

# 플래시 메모리를 위한 페이지 비율 분석 기반의 적응적 가비지 컬렉션 정책

## (Adaptive Garbage Collection Policy based on Analysis of Page Ratio for Flash Memory)

이 승 환 †      이 태 훈 †      정 기 동 ††  
(Lee Soung Hwan)      (Lee Tae Hoon)      (Chung Ki Dong)

**요약** 플래시 메모리는 부피가 작고, 가볍고, 소비전력이 낮으며 입출력이 빨라 최근 소형기기의 저장 장치로 널리 사용이 되고 있다. 그러나 플래시 메모리는 지움 연산을 수반하는 가비지 컬렉션을 수행해야 한다. 지움 연산은 속도가 느리고, 각 블록마다 지움 연산 횟수가 제한이 있다. 따라서 본 논문에서는 지움 연산 횟수와 각 블록의 지움 횟수 편차를 줄이는데 초점을 맞춘 균등화 정책을 제안한다. 따라서 플래시 메모리의 페이지 사용률에 기반을 둔 두 가지 가비지 컬렉션 수행 모드를 정의하고, 그리고 각 모드에 대해 다른 지움 비용을 계산하여 전체 지움 연산 횟수와 각 블록의 지움 횟수 편차를 최소화하는 가비지 컬렉션 기법을 제안한다. 추가로 가비지 컬렉션 연산 시간을 최소화하기 위해 그룹 관리 기법을 제안해 보다 빠른 수행 시간을 가질 수 있도록 한다. 실험 결과 제안하는 정책은 기존의 Greedy 와 CAT 기법의 장점들을 동시에 나타내었고, 지움 횟수 편차를 평균 85% 감소 시켰고 가비지 컬렉션 수행 시간을 최대 6% 단축 시켰다.

**키워드** : 플래시 메모리, 가비지 컬렉션

**Abstract** NAND flash memory is widely used in embedded systems because of many attractive features, such as small size, light weight, low power consumption and fast access speed. However, it requires garbage collection, which includes erase operations. Erase operation is slower than other operations. Further, a block has a limited erase lifetime (typically 100,000) after which a block becomes unusable. The proposed garbage collection policy focuses on minimizing the total number of erase operations, the deviation value of each block and the garbage collection time. NAND flash memory consists of pages of three types, such as valid pages, invalid pages and free pages. In order to achieve above goals, we use a page ratio to decide when to do garbage collection and to select the target victimblock. Additionally, we implement allocating method and group management method. Simulation results show that the proposed policy performs better than Greedy or CAT with the maximum rate 85% of reduction in the deviation value of the erase operations and 6% reduction in garbage collection time.

**Key words** : flash memory, garbage collection

· 이 논문은 제34회 추계학술대회에서 '플래시 메모리를 위한 페이지 비율 분석 기반의 적응적 가비지 컬렉션 정책'의 제목으로 발표된 논문을 확장한 것임 Copyright©2009 한국정보과학회; 개인 목적이나 교육 목적인 경우, 이 저작물의 전체 또는 일부에 대한 복사본 혹은 디지털 사본의 제작을 허가합니다.

† 학생회원 : 부산대학교 컴퓨터공학과  
cat@pusan.ac.kr

withsoul@pusan.ac.kr

†† 종신회원 : 부산대학교 컴퓨터공학과 교수

kdcungf@melon.cs.pusan.ac.kr

논문접수 : 2008년 12월 19일

심사완료 : 2009년 5월 22일

이 때, 사본은 상업적 수단으로 사용할 수 없으며 첫 페이지에 본 문구와 출처를 반드시 명시해야 합니다. 이 외의 목적으로 복제, 배포, 출판, 전송 등 모든 유형의 사용행위를 하는 경우에 대하여는 사전에 허가를 얻고 비용을 지불해야 합니다.

