1)}, DT_{(i,2)}, \dots, DT_{(i,x)})$ 이다(단, 여기서 x 는 nC_{k-1}).

여기서, DT 는 각각의 저장 장치가 파일(원본 파일, 수정 파일 또는 재배열된 파일)의 세그먼트 중 어떤 세그먼트를 포함하고 있는지 알려주는 테이블이다. 예를 들어, $DT_i = (d_1, d_2, \dots, d_x)$ (단, x 는 nC_{k-1})는 i 번째 저장 장치(M_i)에 저장되는 파일조각(f_i)이 $S[d_1] \parallel S[d_2] \parallel \dots \parallel S[d_x]$ (단, x 는 nC_{k-1})와 같은 세그먼트들로 구성됨을 의미한다.

$D_i: (DT_{(i,1)}, DT_{(i,2)}, \dots, DT_{(i,x)})$ (단, x 는 nC_{k-1})

PT : Possession Table ($n \times nC_{k-1}$ 행렬로 구성됨), $PT = \begin{pmatrix} PT_1 \\ \vdots \\ PT_n \end{pmatrix}$ 이며, $PT_i = (PT_{(i,1)}, PT_{(i,2)}, \dots, PT_{(i,y)})$ 이다(단, y 는 nC_{k-1}).

한편, 여기서 $PT_{(i,j)} = \begin{cases} S[j] \text{를 이루는 블록의 개수}, & j \in D_i \\ 0, & j \notin D_i \end{cases}$ 로 정의된다. 즉, 파일 조각(f_i)가 어떤 세그먼트 $S[j]$ 를 포함하고 있으면 $PT_{(i,j)}$ 의 값은 $S[j]$ 에 포함된 블록의 개수가 되고, 포함하고 있지 않으면 $PT_{(i,j)}$ 의 값은 0이 된다.

ST : Scheduling Table ($n \times nC_{k-1}$ 행렬로 구성됨), $ST = \begin{pmatrix} ST_1 \\ \vdots \\ ST_n \end{pmatrix}$ 이며, $ST_i = (ST_{(i,1)}, ST_{(i,2)}, \dots, ST_{(i,y)})$ 이다(단, y 는 nC_{k-1}).

여기서, ST 는 파일 관리 장치(1100)가 k 개의 저장 장치 각각에 대해, 저장 장치에 저장된 세그먼트 중에서 몇 개의 블록을 수신할 것인지를 결정하는 테이블이다. 예를 들어, $DT_i = (d_1, d_2, \dots, d_y)$ (단, y 는 nC_{k-1})일 때 $ST_i = (h_1, h_2, \dots, h_y)$ 는, 저장 장치(M_i)가 $S[d_1] \parallel S[d_2] \parallel \dots \parallel S[d_y]$ 을 저장하고 있고 파일 관리 장치(1100)가 저장 장치(M_i)의 세그먼트 중 $S[d_1]$ 에서 h_1 개의 블록을, $S[d_2]$ 에서 h_2 개의 블록을, ..., $S[d_y]$ 에서 h_y 개의 블록을 다 운반도록 스케줄링 테이블이 구성됨을 의미한다.

한편, ST 를 결정하기 위한 구체적인 방법에 대해서는 후술될 것이다.

$X_i = ST_{(i,1)} + ST_{(i,2)} + \dots + ST_{(i,x)}$ 이다(단, 여기서 x 는 nC_{k-1})

$A_i: j=1, 2, \dots, nC_{k-1}$ 에 대하여, $ST_{(i,j)} \neq 0$ 인 j 의 집합. 즉, $A_i = \{j \mid ST_{(i,j)} \neq 0, 1 \leq j \leq nC_{k-1}\}$

$$N = \{1, 2, 3, \dots, n\}$$

$$W = \{1, 2, 3, \dots, {}_nC_{k-1}\}$$

T: N의 원소가 $n-k+1$ 개인 모든 부분집합의 집합

$$Q_i = \{C \in T \mid i \in C\}$$

본 발명에서 제안하는 기술은 파일 관리 장치가 n 개 중 k 개의 저장 장치($M_1, M_2, \dots, M_k$)로부터 k 개의 파일 조각들을 수신하여 원본 파일을 복원되되, 수신하는 파일 조각들이 서로 중복되는 파일 블록을 포함하지 않도록 한다. 또한, 파일 관리 장치와 저장 장치 사이의 통신 속도에 따라 각각의 저장 장치로부터 수신하는 파일 조각의 크기(또는, 파일 조각에 포함된 블록들의 전체 개수)를 조정함으로써, 파일 관리 장치가 파일 조각들을 모두 수신하는 데 걸리는 시간이 최소화되게 한다.

이 경우, 일반적으로 저장 장치들의 통신 속도의 비율과 각 저장 장치로부터 수신하는 파일 블록 개수의 비율이 같을 때 파일 조각의 수신 시간이 가장 적게 걸릴 수 있다.

이하에서는, 파일 관리 장치(1100)가 각 저장 장치들(1200)로부터 수신할 파일 조각들을 결정 또는 스케줄링하는 방법이 설명된다.

이를 위해, 먼저 다음과 같은 인기도 함수 $P:W \rightarrow Z$ 가 정의된다.

$$P(w) = \sum_{j=1}^k \delta(w, p_j)$$

$$\text{단, } \delta(w, p_j) = \begin{cases} r_{p_j} - X_{p_j} & , r_{p_j} > X_{p_j}, w \in S_{p_j} \\ 0 & , \text{otherwise} \end{cases}$$

실시 예로서, 파일 관리 장치(1100)는 위에서 정의한 DT, PT 및 ST를 포함할 수 있다. 또는, 파일 관리 장치(1100)는 DT, PT, ST를 n 개의 저장 장치들($M_1, M_2, M_3, \dots, M_n$) 또는 별도의 저장 매체로부터 제공받을 수도 있다.

파일 관리 장치(1100)가 각 저장 장치들(1200)로부터 수신할 파일 조각들을 결정 또는 스케줄링하는 방법은 아래의 알고리즘에 의해 결정된다.

스케줄링 방법

입력값: 파일 조각을 제공하는 k 개 저장 장치들의 번호 $p_1, p_2, \dots, p_k$. 단, $r_{p_1} \geq r_{p_2} \geq \dots \geq r_{p_k}$ 이다.

1: ST의 원소를 모두 0으로 초기화한다.

2: W 를 W' 에 대입한다.

3: (반복문 시작) $i=1$ 부터 k 까지 반복

4: (반복문 시작) 모든 $j \in D_i \cap W'$ 에 대하여 다음을 반복

5: $ST_{(p_i, j)}$ 에 $S[j]$ 를 구성하는 블록들의 개수를 대입한다.

