

Wear-Leveling Queue를 이용한 효율적인 플래시 메모리 균등화 정책

이승환[○] 옥동석 이태훈 정기동

부산대학교 컴퓨터공학과

{cat[○],ok3, withsoul, kdchung}@pusan.ac.kr

Efficient Wear-Leveling Policy for Flash File System Using Wear-Leveling Queue

Seunghwan Lee[○], Dongseuk Ok, Teahoon Lee, Kidong Chung

Dept of Computer Engineering Pusan Nat'l Univ

1. 서론

최근 휴대용 멀티미디어 정보기기에 대한 저장 장치로 플래시 메모리에 대한 수요가 증가하고 있다. 플래시 메모리는 일반 하드 디스크 보다 내구성이 강하고, 부피가 작으며, 소비전력과 응답 시간이 작지만 제자리 덮어쓰기가 불가능한 특징을 있어 지움 연산이 추가적으로 필요하다. 또한 플래시 메모리의 블록은 지움 연산 횟수가 보통 10만 번으로 한계가 있고, 이를 초과하게 되면 배드 블록이 되어 사용 할 수 없게 된다. 따라서 플래시 메모리를 위한 파일 시스템 설계 시 지움 연산에 대한 고려를 중점적으로 해야 한다.

본 논문은 플래시 메모리의 수명을 향상시키기 위한 Wear-Leveling Queue를 이용한 효율적인 균등화 기법을 제안한다.

2. 관련 연구

Greedy 정책은 블록의 사용률이 가장 낮은 블록을 선택하여 지움 연산을 수행하여 지움 연산의 비용을 최소화한다. 하지만 지움 연산이 특정 부분에 편중되어 수행이 될 경우 전체 블록을 균등하게 사용 할 수 없기 때문에 플래시 메모리의 수명을 오랫동안 유지할 수 없게 된다.

Cost-Benefit 정책은 Greedy 정책에서 제안한 최소 비용을 선택하는 정책과 블록이 할당된 시간(age)을 고려하여 비용을 계산한다. 하지만 지움 횟수에 대한 고려가 없어 전체 블록의 균등 사용을 충분히 만족 시키지 못한다.

CAT 정책은 세그먼트의 사용률(u)과 세그먼트가 할당된 시간(age), 그리고 세그먼트의 지움 횟수(ec)를 고려 한 세그먼트의 지움 비용을 계산을 하여 비용이 가장 낮은 값을 선택 한다. 이는 세그먼트의 지움 횟수를 고려하여 해당 세그먼트가 우선적으로 지움 연산이 일어 날수 있도록 하여 Greedy 정책이 지니고 있는 문제점인 높은 지역성을 가지는 연산에 불리한 점과 Cost-Benefit가 지니고 있는 문제점인 전체 블록의 지움 횟수에 대한 균등 사용을 지원 하지 않는 점을 극복하여 전체 블록이 균등하게 사용될 수 있도록 하였다. 하지만 식의 복잡도로 연산 시간이 매우 오래 걸리는 문제가 있게 된다.

3. 제안하는 기법

3.1 균등화를 위한 할당 기법

플래시 메모리의 균등화를 위해서는 블록의 지움 횟수를 고려하여 할당하여 블록들 간의 지움 횟수 차이를 최소화해야 한다. [그림 1]의 (b) 리스트는 할당을 위한 빈 블록 리스트이며 전체 빈 블록 리스트 중 지움 횟수가 낮은 m개의 빈 블록으로 구성되어 새로운 데이터를 위한 할당 시 가장 낮은 지움 횟수를 가지는 블록을 우선적으로 할당한다.(c) 리스트 역시 빈 블록을 유지하며 데이터 블록이 지움 연산을 통해 빈 블록이 되었을 때 삽입이 된다. 그리고 (b) 리스트의 사이즈가 0이 되면 (c) 리스트에서 지움 횟수가 낮은 블록 순으로 (b) 리스트를 구축한다.

3.2 균등화를 위한 지움 연산 기법

[○]이 논문은 2단계 두뇌한국21사업에 의하여 지원되었음

플래시 메모리내의 특정 블록의 오랫동안 갱신되지 않은 데이터들은 다른 블록으로 복사한 후 해당 블록에 지움 연산을 수행 하는 균등화를 위한 지움 연산 기법을 적용하고 새로운 데이터 할당 시 균등화를 위한 할당 기법을 수행하여 전체 블록들의 균등 사용을 할 수 있도록 해야 한다.