정보과학회논문지 : 시스템 및 이론 제36권 제5호(2009.10)

## 1. 서론

플래시 메모리는 비휘발성이며, 일반적인 컴퓨터 환경에서 사용되는 하드디스크 보다 부피가 작고, 가볍고, 소비 전력이 낮으며, 기계적인 동작이 필요 없기 때문에 응답 시간이 빠르고, 충격에 대한 내구성이 뛰어난 장점을 가지고 있어 소형 기기의 저장장치로 적합하다[1]. 최근 유비쿼터스 컴퓨팅 환경의 발전으로 휴대폰, PDA, 자동차 네비게이션 등 이동 가능한 정보기기들이 널리 보급되고 있으며, 플래시 메모리는 이러한 휴대용 정보기기의 저장장치로써 사용되며 지속적으로 발전하는 추세이다. 그러나 플래시 메모리는 기존 하드디스크와 다른 두 가지 단점이 있다[2,3]. 첫 번째 단점은 제자리 덮어쓰기가 불가능하여 갱신 연산을 수행할 때 동일한 주소에 바로 덮어 쓰기가 되지 않는다. 따라서 갱신 연산을 하기 위해서는 지움 연산이 추가적으로 발생 한다. 두 번째 단점은 플래시 메모리를 구성하고 있는 블록은 지움 연산 횟수가 제한이 되어 있다. 따라서 제한 횟수를 초과하는 블록들은 이후 쓰기 오류가 발생하게 되고 블록은 배드 블록 처리를 해야 한다. 결국 블록들의 지움 횟수가 균등하게 유지 되지 못하면 가용공간은 급격히 줄어드는 현상이 발생한다.

플래시 메모리 기반 임베디드 기기는 성능과 신뢰성을 만족시키기 위해 앞에서 언급한 두 가지 단점을 극복하는 것이 필요하다. 첫 번째 단점을 극복하기 위해서는 갱신 연산이 발생 했을 때 다른 공간에 새로운 데이터를 쓰고 이전의 데이터는 무효화시키는 방법을 일반적으로 사용하고 있다. 그리고 새로운 공간 확보를 위해서는 무효화된 데이터를 가지고 있는 블록들에 대해서 '가비지 컬렉션(Garbage Collection)'을 수행한다. 두 번째 단점을 극복하기 위해서는 블록의 지움 횟수를 고려하여 블록을 관리를 해야 한다. 지움 횟수가 지나치게 높은 블록의 활용을 줄이고 비교적 낮은 지움 횟수를 가지는 블록의 활용도를 높임으로써 전체 블록의 지움 횟수에 대한 균등화 정책(Wear-leveling policy)이 필요하다.

가비지 컬렉션은 지움 연산을 수반함과 동시에 대상 블록의 유효 페이지들을 새로운 영역에 복사하는 오버헤드가 존재한다. 특히 지움 연산은 읽기, 쓰기 연산에 비해 느리기 때문에 가비지 컬렉션 연산 횟수를 줄여야 하고, 균등화를 고려하기 위해서는 지움 대상 블록을 적절히 선택해야한다. 따라서 본 논문에서는 가비지 컬렉션 연산 횟수를 줄임과 동시에 블록의 지움 횟수를 균등하게 관리하는 균등화 정책을 제안을 한다. 이를 위해 가비지 컬렉션 수행을 위한 두 가지 조건을 제시하고, 각 조건에 따라 수행하는 가비지 컬렉션에서 Reclaim

Mode와 Wear-leveling Mode를 수행한다. 그래서 각 가비지 컬렉션에 맞는 지움 대상 블록 선택 기법을 적용하여 추가적인 지움 연산 오버헤드 없는 균등화 정책을 제안한다. 또한 가비지 컬렉션 시 블록의 탐색시간을 최소화 하는 그룹 관리 기법을 적용해 보다 효율적인 가비지 컬렉션 연산을 제안한다.

본 논문의 구성은 다음과 같다. 2장에서는 기존의 가비지 컬렉션 기법 연구에 대해 기술하고, 3장에서는 본 논문에서 제안하는 균등화 정책에 대해 설명한다. 4장에서는 제안하는 기법의 성능을 실험을 통해 평가하고, 5장에서는 결론 및 향후 과제에 대해 제시한다.

## 2. 관련 연구

과거 플래시 메모리 기반 시스템에서 가비지 컬렉션 연구는 가비지 컬렉션 횟수를 최소화 하는데 목표가 있었다[4,5]. 가비지 컬렉션 연산 시 Greedy 기법[4]은 유효페이지가 가장 적은 블록을 선택하여 유효 페이지를 복사하는 오버헤드를 최소화 하고 동시에 공간 확보율을 높였다. 하지만 블록의 균등화를 고려하지 않아 최악에 상황에서는 플래시 메모리의 수명을 단축되는 단점이 있다. Greedy 기법을 개선한 SWAP 기법[5]은 가비지 컬렉션 대상 블록 선정 시, Greedy와 동일하게 대상 블록을 선정하고 균등화를 위한 할당 기법을 제안하여 전체 지움 횟수를 낮추면서 균등화를 지원 하였다.