6: W' 에 $W' \setminus D_i$ 를 대입한다.

7: (반복문 끝)

8: (반복문 끝)

9: (반복문 시작)

10: $\{p_1, p_2, \dots, p_k\}$ 의 원소들의 순서를 재배열한 결과를 $\{q_1, q_2, \dots, q_k\}$ 로 정의하되, $\{q_1, q_2, \dots, q_k\}$ 는 아래 식을 만족하도록 정의된다.

$$X_{q1} - r_{q1} \geq X_{q2} - r_{q2} \geq \dots \geq X_{qk} - r_{qk}$$

11: (조건문 시작) 만일 $X_{q1} = r_{q1}$ 이면,

12: ST를 반환하고 프로세스를 종료한다.

13: (조건문 끝)

14: t에 k를 대입하고, g에 $X_{q1} - r_{q1}$ 을 대입한다.

15: (반복문 시작) $X_{q1} > r_{q1}$ 및 $X_{qt} > r_{qt}$ 인 동안 다음을 반복한다.

16: 집합 $A_{q1} \cap D_{qt}$ 의 원소들을 순서를 재배열한 결과를 $\{\pi_1, \pi_2, \dots, \pi_\mu\}$ 로 정의하되, $\{\pi_1, \pi_2, \dots, \pi_\mu\}$ 는 아래 식을 만족하도록 정의된다.

$$\mu = |A_{q1} \cap D_{qt}|, P(\pi_1) \leq P(\pi_2) \leq \dots \leq P(\pi_\mu)$$

17: (반복문 시작) m=1 부터 μ 까지 반복

18: v에 $\min\{ST_{(q1, \pi m)}, X_{q1} - r_{q1}, r_{qt} - X_{qt}\}$ 를 대입

19: $ST_{(q1, \pi m)}$ 에 $ST_{(q1, \pi m)} - v$ 를 대입

20: $ST_{(qt, \pi m)}$ 에 $ST_{(qt, \pi m)} + v$ 를 대입

21: (반복문 끝)

22: t에 t-1을 대입(t = t-1)

23: (반복문 끝)

24: (조건문 시작) 만일, $g = X_{q1} - r_{q1}$ 이면,

25: ST를 반환하고 프로세스를 종료

26: (조건문 끝)

27: (반복문 끝)

단, 여기서는 $r_{p1}, r_{p2}, \dots, r_{pk}$ 의 합이 원본 파일(또는, 수정 파일)에 포함되는 전체 블록들의 개수와 동일한 것으로 가정하였다.

본 발명에 따른 파일 관리 장치(예를 들어, 도 1의 1100)는 위에서 설명된 알고리즘에 따라 수신할 파일 조각들을 결정 또는 스케줄링하고, 결정 또는 스케줄링된 파일 조각들을 k개의 분산 저장 장치들(예를 들어, 도 1의 1200_1, 1200_2, ..., 1200_k, 단 k는 0보다 크고 n이하인 정수)로부터 파일 조각들을 수신한다.

위에서 설명된, 알고리즘에 따르면, 복수의 저장 장치들(1200_1, 1200_2, ..., 1200_k)로부터 수신할 파일 조각들이 표현된 스케줄링 테이블(ST)이 출력되고, 이때 스케줄링 테이블에 따라 결정되는 파일 조각들($f_1, f_2, \dots, f_k$)의 크기 비는 파일 관리 장치(1100)와 분산 저장 장치들(1200_1, 1200_2, ..., 1200_k) 사이의 통신 속도 비($r_{p1}, r_{p2}, \dots, r_{pk}$)와 같거나, 그에 근접해진다.

도 4는 본 발명의 실시 예에 따른, k=4, n=6인 경우의 재배열된 파일(F')을 예시적으로 나타내는 도면이다. 그리고, 도 5 및 도 6은 도 4에 도시된 재배열된 파일(F')에 대응되는 DT 및 PT를 나타내는 행렬도이다.

도 4의 각 칸에서 위쪽 문자(S[i])는 세그먼트의 번호 또는 명칭을 나타내는 것이고, 아래쪽 숫자는 해당 세그먼트를 구성하는 블록의 개수를 나타낸다. 예를 들어, 재배열된 파일(F')은 제 1 세그먼트(S[1])부터 제 20 세그먼트(S[20])까지 순차적으로 배열된 복수의 세그먼트들로 구성되고, 각각의 세그먼트들은 해당 칸의 아래쪽에 기재된 숫자만큼의 블록들로 구성되며(예를 들어, S[1]은 100개의 블록으로 구성됨), 재배열된 파일(F')은 총 1200개의 블록으로 구성된다.

도 5를 참조하면, 20개의 세그먼트들로 구성된 재배열된 파일(F')을 n개(여기서는 6개)의 저장 장치에 분산하기 위한 분산 테이블(Distribution Table, DT)이 도시된다. DT는 분산 파일 조각들에 포함될 세그먼트들을 나타낸다. 여기서 분산 파일 조각들은 재배열된 파일(F')의 적어도 일부를 포함하는 파일 조각들로서 대응되는 저장 장치에 저장되는 파일 조각을 의미한다. 도 5에서, DT의 각 행은 재배열된 파일(F')을 구성하는 세그먼트들 중 각각의 분산 파일 조각에 포함될 세그먼트들을 나타낸다.

예를 들어, 제 1 행의 원소가 1, 2, 3, 4, 5, 6, 7, 8, 9, 10이면, 제 1 파일 조각은 재배열된 파일(F')의 세그먼트들 중 1, 2, 3, 4, 5, 6, 7, 8, 9, 10 번째 세그먼트들(즉, S[1], S[2], S[3], S[4], S[5], S[6], S[7], S[8], S[9], S[10])을 포함하게 된다. 그리고, 제 1 파일 조각은 6개의 저장 장치들(M₁, M₂, M₃, M₄, M₅, M₆) 중 어느 하나(예를 들어, M₁)에 저장된다. 유사하게, 제 2 행의 원소가 1, 2, 3, 4, 11, 12, 13, 14, 15, 16이면, 제 2 파일 조각은 재배열된 파일(F')의 세그먼트들 중 1, 2, 3, 4, 11, 12, 13, 14, 15, 16 번째 세그먼트들(즉, S[1], S[2], S[3], S[4], S[11], S[12], S[13], S[14], S[15], S[16])을 포함하게 된다. 그리고, 제 2 파일 조각은 6개의 저장 장치들 중 제 1 파일 조각이 저장된 저장 장치가 아닌 다른 어느 하나(예를 들어, M₂)에 저장된다. 위와 같이, 6개의 저장 장치들(M₁, M₂, M₃, M₄, M₅, M₆)은 재배열된 파일(F')의 세그먼트들을 나누어 저장하고, 그 중 임의의 4개의 저장 장치가 재배열된 파일(F')의 모든 세그먼트들을 포함하도록 저장된다.