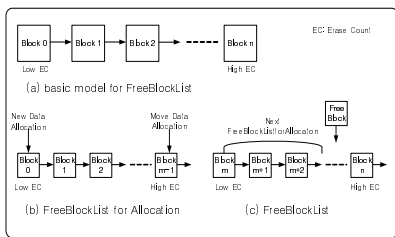
3.2.1 WLQ (WearLeveling Queue)

새로운 블록 할당이 발생하면 해당 블록 정보를 WLQ의 마지막 노드로 삽입을 한다. 그리고 블록의 지움 횟수에 따른 정렬은 하지 않고 WLQ 첫 노드를 대상으로 균등화 정책 조건을 판단한다. 균등화 정책 조건 판단 후 WLQ는 다음 노드에 대한 조건을 판단하기 위해 회전 연산을 하며 WLQ의 마지막 노드로 회전되는 노드의 회전 횟수 값을 증가 시켜 해당 노드가 WLQ에서 얼마나 회전 하였는지 판단한다.

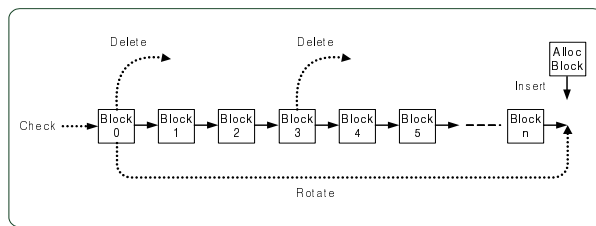
3.2.2 균등화 정책 조건 판단

블록의 지움 횟수의 최대값과 현재 값의 차이가 임계값을 넘으면 데이터 이동을 수행하고 그렇지 않은 경우에는 블록에 대한 지움 비용을 블록 사용률(u)과 지움 횟수(EC) 그리고 WLQ에서 블록 회전 횟수(rotation_count)를 가지고 계산하여 식(1)과 같이 판단한다. 균등화 정책 조건 판단은 1회 판단 시 WLQ의 전체 노드가 아닌 미리 정해진 개수만큼만 계산을 하여 계산 비용을 줄인다.

$$Cost = u \times \frac{EC_{MAX} - EC_{CUR}}{rotation_count} \quad (1)$$



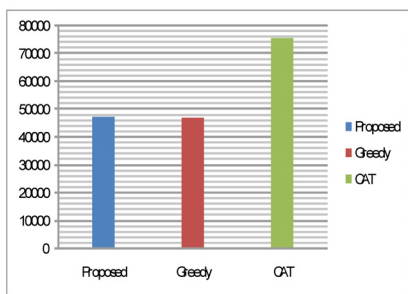
[그림 1] 할당리스트



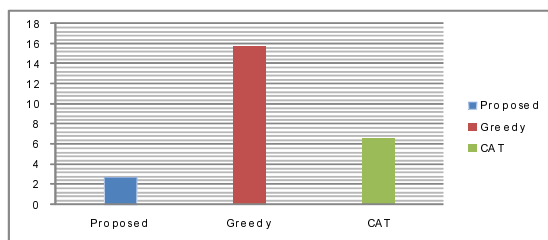
[그림 2] WLQ(Wear-Leveling Queue)

4. 성능 평가

제안 하는 기법은 [그림 3]와 같이 전체 지움 연산 횟수를 CAT 와 비교해 약 60% 정도로 줄여 Greedy와 비슷한 결과를 나타내 지움 연산 시 오버헤드를 줄였으며, [그림 4]과 같이 전체 블록의 지움 횟수의 편차를 최소화 하여 CAT의 결과의 약 40%정도 Greedy의 결과 값의 약 18%의 결과가 나타났다.



[그림 3] 전체 지움 횟수 비교



[그림 4] 전체 블록의 지움 횟수에 대한 표준 편차

5. 결론

본 논문은 NAND 플래시에서 블록의 지움 횟수를 균등화하기 위한 기법을 제안하였다. 균등화를 위한 할당 기법으로 낮은 지움 연산을 우선 적으로 할당하여 높은 지움 연산을 갖는 블록들의 할당을 늦추어 균등화를 지원하고, 균등화를 위한 지움 연산으로 블록의 지움 횟수와 사용률 그리고 WLQ 회전 횟수를 모두 고려하여 균등화를 지원한다. 그리고 부분적인 비용계산을 통해 오버헤드를 줄여 기존의 방식보다 향상된 균등화를 지원한다.

앞으로 NAND 플래시의 대용량화가 실현되고 있고 실제 관리해야 할 블록이 더욱 많아짐에 따라 연산의 오버헤드가 발생하게 된다. 그러므로 균등화 정책도 낮은 비용으로 지원이 될 수 있도록 개발 되어야 할 것이다.