MODA[6]는 가비지 컬렉션에 영향을 끼치는 요소를 이용률, 무효율, 순수도로 정의하고 순수도를 높게 유지 하는 할당 정책을 제안하여 가비지 컬렉션 연산 횟수를 낮추었다. 하지만 역시 균등화에 대한 고려가 없는 한계가 있었다. 전체 블록의 지움 횟수를 고려한 균등화 정책으로는 CAT[7]기법이 있다.

CAT기법은 블록의 사용률과 블록이 할당된 시간, 그리고 블록의 지움 횟수를 고려해 블록의 지움 비용을 계산 하며, 가장 작은 비용을 가지는 블록을 가비지 컬렉션 대상 블록으로 선택하여 균등화를 지원한다. 하지만 가비지 컬렉션 연산 시 복사 되는 유효 페이지 수가 Greedy 기반의 기법에 비해 많아 회수되는 빈 페이지 수가 적어 전체 지움 횟수가 많아지는 단점이 있다. 따라서 본 논문에서는 가비지 컬렉션 연산 횟수를 최소화함과 동시에 블록의 지움 횟수 균등화를 고려한 기법을 연구하였다.

## 3. 제안하는 기법

### 3.1 페이지 비율 분석

플래시 메모리의 페이지는 사용되지 않은 빈 페이지(free page), 데이터를 저장 하는 유효 페이지(valid page), 삭제가 필요한 무효 페이지(invalid page)로 구

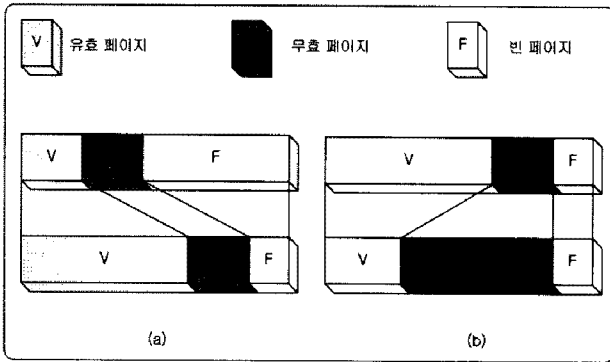


그림 1 파일 생성 및 삭제 시 플래시 메모리 페이지 비율 변화

분할 수 있다. 전체 플래시 메모리에서 각 페이지가 차지하는 비율은 파일연산 수행 시 항상 변하게 된다.

새로운 파일 생성 시 빈 페이지에 새로운 데이터를 할당함으로써 그림 1의 (a)와 같이 유효 페이지 수는 증가한다. 또한, 파일 삭제 연산 시 해당 데이터를 저장하는 유효 페이지는 무효 페이지로 상태가 바뀌므로 그림 1의 (b)와 같이 무효 페이지 수는 증가한다.

3.2 가비지 컬렉션 수행 조건

본 논문에서는 파일의 생성 또는 삭제 시 달라지는 페이지들의 비율을 분석하고 이를 이용해 두 가지 조건 중 하나를 만족하면 가비지 컬렉션을 수행한다.

첫 번째 조건은 그림 1의 (a) 상태와 같이 새로운 파일 생성 시 감소하는 빈 페이지의 비율(Freepage\_ratio)이 임계값  $TH_{free}$  이하일 때 가비지 컬렉션을 수행한다. 빈 페이지 비율은 전체 페이지 수( $n_{Total\_page}$ )와 현재 빈 페이지 수( $n_{Free\_page}$ )를 이용해 식 (1)과 같이 나타낼 수 있다. 이를 이용해 현재 빈 페이지 비율을 판단한다.

$$Freepage\_ratio = \frac{n_{Freepage}}{n_{Totalpage}} \quad (1)$$

두 번째 조건은 그림 1의 (b) 상태와 같이 기존 파일 삭제 시 증가하는 무효 페이지의 비율이 임계값  $TH_{invalid}$  이상일 때 가비지 컬렉션을 수행한다. 따라서 할당을 위한 빈 블록을 많이 확보하면 블록의 지움 횟수를 고려한 할당을 할 수 있어 전체 블록의 지움 횟수를 균등하게 유지할 수 있다.

가비지 컬렉션 연산 시 대상 블록 내 유효 페이지 복사 비용은 오버헤드로 작용한다. 따라서 두 번째 조건에 의한 가비지 컬렉션은 플래시 메모리 내에서 무효 페이지 비율이 높을 때 수행해야 바람직하다.