즉, DT는 복수의 저장 장치에 분산 저장되는 파일 조각들의 구성을 나타내며, 각 저장 장치에 원본 파일(또는, 재배열된 파일)의 어떠한 세그먼트가 저장되었는지를 알기 위해 참조될 수 있다.

도 6을 참조하면, 도 5에 도시된 DT에 따른 점유 테이블(Possession Table, PT)이 도시된다. PT는 6개의 저장 장치들(M₁, M₂, M₃, M₄, M₅, M₆)에 저장되는 파일 조각이 포함하는 세그먼트와, 세그먼트들에 포함된 블록의 수를 나타낸다. 구체적으로, PT의 각 행은 각 파일 조각의 구성을 나타내며, 각 행의 0이 아닌 원소는 대응하는 세그먼트를 각 파일 조각이 포함함을 의미하며, 0인 원소는 대응하는 세그먼트를 각 파일 조각이 포함하지 않음을 의미한다. 한편, 각 원소가 나타내는 숫자는 대응하는 세그먼트에 포함된 블록을 의미한다. 예를 들면, 제 1 행의 원소들은 제 1 파일 조각의 구성을 나타내고, 제 1 행 1 열의 원소가 100이라는 것은 제 1 파일 조각이 제 1 세그먼트를 포함하는 동시에, 제 1 세그먼트가 100개의 블록을 포함함을 의미한다. 반면에, 제 1 행 20 열의 원소가 0이라는 것은 제 1 파일 조각이 제 20 세그먼트를 포함하지 않음을 의미한다.

도 7은 도 4 내지 도 6의 실시 예에 있어서, 각 저장 장치들에 저장되는 파일 조각들을 나타내는 도면이다.

도 7을 참조하면, 6개의 파일 조각들(f₁, f₂, f₃, f₄, f₅, f₆)이 6개의 저장 장치들(M₁, M₂, M₃, M₄, M₅, M₆)에 나누어 저장되며, 각 파일 조각들은 10개의 세그먼트들을 포함한다. 각 파일 조각들의 구분된 칸(또는 박스) 속에 기재된 문자는 각 파일 조각들이 포함하는 세그먼트의 번호 또는 명칭을 나타낸다. 예를 들어, 제 1 조각 파일(f₁)은 제 1 세그먼트(S[1]) 내지 제 10 세그먼트(S[10])를 포함한다.

이하에서는, 파일 관리 시스템이 6개의 저장 장치들(M₁, M₂, M₃, M₄, M₅, M₆)에 저장된 파일 조각들로부터 원본 파일을 복원하는 방법에 대해서 앞서 설명된 알고리즘을 참조하여 설명한다.

먼저, 파일 관리 장치는 6개의 저장 장치들(M₁, M₂, M₃, M₄, M₅, M₆) 중 파일 조각들을 수신할 임의의 4개의 저장 장치를 결정한다. 여기서는, 제 1, 제 2, 제 4, 및 제 6 저장 장치(M₁, M₂, M₄, M₆)가 선택된 것으로 가정한다. 또한, 파일 관리 장치와 선택된 4개의 저장 장치(M₁, M₂, M₄, M₆)의 각각의 통신 속도 비는 5:9:2:4라고 가정한다. 파일 관리 장치와의 통신 속도가 M₁, M₂, M₆, M₄의 순서대로 빠르므로, 위에서 알고리즘을 참조하여 설명된 스케줄링 방법(이하 '스케줄링 방법'이라 한다)의 제 10 단계에 의해 (p₁, p₂, p₃, p₄)=(2, 1, 6, 4)이 된다. 그리고, 재배열된 파일(F')을 이루는 파일 블록의 전체 개수가 1200이므로 앞서 용어 정의에서 설명된 내용에 따라 (r_{p1}, r_{p2}, r_{p3}, r_{p4})=(r₂, r₁, r₆, r₄)=(540, 300, 240, 120)이 된다. 즉, 파일 관리 장치는 4개의 저장 장치 M₁, M₂, M₆, M₄로부터 각각 서로 중복되지 않는 300개, 540개, 120개, 240개의 블록을 받아오도록 스케줄링 테이블(Scheduling Table, ST)을 결정함으로써, 최소의 시간 및 최소의 통신량만으로 원본 파일을 복원할 수 있다.

다음으로, 본 발명의 스케줄링 방법에 따라 스케줄링 테이블(ST)을 결정하는 방법이 도 8 내지 도 10의 행렬도를 참조하여 설명된다.

본 발명의 스케줄링 방법에서는 먼저 ST를 초기화한다. ST의 초기화는 스케줄링 방법의 초기화 반복문(스케줄링 방법의 3 단계 내지 8 단계)에 의해 수행된다. 이때, 저장 장치들의 통신 속도들에 따라, 할당 우선 순위가 결정된다. 예를 들어, 통신 속도 빠른 순서는 $(p_1, p_2, p_3, p_4) = (2, 1, 6, 4)$ 이므로, ST의 2번째, 1번째, 6번째, 4번째 행의 순서대로 우선 순위가 결정된다. 그리고, 결정된 우선 순위에 따라 ST의 각 행이 중복되는 않는 범위에서 순차적으로 채워진다. 여기서 중복되지 않는 범위에서 채워진다는 것은 ST의 각 행이 서로 중복되는 세그먼트를 포함하지 않도록 배타적으로 채워진다는 것을 말하며, 행렬상으로는 하나의 열에 대해서 단 하나의 행만이(다른 행에 대해서 배타적으로) 0이 아닌 값을 갖는 것을 의미한다.

구체적으로, 그리고, ST의 두 번째 행에는 D_{p_1} (즉, D_2)로부터 수신할 블록들의 개수가 저장된다. 이때, 제 2 행은 먼저 채워지는 행이고 $W' = \{1, 2, 3, 4, \dots, 20\}$, $D_2 = \{1, 2, 3, 4, 11, 12, 13, 14, 15, 16\} \subset W'$ 이므로 ($D_2 \cap W' = D_2$), 제 2 행의 1, 2, 3, 4, 11, 12, 13, 14, 15 및 16번째 원소에는 각각 $S[1]$, $S[2]$, $S[3]$, $S[4]$, $S[11]$, $S[12]$, $S[13]$, $S[14]$, $S[15]$, $S[16]$ 의 블록의 개수가 온전하게 채워진다.