그림 2는 무효 페이지 비율(Invalidity\_ratio)에 따른 할당된 블록의 상태를 나타낸다. 무효 페이지 비율이 높으면 블록 전체가 무효 페이지로만 구성된 블록의 수가 많이 존재하거나 블록 내의 유효 페이지 수( $n_{Valid\_}$

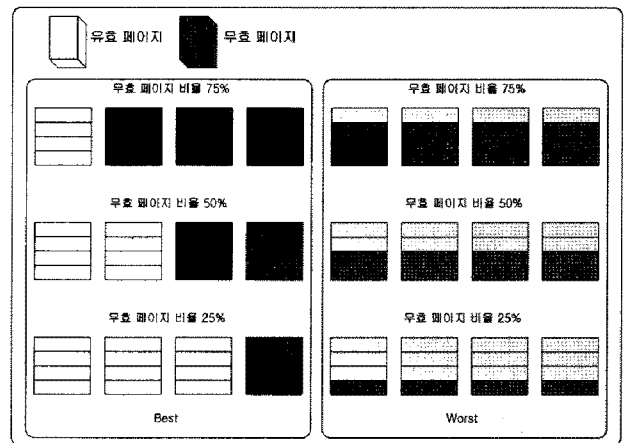


그림 2 무효 페이지 비율에 대한 블록 상태

page)가 낮게 유지된다. 따라서 무효 페이지 비율이 높을 때 가비지 컬렉션을 수행하면 오버헤드가 크게 증가하지 않는다. 무효 페이지 비율은 현재 사용된 페이지 수 중 무효 페이지 수( $n_{Invalid\_page}$ )의 비율이며, 사용된 페이지 수는 데이터가 기록 중이거나 기록되었던 유효 페이지 수와 무효 페이지 수의 합으로 식 (2)와 같이 표현한다.

$$Invalidity\_ratio = \frac{n_{Invalidpage}}{n_{Invalidpage} + n_{Validpage}} \quad (2)$$

3.3 가비지 컬렉션 모드

3.2에서 언급한 가비지 컬렉션 연산의 두 가지 수행 조건에 따라 가비지 컬렉션을 두 가지 모드로 나눌 수 있다. 본 논문에서는 3.2의 조건 중 첫 번째 조건인 플래시 메모리의 빈 페이지의 비율이 일정 수준 이하일 때의 가비지 컬렉션 모드를 회수 모드(Reclaim Mode), 두 번째 조건인 플래시 메모리의 무효 페이지 비율이 일정 수준 이상일 때의 가비지 컬렉션 모드를 균등화 모드(Wear-leveling Mode)로 정의하여 구분한다.

회수 모드에서는 플래시 메모리의 빈 공간이 부족하기 때문에 공간 확보에 초점을 맞춘다. 균등화 모드에서는 가비지 컬렉션 대상 블록 선정 시 블록의 지움 횟수를 고려하여 전체 블록의 지움 횟수를 균등화하는 데 초점을 맞춘다.

3.4 모드별 대상 블록 선정 방법

가비지 컬렉션 연산은 데이터가 할당된 블록들로 이루어진 데이터 블록 리스트를 탐색하여 지움 비용을 계산하는 것으로 이루어진다. 이때 가비지 컬렉션의 모드에 따라 각 블록 정보 영역의 필요한 정보를 참조해 블록의 지움 비용을 계산하고 가장 지움 비용이 낮은 블록을 가비지 컬렉션 대상 블록으로 선정한다.

따라서 회수 모드에서 가비지 컬렉션 대상 블록 선정 시 지움 비용은 식 (3)과 같이 블록의 유효 페이지 수를

고려하여 계산되며 가장 낮은 지움 비용(Costreclaim)을 가지는 블록이 선정되어 공간 확보를 할 수 있도록 한다.

$$Cost_{reclaim} = pagesInUse \quad (3)$$

균등화 모드에서 가비지 컬렉션 대상 블록 선정은 식 (4)와 같이 블록의 유효 페이지 수와 블록의 지움 횟수를 동시에 고려한 지움 비용(Costwearleveling)을 계산해 가장 낮은 지움 비용을 가지는 블록이 선정되어 공간 확보와 균등화를 동시에 만족하도록 한다.

$$Cost_{wearleveling} = pagesInUse \times eraseCount \quad (4)$$

### 3.5 그룹 관리 기법

가비지 컬렉션 대상 블록을 선정하려면 플래시 메모리의 전체 블록을 탐색하며 해당 블록의 지움 비용을 계산하고 가장 낮은 지움 비용을 갖는 블록을 선택하는 것이 좋다. 그러나 가비지 컬렉션을 수행할 때마다 전체 블록을 탐색하고 지움 비용을 계산하는 것은 또 다른 오버헤드가 될 수 있다. 따라서 본 절에서는 그림 3과 같이 그룹 관리 기법을 적용하여 가비지 컬렉션 시 탐색 범위를 줄인다.

플래시 메모리의 실제 블록들과 1:1 매핑이 되는 구조체인 블록 정보 영역을 일정 개수로 묶어 그룹을 구성한다. 그리고 그룹에는 그룹 정보 영역(GroupInfo)구조체를 유지하여 그룹 내 블록들의 정보를 관리한다. 그룹 정보 영역에는 그룹 내 블록들의 평균 지움 횟수 값(G\_avg\_ec)과 그룹 내 블록들의 평균 유효 페이지 수(G\_avg\_piu)와 같은 상태 정보를 가진다.

이 두 가지 상태 정보는 가비지 컬렉션 시 대상 블록을 찾기 전 대상 그룹을 찾는 데 활용된다. 대상 그룹은 3.3에서 제안한 회수 모드와 균등화 모드에 따라 다르게 선택된다. 회수 모드에서는 공간 확보에 초점을 맞추기 때문에 그룹 내 평균 유효 페이지 수를 비교해 가장 낮은 그룹을 선택한다. 따라서 그림 3의 그룹 1이 선택된다. 균등화 모드에서는 전체 블록의 지움 횟수를 균등화하는 데 초점을 맞추기 때문에 그룹 내 블록들의 지움 횟수 평균값이 가장 낮은 블록을 선택한다. 따라서 그림 3]에서 그룹 0이 선택된다.

가비지 컬렉션 대상 블록은 모드에 대한 그룹 선택 후 해당 그룹 내에서 선택을 하므로 가비지 컬렉션 탐색 영역은 전체 그룹의 수와 그룹 내 블록의 수의 합이 되어 전체 블록을 탐색하는 것에 비해 더욱 줄어든다.

### 3.5 전체 흐름도

제안하는 페이지 비율 분석 기반의 가비지 컬렉션 기법은 그림 4와 같은 흐름도로 표현할 수 있고, 크게 4단계로 나눌 수 있다.

1단계에서는 현재 플래시 메모리 상태가 가비지 컬렉션을 수행해야 하는지에 대해 판단을 한다. 이때 플래시 메모리의 페이지 비율을 분석하여 빈 페이지 비율이 임계값 이하일 때 회수 모드, 무효 페이지 비율이 임계값 이상일 때 균등화 모드를 선택하고 그렇지 않으면 가비지 컬렉션을 수행하지 않는다.

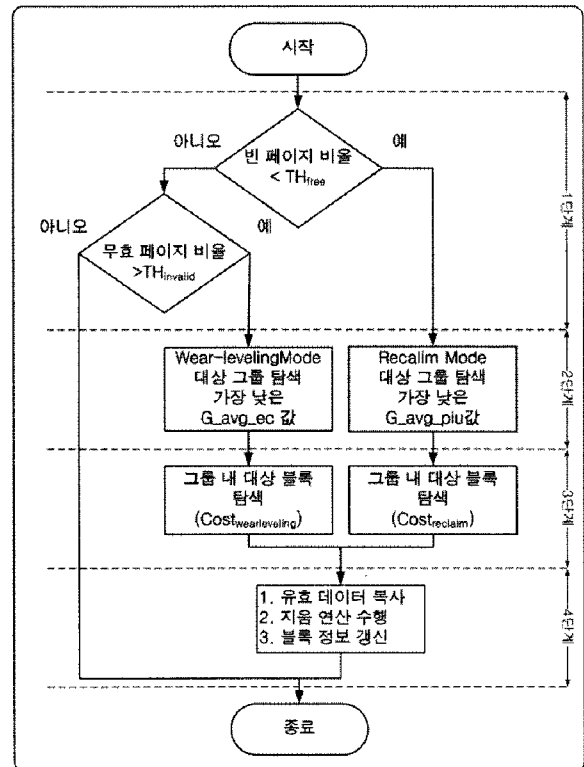


그림 4 전체 흐름도

블록 정보 리스트	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19
그룹 정보	그룹0				그룹1				그룹2				그룹3				그룹4			
eraseCount	2	4	6	8	6	6	6	6	6	7	8	9	8	8	8	8	10	10	10	10
pagesInUse	16	16	16	16	0	0	0	0	8	8	16	16	12	13	14	15	16	17	18	19
G_avg_ec	5				6				7.5				8				10			
G_avg_piu	16				0				24				13.5				17.5			