다음으로, 제 2 행과 다른 행들의 중복을 피하기 위해 집합 W' 는 D_2 를 뺀 집합인 $\{5, 6, 7, 8, 9, 10, 17, 18, 19, 20\}$ 으로 업데이트되며, 이어서 제 1 행이 채워진다. 이때, $D_1 \cap W' = \{5, 6, 7, 8, 9, 10\}$ 이므로, 제 1 행의 5, 6, 7, 8, 9 및 10번째 원소에만 각각 $S[5]$, $S[6]$, $S[7]$, $S[8]$, $S[9]$, $S[10]$ 의 블록의 개수가 채워진다.

그리고, 집합 W' 는 행들의 중복을 피하기 위해 다시 D_1 을 뺀 집합인 $\{17, 18, 19, 20\}$ 으로 업데이트된다. 이어서, ST의 제 6 행이 채워진다. 이때, $W' = \{17, 18, 19, 20\}$, $D_6 \cap W' = \{18, 19, 20\}$ 이므로, 제 6 행의 18, 19 및 20 번째 원소에만 각각 $S[18]$, $S[19]$, $S[20]$ 의 블록의 개수가 채워진다.

그리고 집합 W' 는 행들의 중복을 피하기 위해 또 다시 D_6 를 뺀 집합인 $\{17\}$ 로 업데이트된다. 마찬가지로 방법으로 ST의 제 4 행이 채워지며, 제 4 행의 17번째 원소에만 $S[17]$ 의 블록의 개수가 채워지고 나머지는 0이 채워진다.

위와 같은 과정에 따라 ST의 초기화가 완료되면, 스케줄링 방법의 제 9 단계 내지 제 27 단계의 반복문(이하, '주 반복문'이라 함)이 시작된다.

i) 주 반복문의 제 1 루프

우선, 초기화된 ST에 따라 X_i 가 다음과 같이 결정되고,

$$X_1=425, X_2=659, X_4=35, X_6=81$$

이어서, 어떤 파일 조각들에 블록들이 과다하게 또는 과소하게 할당되었는지 판단한다. 이때, $(r_2, r_1, r_6, r_4) = (540, 300, 240, 120)$ 이므로, $X_1 - r_1 > X_2 - r_2 > 0 > X_4 - r_4 > X_6 - r_6$ 이다. 따라서, $(q_1, q_2, q_3, q_4) = (1, 2, 4, 6)$ 이 되고, 이는 제 1 파일 조각, 제 2 파일 조각, 제 4 파일 조각 및 제 6 파일 조각 순으로 블록들이 과다하게(또는, 역순으로 블록들이 과소하게) 할당되었음을 의미한다. 이때, $X_1 \neq r_1$ 이므로 스케줄링 방법의 제 11 단계 내지 제 13 단계의 조건문은 만족하지 않고, 이는 블록들이 과부족되게 할당된 파일 조각들이 존재함을 의미한다. 따라서, 본 발명의 스케줄링 방법은 계속 진행하고, 제 14 단계의 과정에 의해 $t=4$, $g=X_1-r_1=125$ 가 설정된다.

a) $t=4$ 일 때,

본 발명의 스케줄링 방법은 대응하는 저장 장치의 속도에 비해 상대적으로 더 많은 블록이 할당된 파일 조각의 블록들을 다른 파일 조각으로 재할당한다. 실시 예로서, 재할당하는 파일 조각은 대응하는 저장 장치의 속도에 비해 상대적으로 블록이 가장 과다하게 할당된 파일 조각(여기서는, 제 1 파일 조각)일 수 있다. 실시 예로서, 재할당받는 파일 조각은 대응하는 저장 장치의 속도에 비해 상대적으로 블록이 가장 과소하게 할당된 파일 조각(여기서는, 제 6 파일 조각)일 수 있다.

재할당을 위해 아래와 같이 인기도 함수를 이용할 수 있다.

$$A_1 = \{5, 6, 7, 8, 9, 10\}, D_6 = \{4, 7, 9, 10, 13, 15, 16, 18, 19, 20\} \text{이므로 } A_1 \cap D_6 = \{7, 9, 10\}.$$

$$\begin{aligned} P(7) &= \delta(7,1) + \delta(7,2) + \delta(7,4) + \delta(7,6) = 0 + 0 + 0 + (240 - 81) = 159 \\ P(9) &= \delta(9,1) + \delta(9,2) + \delta(9,4) + \delta(9,6) = 0 + 0 + (120 - 35) + (240 - 81) = 244 \\ P(10) &= \delta(10,1) + \delta(10,2) + \delta(10,4) + \delta(10,6) = 0 + 0 + 0 + (240 - 81) = 159 \end{aligned}$$

위의 인기도 함수의 크기에 따라, (π_1, π_2, π_3) 는 각각 (7, 10, 9)로 결정된다. 실시 예로서, 스케줄링 방법은 인기도 함수가 작은 순서로 블록들을 재할당할 수 있다. 재할당 단계는 스케줄링 방법의 제 17 단계 내지 제 22 단계의 반복문을 이용하여 수행된다.

먼저, $m=1$ 일때, 인기도 함수가 가장 낮은 제 7 열의 블록들(즉, 7 번째 세그먼트의 블록들)을 대상으로 재할당이 수행된다. 이때, $X_1-r_1=125$, $r_6-X_6=159$, $ST_{(1,7)}=71$ 이므로 $v=71$ 이다. 제 1 파일 조각의 7 번째 세그먼트의 블록들이 제 1 파일 조각의 과다 블록 및 제 6 파일 조각의 과소 블록 수보다 적으므로, 제 1 파일 조각의 7 번째 세그먼트의 블록들은 전체가 제 6 파일 조각으로 재할당되고, $ST_{(1,7)}=71-71=0$, $ST_{(6,7)}=0+71=71$ 이 된다. 그리고, 재할당 결과에 따라 X_1 , X_6 이 각각 $X_1=354$, $X_6=152$ 로 바뀐다.