그림 3 그룹 관리 기법

2단계에서는 가비지 컬렉션의 모드에 따라 가비지 컬렉션 대상 그룹을 선정한다. 따라서 회수 모드일 때 평균 유효 페이지 수가 가장 낮은 그룹, 균등화 모드일 때 평균 지움 횟수 값이 가장 낮은 그룹을 선택한다.

3단계에서는 각 모드에 따라 식 (3)과 식 (4)를 적용하여 그룹내에서 가비지 컬렉션 대상 블록을 찾는다.

4단계에서는 선택된 지움 대상 블록 내의 유효 페이지들은 새로운 블록으로 복사되고, 해당 블록은 지움 연산을 수행한다. 그리고 블록에 대한 정보를 갱신함으로써 가비지 컬렉션은 종료된다.

### 4. 성능 평가

제안한 기법의 성능을 평가하기 위해 NAND 플래시 시뮬레이터[8]를 사용하였고, 표 1과 같은 조건에서 수행하였다. 로컬리티는 x/y로써, 전체 파일 중 x%의 파일에 대해 전체 연산의 y%를 집중한다고 정의한다. 따라서 50/50은 낮은 로컬리티, 10/90은 높은 로컬리티를 나타낸다.

TH<sub>free</sub>는 가비지 컬렉션 수행을 위한 최소한의 여유블럭을 두기 위해 0.01로 두고, TH<sub>invalid</sub>는 그림 5와 같이 플래시메모리의 사용량을 80%, 60%, 40%, 20%로 한 경우에 대해 로컬리티가 50/50인 trace로 실험을 하여 0.6으로 설정하였다. TH<sub>invalid</sub>가 낮으면 불필요한 가비지 컬렉션을 수행할 것이고, 높으면 wear-leveling mode가 수행되지 않아 균등화를 잘 지원하지 못할 것이다.

표 1 실험 조건 표

조건	값
블록 개수	320개
블록당 페이지 개수	32개
파일 크기	4+15KB
플래시 사용량	40%
파일 연산 횟수	10만번
로컬리티	50/50, 40/60, 30/70, 20/80, 10/90
TH <sub>free</sub>	0.01
TH <sub>invalid</sub>	0.60

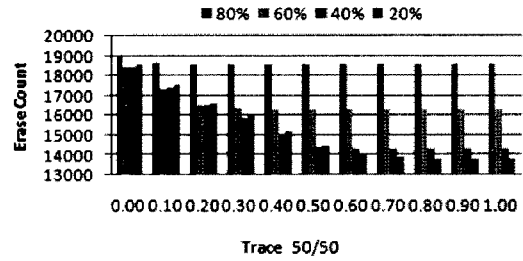


그림 5 플래시 메모리 사용률 및 TH<sub>invalid</sub>에 따른 Erase-Count 측정

#### 4.1 기존 기법과의 성능 비교

기존 기법에 대한 제안하는 기법의 성능의 증감은 그림 6과 같다. 전체 지움 횟수는 Greedy에 비해 평균 0.67%정도로 증가했지만, CAT에 비해 평균 1.06%정도로 감소했음을 알 수 있다. 지움 횟수 편차는 CAT에 비해 평균 11.33% 정도로 증가했지만, Greedy에 비해 평균 85.22% 정도로 크게 감소했음을 알 수 있어 블록을 균등하게 사용함을 나타내고 있다.

또한 배드 블록 조건을 지움 횟수 100으로 정의하여 측정된 배드 블록 발생시간은 그림 7과 같이 나타나 제안하는 기법은 Greedy 기법 보다 약 348%, CAT 기법보다 약 2%의 수명 향상 효과가 있다는 것을 알 수 있다.

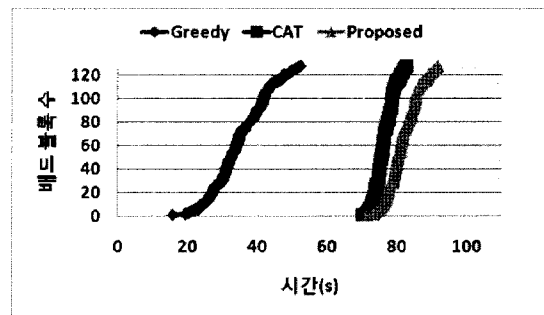


그림 7 배드 블록 발생 시간

#### 4.2 그룹 관리 기법의 성능 비교

그림 8은 그룹 내 블록 수를 10개, 16개로 묶은 그룹

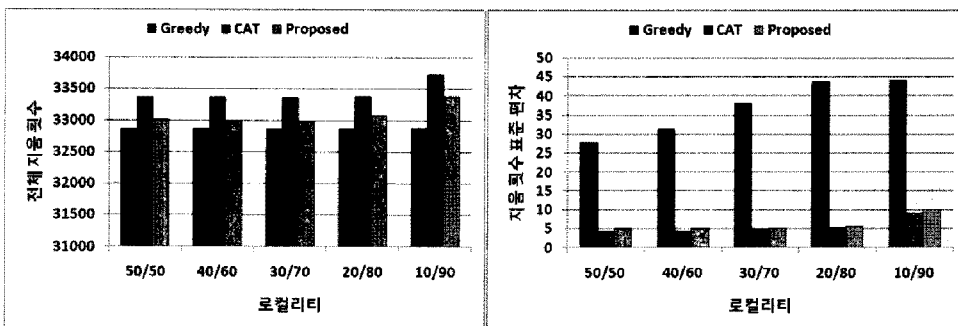


그림 6 기존 기법과의 성능 비교

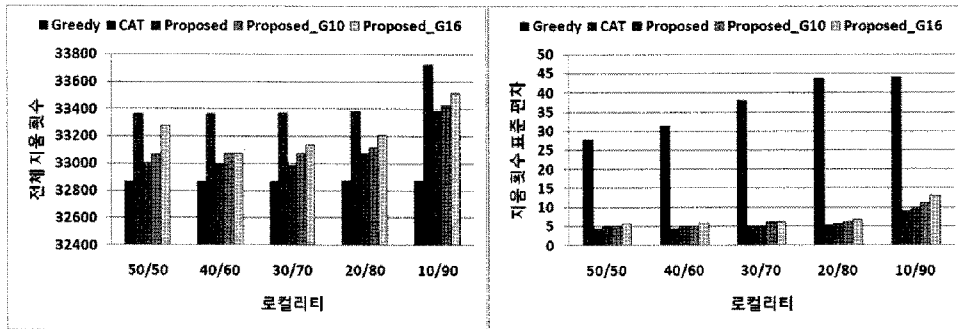


그림 8 그룹 관리 기법의 성능 비교

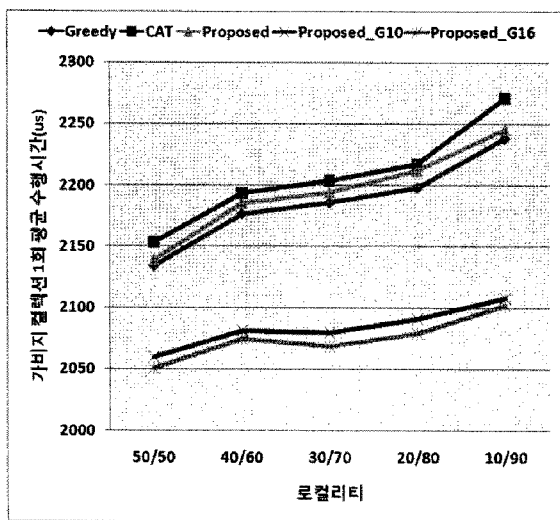


그림 9 가비지 컬렉션 1회 평균 수행 시간

관리기법을 그룹 관리 기법을 적용하지 않은 것과 비교한 결과이다. 그룹 관리 기법을 적용했을 때 지움 횟수 편차는 약 33% 정도 높게 나타나지만 Greedy기법과 비교했을 때 평균 80.14% 정도 감소시킴으로써 전체 지움 횟수 균등화를 만족하고 있음을 확인할 수 있다.

그림 9는 각 기법에 대해 가비지 컬렉션 시 평균 수행시간을 나타낸 결과이다. 그룹 관리 기법은 가비지 컬렉션 시 할당된 블록의 전체를 탐색하지 않기 때문에, 그룹 관리 기법을 적용하지 않은 기법에 비해 최대 6% 정도로 수행시간을 단축 시켰다.