그리고, $m=2$ 일때, 인기도 함수가 다음으로 낮은 제 10 열의 블록들(즉, 10 번째 세그먼트의 블록들)을 대상으로 재할당이 수행된다. 이때, $X_1-r_1=54$, $r_6-X_6=88$, $ST_{(1,10)}=61$ 이므로 $v=54$ 이고, 제 1 파일 조각의 10 번째 세그먼트의 블록들은 제 1 파일 조각의 과다 블록의 수보다 더 많다. 따라서, 재할당에 의해 제 1 파일 조각의 블록들이 과소하게 되는 것을 피하기 위해, 제 1 파일 조각의 10 번째 세그먼트의 블록들은 일부만이(예를 들어, 제 1 파일 조각의 과다 블록의 수만큼) 제 6 파일 조각으로 재할당된다. 따라서, $ST_{(1,10)}=61-54=7$, $ST_{(6,10)}=0+54=54$ 와 같이 ST상에서 파일 블록들이 재할당 되고, 재할당 결과에 따라, X_1 , X_6 이 각각 $X_1=300$, $X_6=206$ 로 바뀐다.

다음으로, $m=3$ 일때, $X_1-r_1=0$ 이므로 $v=0$ 이다. 따라서, 재할당은 수행되지 않고 ST는 변하지 않는다.

b) $t=3$ 일 때,

이때는, 제 1 파일 조각의 과다 블록을 차순위의 과소 파일 조각(제 10 단계에서 두 번째로 과소하게 블록들이 할당된 것으로 판단된 파일 조각, 여기서는 제 4 파일 조각)에 재할당한다. 그러나, $X_1=r_1$ 이므로 제 1 파일 조각은 과다 블록이 존재하지 않고, 제 15 단계의 조건을 만족하지 않는다. 따라서, 제 15 단계 내지 제 23 단계의 반복문(이하, '부 반복문'이라 함)은 종료된다.

제 24 단계 내지 제 26 단계의 조건문은 주 반복문(제 9 단계 내지 제 27 단계)의 종료 조건을 지정한 것으로서, 부 반복문을 수행하여도 ST가 변화하지 않으면 더 이상의 재할당이 필요하지 않은 것으로 보고 ST 결정을 완료하도록 한 것이다.

한편, 주 반복문의 제 1 루프가 수행된 결과, ST는 도 9와 같이 바뀌게 되고, $X_1=300$, $X_2=659$, $X_4=35$, $X_6=206$ 이 된다.

그리고, 주 반복문의 제 2 루프가 수행된다.

ii) 주 반복문의 제 2 루프

주 반복문의 제 1 루프에 의해 ST가 변화하고 파일 조각의 블록들이 재할당되었으므로, 스케줄링 방법은 파일 조각들이 블록들이 과다 또는 과소하게 분포되었는지 여부를 다시 판단한다. 이때, $(r_2, r_1, r_6, r_4)=(540, 300, 240, 120)$ 이므로, $X_2-r_2 > X_1-r_1 = 0 > X_6-r_6 > X_4-r_4$ 이다. 따라서, $(q_1, q_2, q_3, q_4)=(2, 1, 6, 4)$ 가 된다. 이때, $X_2 \neq r_2$ 이므로 스케줄링 방법의 제 11 단계 내지 제 13 단계의 조건문을 만족하지 않고, 이는 블록들이 과부족되게 할당된 파일 조각들이 존재함을 의미한다. 따라서, 본 발명의 스케줄링 방법은 계속 진행하고, 제 14 단계의 과정에 의해 $t=4$, $g=X_2-r_2=119$ 가 설정된다.

a) $t=4$ 일 때

제 2 루프에서도 앞서의 제 1 루프와 동일하게, 인기도 함수를 이용하여 블록들의 재할당을 수행한다.

$$A_2 = \{1,2,3,4,11,12,13,14,15,16\}, D_4 = \{2,5,8,9,11,14,15,17,18,20\} \text{이므로 } A_2 \cap D_4 = \{2,11,14,15\}.$$

$$P(2) = \delta(2,1) + \delta(2,2) + \delta(2,4) + \delta(2,6) = 0 + 0 + (120 - 35) + 0 = 85$$

$$P(11) = \delta(11,1) + \delta(11,2) + \delta(11,4) + \delta(11,6) = 0 + 0 + (120 - 35) + 0 = 85$$

$$P(14) = \delta(14,1) + \delta(14,2) + \delta(14,4) + \delta(14,6) = 0 + 0 + (120 - 35) + 0 = 85$$

$$P(15) = \delta(15,1) + \delta(15,2) + \delta(15,4) + \delta(15,6) = 0 + 0 + (120 - 35) + (240 - 206) = 119$$

따라서, $(\pi_1, \pi_2, \pi_3, \pi_4) = (2, 11, 14, 15)$

먼저, $m=1$ 일때, $X_2-r_2=119$, $r_4-X_4=85$, $ST_{(2,2)}=96$ 이므로 $v=85$. 따라서, 재할당될 블록들의 수는 제 4 파일 조각의 과소 블록의 수에 의해 제한되고, 제 2 파일 조각의 2 번째 세그먼트의 블록들은 85개만이 제 4 파일 조각에 재할당된다. 그 결과, $ST_{(2,2)}=96-85=11$, $ST_{(4,2)}=0+85=85$ 가 되고, X_2 , X_4 는 각각 $X_2=574$, $X_4=120$ 로 바뀐다.

그리고, $m=2, 3, 4$ 일때, $r_4-X_4=0$ 이 되므로 $v=0$ 이다. 따라서, 재할당은 수행되지 않고 ST는 변하지 않는다.

b) $t=3$ 일 때,

여기서는, 제 2 파일 조각의 과다 블록들을 차순위 과소 파일 조각(여기서는, 제 6 파일 조각)으로 재할당한다. 유사하게, 인기도 함수를 구하면 아래와 같다.

$$A_2 = \{1,2,3,4,11,12,13,14,15,16\}, D_6 = \{4,7,9,10,13,15,16,18,19,20\} \text{이므로 } A_2 \cap D_6 = \{4,13,15,16\}.$$

$$P(4) = \delta(4,1) + \delta(4,2) + \delta(4,4) + \delta(4,6) = 0 + 0 + 0 + (240 - 206) = 34$$

$$P(13) = \delta(13,1) + \delta(13,2) + \delta(13,4) + \delta(13,6) = 0 + 0 + 0 + (240 - 206) = 34$$

$$P(15) = \delta(15,1) + \delta(15,2) + \delta(15,4) + \delta(15,6) = 0 + 0 + 0 + (240 - 206) = 34$$

$$P(16) = \delta(16,1) + \delta(16,2) + \delta(16,4) + \delta(16,6) = 0 + 0 + 0 + (240 - 206) = 34$$

따라서, $(\pi_1, \pi_2, \pi_3, \pi_4) = (4, 13, 15, 16)$

먼저, $m=1$ 일때, $X_2-r_2=34$, $r_6-X_6=34$, $ST_{(2,4)}=88$ 이므로 $v=34$. 따라서, 재할당될 블록들의 수는 제 2 파일 조각의 과다 블록의 수 또는 제 6 파일 조각의 과소 블록의 수에 의해 제한되고, 제 2 파일 조각의 4 번째 세그먼트의 블록들은 34개만이 제 6 파일 조각에 재할당된다. 그 결과, $ST_{(2,4)}=88-34=54$, $ST_{(6,4)}=0+34=34$ 가 되고, X_2 , X_6 는 각각 $X_2=540$, $X_6=240$ 이 된다.

그리고, $m=2, 3, 4$ 일때, $X_2-r_2=r_6-X_6=0$ 이 되므로 $v=0$ 이다. 따라서, 재할당은 수행되지 않고 ST는 변하지 않는다.

제 2 루프의 부 반복문에 의해 재할당이 ST가 변화하였으므로, 제 24 단계의 조건을 만족하지 않는다. 따라서, 제 24 단계 내지 제 26 단계의 조건문은 통과된다.

한편, 주 반복문의 제 2 루프가 수행된 결과, ST는 도 10과 같이 바뀌게 되고, $X_1=300$, $X_2=540$, $X_4=120$, $X_6=240$ 이 된다.

그리고, 주 반복문의 제 3 루프가 수행된다.

iii) 주 반복문의 제 3 루프

마찬가지로, 스케줄링 방법은 파일 조각들이 블록들이 과다 또는 과소하게 분포되었는지 여부를 다시 판단한다.

이때, $(r_2, r_1, r_6, r_4) = (540, 300, 240, 120)$ 이므로, $X_1-r_1 = X_2-r_2 = X_4-r_4 = X_6-r_6$ 이다. 이때, $X_1=r_1$ 이므로 제 11 단계의 조건이 충족되고, 이는 블록들의 과부족 할당 상태가 해소되었음을 의미한다. 따라서, 제 11 단계 내지 제 13 단계의 조건문이 수행되어, ST를 출력하고 주 반복문을 종료한다.

상기와 같은 본 발명의 구성에 따르면, 파일 관리 장치(1100)는 저장 장치와의 통신 속도를 고려하여 수신할 파일 조각들을 스케줄링한다. 그리고, 스케줄링한 결과에 따라 파일 조각들을 수신하여 원본 파일을 복원한다. 그 결과, 통신 속도가 빠른 저장 장치로부터 더 많은 블록(또는, 더 큰 파일 조각)을 수신하므로, 파일 관리 장치(1100)가 원본 파일을 복원하는 데 걸리는 시간이 최소화될 수 있다.

또한, 파일 관리 장치(1100)는 저장 장치들로부터 수신하는 파일 조각들이 서로 중복되는 블록을 포함하지 않도록

록 수신할 파일 조각들을 스케줄링한다. 따라서, 파일 관리 장치(1100)가 원본 파일을 복원하는데 사용되는 통신량이 최소화될 수 있다.

도 11은 본 발명의 실시 예에 따른, 파일 관리 방법을 나타내는 순서도이다. 도 11을 참조하면, 파일 관리 방법은 S110 단계 내지 S130 단계를 포함한다.

S110 단계에서, 파일 관리 장치(1100)는 파일 관리 장치(1100)와 저장 장치들 사이의 통신 속도를 측정한다. 실시 예로서, 통신 속도를 측정하는 대신, 파일 관리 장치(1100)는 미리 측정된 통신 속도를 저장 장치들 또는 다른 저장 매체로부터 제공받을 수 있다.

S120 단계에서, 파일 관리 장치(1100)는 파일 관리 장치(1100)와 저장 장치들 사이의 통신 속도에 따라, 파일 관리 장치(1100)가 어떤 저장 장치로부터 어떤 세그먼트의 블록들을 받아들일지 여부를 결정한다. 구체적으로, 파일 관리 장치(1100)는 측정된 또는 제공된 통신 속도를 참조하여, 각 저장 장치로부터 수신할 파일 조각들을 스케줄링한다. 스케줄링된 파일 조각들 각각은 대응하는 저장 장치에 저장된 원본 파일의 파일 조각 중 적어도 일부를 포함한다. 이때, 파일 조각들은 대응하는 저장 장치의 통신 속도에 비례하여 더 많은 블록들을 포함하도록(또는, 더 큰 파일 크기를 갖도록) 스케줄링된다. 또한, 파일 조각들은 서로 중복되는 블록을 포함하지 않도록 스케줄링된다.

파일 관리 장치(1100)가 위와 같은 구성을 갖도록 파일 조각들을 스케줄링하는 구체적인 방법은 도 4 내지 도 10과 관련하여 자세히 설명되었으므로, 여기서는 그에 대한 설명은 생략한다.

S130 단계에서, 파일 관리 장치(1100)는 스케줄링된 결과에 따라, 각 저장 장치들로부터 파일 조각들을 수신한다. 이때, 각각의 파일 조각들은 원본 파일의 블록들 중 적어도 일부를 포함하며, 하나의 파일 조각에 포함된 블록들은 원본 파일의 특정 세그먼트의 전부 또는 일부를 구성할 수 있다.

실시 예로서, 각 파일 조각은 자신이 포함하는 적어도 일부의 세그먼트들의 주소 또는 크기 정보를 포함할 수 있다.

이어서, 파일 관리 장치(1100)는 수신한 파일 조각들로부터 원본 파일을 복원한다. 이때, 파일 관리 장치(1100)가 원본 파일을 복원하는 순서는 원본 파일을 파일 조각들로 분산하는 순서의 역순일 수 있다.

예를 들어, 파일 관리 장치(1100)는 세그먼트들의 주소 정보 또는 크기 정보를 참조하여 수신한 파일 조각들을 세그먼트 단위로 분리한다. 그리고, 파일 관리 장치(1100)는 분리된 세그먼트 또는 세그먼트 조각들을 순서대로 정렬하여 재배열된 파일(F'')을 복원한다. 이어서, 파일 관리 장치(1100)는 재배열키(e)를 이용한 재배열 방법을 통해 재배열된 파일(F'')로부터 수정 파일(F')을 복원한다. 이때, 재배열 방법은 재배열된 파일(F'')을 블록 단위로 재배열하여 수정 파일(F')을 복원할 수 있다. 여기서 사용된 재배열 방법은 당해 기술 분야에 널리 알려진 랜덤화 방법을 이용할 수 있다.

도 12는 본 발명의 실시 예에 따른, 파일 관리 장치가 스케줄링 테이블(ST)을 참조하여 저장 장치들로부터 파일 조각들을 전송받아 원본 파일을 복원하는 방법을 나타내는 개념도이다.