### 5. 결론

본 논문에서는 페이지 비율 분석 기반의 가비지 컬렉션 기법을 제안하였다. 가비지 컬렉션을 수행하기 위해 페이지 비율 분석 기반으로 플래시 메모리의 빈 페이지 비율과 무효 페이지 비율을 이용하였고, 가비지 컬렉션 연산의 수행 조건을 두 가지로 정의하였다. 또한, 수행 조건에 따라 가비지 컬렉션 연산 모드를 회수 모드와 균등화 모드로 나누었고 각 모드에 따라 가비지 컬렉션 대상 블록을 찾는 지움 비용 계산을 달리였다. 그리고

그룹 관리 기법을 적용하여 전체 지움 연산 횟수와 각 블록의 지움 연산 횟수 표준 편차를 동시에 낮추고 가비지 컬렉션 수행 시간을 단축시켰다.

성능 측정 결과 제안하는 기법은 기존의 Greedy 기법과 비교해 지움 횟수 표준 편차를 평균 85.22% 감소하였고, CAT 기법과 비교해 전체 지움 횟수를 평균 0.67% 감소하였다. 그 결과 플래시 메모리의 배드 블록 발생 시간을 Greedy 기법의 348%, CAT 기법의 2% 정도 향상 시킬 수 있었다. 또한 그룹 관리 기법을 통해 가비지 컬렉션 수행 시간을 최대 6% 단축시켜 가비지 컬렉션 시 소요되는 오버헤드를 감소 시켰다.

향후 본 연구는 로컬리티와 파일 사이즈와 같은 특징들을 분석하여 입출력 파일 트레이스에 따라 적응적으로 변화하는 THinvalid를 구함으로써 범용적이고 더욱 효율적인 균등화 정책을 제안하는 방향으로 나아갈 것이다.

### 참고 문헌

- [1] F. Douglis, R. Caceres, F. Kaashoek, K. Li, B. Marsh and J. A. Tauber, "Storage Alternatives for Mobile Computers," *Proceedings of the 1st Symposium on Operating Systems Design and Implementation(OSDI)*, pp.25-37, 1994.
- [2] Samsung Electronics, "NAND Flash data sheet," [http://www.samsung.com/global/business/semiconductor/products/flash/Products\\_NANDFlash.html/](http://www.samsung.com/global/business/semiconductor/products/flash/Products_NANDFlash.html/)
- [3] Intel Corporation, "Flash File Systems Overview," <http://www.intel.com/>
- [4] 김경운, 김영필, 송인준, 유혁, "Greedy 방법을 개선한 플래시 메모리 지움 정책," *한국정보처리학회 춘계학술발표대회 논문집*, 제11권 제1호, pp.1685-1688, 2004.
- [5] Ohhoon Kwon, Kern Koh, "Swap-aware Garbage Collection for NAND Flash Memory Based Embedded Systems," *Seventh International Conference on Computer and Information Technology*, pp. 787-792, 2007.
- [6] 백승재, 최종무, "블록 클리닝 비용 분석에 기초한 MODA할당 정책 설계 및 구현", *정보과학회논문지*

시스템 및 이론, 제34권 제11호, pp.599-609, 2007.

- [7] Mei-Ling Chiang, Paul C. H. Lee, Rwei-Chuan Chang, "Cleaning policies in mobile computers using flash memory," *Journal of Systems and Software*, vol.48, 1999.
- [8] 옥동석, 이승환, 이태훈, 정기동, "NAND 플래시 파일 시스템 시뮬레이터 구현", *한국정보과학회 2008 한국 컴퓨터종합학술대회 논문집(A)*, 제35권 제1호, pp.331-332, 2008년 6월.



이 승 환

2007년 부산대학교 컴퓨터공학과 졸업(학사). 2009년 부산대학교 컴퓨터공학과 졸업(석사). 관심분야는 내장형 시스템, 파일 시스템



이 태 훈

2004년 부산대학교 정보컴퓨터공학과(학사). 2006년 부산대학교 컴퓨터공학과 석사. 2006년~현재 부산대학교 컴퓨터공학과박사과정. 관심분야는 내장형 시스템, 파일 시스템



정 기 동

1973년 서울대학교 졸업(학사). 1975년 서울대학교 대학원 졸업(석사). 1986년 서울대학교 대학원 계산통계학과 졸업(이학박사). 1978년~현재 부산대학교 컴퓨터 공학과 교수. 관심분야는 멀티미디어 시스템, 멀티미디어 통신, 병렬처리