도 12를 참조하면, 파일 관리 장치(3100)는 원본 파일을 복원하기 위해 ST(Scheduling Table, 300)를 참조하여, 4개의 저장 장치들로부터 수신할 파일 조각들(211, 221, 223)을 결정(또는 스케줄링)한다. 이때, ST(300)는 각 파일 조각들(211, 221, 223)이 서로 중복되는 블록을 포함하지 않도록, 그리고 각 파일 조각들(211, 221, 223)이 포함하는 블록들의 수가 대응하는 저장 장치의 통신 속도에 비례하도록 결정된다.

실시 예로서, ST(300)의 제 i 행은 제 i 파일 조각의 세그먼트 구성(또는 블록 구성)을 나타낸다. 예를 들어, ST(300)의 제 1 행이 (800, 800, 550, 0, 0, 0)이면, 제 1 파일 조각(211)이 제 1 세그먼트(a) 중 800개의 블록, 제 2 세그먼트(b) 중 800개의 블록, 및 제 3 세그먼트(c) 중 550개의 블록을 포함함을 나타낸다. 이에 따르면, 제 1 파일 조각(211)은 제 1 세그먼트(a) 및 제 3 세그먼트(c)의 일부 블록들(a' , c')을 포함하고, 제 2 세그먼트(b)의 모든 블록들(b)을 포함하는 반면에, 제 4 내지 제 6 세그먼트(d, e, f)의 블록들은 전혀 포함하지 않는다.

이와 같이 결정된 파일 조각들(211, 221, 223)에는 원본 파일(200)의 모든 세그먼트들 및 모든 블록들이 포함되므로, 파일 관리 장치(3100)는 파일 조각들(211, 221, 231)을 수신하여 원본 파일을 온전히 복원해 낼 수 있다. 또한, 각 파일 조각들(211, 221, 231)은 서로 중복되는 블록을 포함하지 않으므로, 원본 파일의 복원에 사용되는 통신량이 최소화될 수 있다. 또한, 각 파일 조각들(211, 221, 231)의 크기(또는, 각 파일 조각들에 포함된 블록들의 수)는 대응하는 저장 장치의 통신 속도에 비례하므로, 원본 파일을 복원하는데 소요되는 시간이 최소

화될 수 있다.

본 발명의 상세한 설명에서는 구체적인 실시 예를 들어 설명하였으나, 본 발명의 범위에서 벗어나지 않는 한 각 실시 예는 여러 가지 형태로 변형될 수 있다.

또한, 여기서 특정한 용어들이 사용되었으나, 이는 단지 본 발명을 설명하기 위한 목적에서 사용된 것이지 의미 한정이나 특허청구범위에 기재된 본 발명의 범위를 제한하기 위하여 사용된 것은 아니다. 그러므로 본 발명의 범위는 상술한 실시 예에 국한되어 정해져서는 안되며 후술하는 특허 청구범위뿐만 아니라 이 발명의 특허 청구 범위와 균등한 것들에 의해 정해져야 한다.

부호의 설명

1000, 2000, 3000: 파일 관리 시스템

1100, 2100, 3100: 파일 관리 장치

1200: 저장 장치들

2201, 2202, 2203, 2204: 저장 장치

1110: 통신부

1111: 통신 상태 관리자

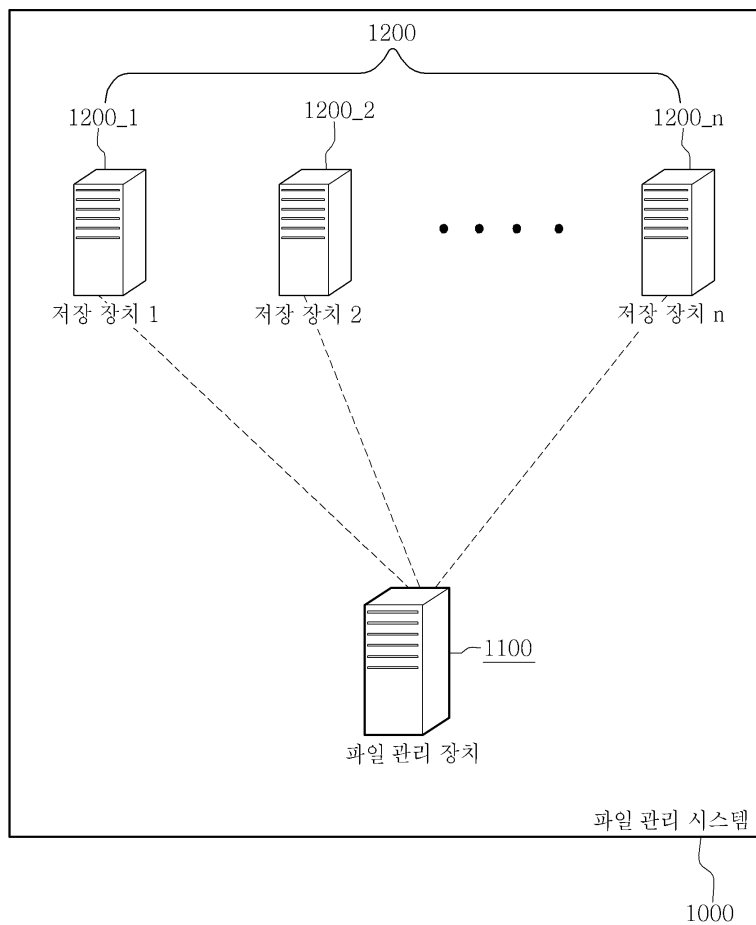
1120: 컨트롤러

1130: 스케줄러

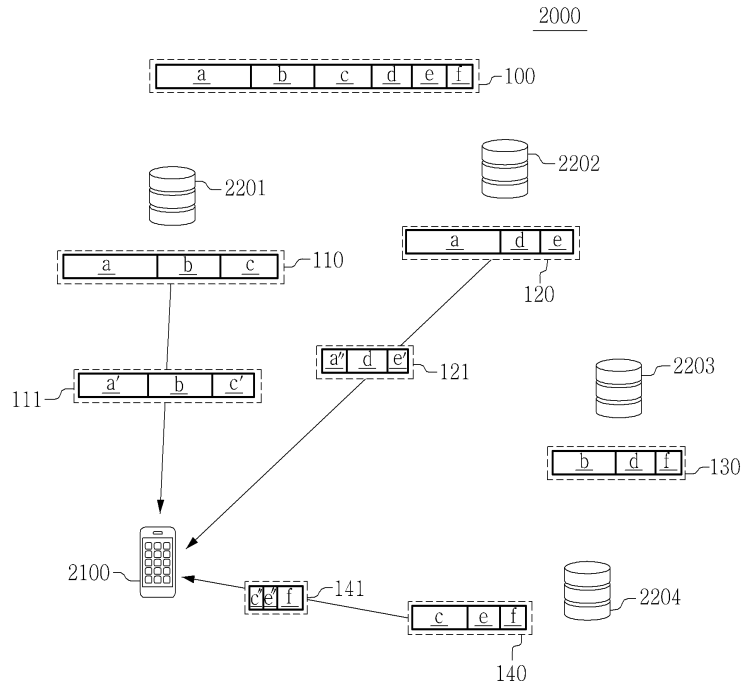
1140: 저장부

도면

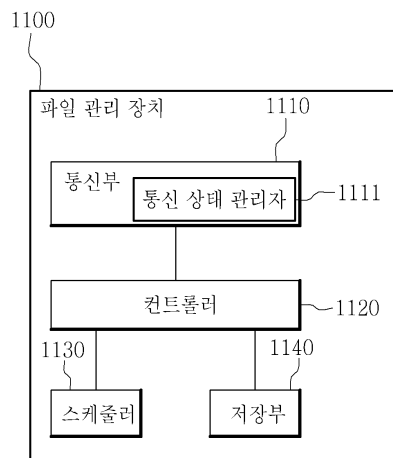
도면1



도면2



도면3



도면4

$$\mathcal{F}'' = \begin{array}{|c|} \hline S[1] & S[2] & S[3] & S[4] & S[5] & S[6] & S[7] & S[8] & S[9] & S[10] & S[11] & S[12] & S[13] & S[14] & S[15] & S[16] & S[17] & S[18] & S[19] & S[20] \\ \hline 100 & 96 & 92 & 88 & 82 & 78 & 71 & 68 & 65 & 61 & 56 & 53 & 48 & 46 & 41 & 39 & 35 & 30 & 27 & 24 \\ \hline \end{array}$$

도면5

$$DT = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 6 & 7 & 8 & 9 & 10 \\ 1 & 2 & 3 & 4 & 11 & 12 & 13 & 14 & 15 & 16 \\ 1 & 5 & 6 & 7 & 11 & 12 & 13 & 17 & 18 & 19 \\ 2 & 5 & 8 & 9 & 11 & 14 & 15 & 17 & 18 & 20 \\ 3 & 6 & 8 & 10 & 12 & 14 & 16 & 17 & 19 & 20 \\ 4 & 7 & 9 & 10 & 13 & 15 & 16 & 18 & 19 & 20 \end{pmatrix}$$

도면6

$$PT = \begin{pmatrix} 100 & 96 & 92 & 88 & 82 & 78 & 71 & 68 & 65 & 61 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 100 & 96 & 92 & 88 & 0 & 0 & 0 & 0 & 0 & 0 & 56 & 53 & 48 & 46 & 41 & 39 & 0 & 0 & 0 \\ 100 & 0 & 0 & 0 & 82 & 78 & 71 & 0 & 0 & 0 & 56 & 53 & 48 & 0 & 0 & 0 & 35 & 30 & 27 \\ 0 & 96 & 0 & 0 & 82 & 0 & 0 & 68 & 65 & 0 & 56 & 0 & 0 & 46 & 41 & 0 & 35 & 30 & 0 \\ 0 & 0 & 92 & 0 & 0 & 78 & 0 & 68 & 0 & 61 & 0 & 53 & 0 & 46 & 0 & 39 & 35 & 0 & 27 \\ 0 & 0 & 0 & 88 & 0 & 0 & 71 & 0 & 65 & 61 & 0 & 0 & 48 & 0 & 41 & 39 & 0 & 30 & 27 \end{pmatrix}$$

도면7

f_1	S[1]	S[2]	S[3]	S[4]	S[5]	S[6]	S[7]	S[8]	S[9]	S[10]
f_2	S[1]	S[2]	S[3]	S[4]	S[11]	S[12]	S[13]	S[14]	S[15]	S[16]
f_3	S[1]	S[5]	S[6]	S[7]	S[11]	S[12]	S[13]	S[17]	S[18]	S[19]
f_4	S[2]	S[5]	S[8]	S[9]	S[11]	S[14]	S[15]	S[17]	S[18]	S[20]
f_5	S[3]	S[6]	S[8]	S[10]	S[12]	S[14]	S[16]	S[17]	S[19]	S[20]
f_6	S[4]	S[7]	S[9]	S[10]	S[13]	S[15]	S[16]	S[18]	S[19]	S[20]

도면8

$$ST = \begin{pmatrix} 0 & 0 & 0 & 0 & 82 & 78 & 71 & 68 & 65 & 61 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 100 & 96 & 92 & 88 & 0 & 0 & 0 & 0 & 0 & 0 & 56 & 53 & 48 & 46 & 41 & 39 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 35 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 30 & 27 \end{pmatrix}$$

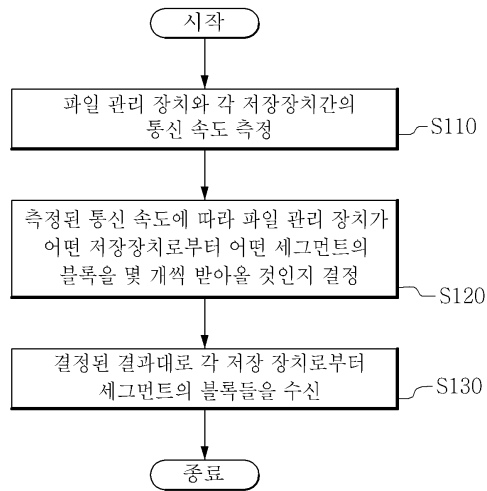
도면9

$$ST = \begin{pmatrix} 0 & 0 & 0 & 0 & 82 & 78 & 0 & 68 & 65 & 7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 100 & 96 & 92 & 88 & 0 & 0 & 0 & 0 & 0 & 0 & 56 & 53 & 48 & 46 & 41 & 39 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 35 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 71 & 0 & 0 & 54 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 30 & 27 \end{pmatrix}$$

도면10

$$ST = \begin{pmatrix} 0 & 0 & 0 & 0 & 82 & 78 & 0 & 68 & 65 & 7 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 100 & 11 & 92 & 54 & 0 & 0 & 0 & 0 & 0 & 0 & 56 & 53 & 48 & 46 & 41 & 39 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 85 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 35 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 34 & 0 & 0 & 71 & 0 & 0 & 54 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 30 & 27 \end{pmatrix}$$

도면11



도면12

