

플래시 메모리의 FTL(Flash Translation Layer)에서 데이터 패턴에 따른 동적 블록 할당 기법

조은영^o 김종진 손진현

한양대학교 컴퓨터 공학과

eycho@database.hanyang.ac.kr, jjkim@database.hanyang.ac.kr,

jhson@cse.hanyang.ac.kr

Dynamic block-allocation Flash Translation Layer according to the hot and cold data patten

Eunyoung Cho^o, Jongjin Kim, Jinhyun Son

Department of Computer Science Engineering, Hanyang University, Ansan

1. 서 론

최근 플래시 메모리는 비휘발성 메모리로 저 전력 소비, 빠른 접근 속도, 높은 신뢰성을 가지고 있으며 가볍고 크기 또한 작다[1]. 이러한 뛰어난 특성과 가격의 감소, 용량의 증대 때문에 플래시 메모리는 소비자, 내장형 시스템, 모바일 컴퓨터 그리고 디스크 대치용의 다양한 기기에서 사용 될 것이다. 그러나 이 메모리는 그러나, 이 메모리는 덮어 쓰기(overwrite)가 가능하지 않고, 쓰기 작업과 지우는 작업의 단위가 다르며 메모리의 수명이 제한되어있는 단점이 있다. 그러므로 물리적인 계층과 논리적인 계층을 사상 시켜주는 역할을 하는 FTL(Flash transfer layer)이 존재한다.

FTL의 문제 해결 기법 중 기존의 대표적인 것으로는 로그 블록 기법[2]과 완전연관섹터변환(Fully Associative Sector Translation, 이하 FAST)[3] 기법이 있다. 이러한 기법들은 순차적인 데이터 패턴과 일정하지 않은 데이터 패턴을 고려하여 설계되었다. 앞서 언급한 로그 블록 기법은 대량의 순차 섹터 연산에는 강하나 갱신 블록의 활용률이 좋지 않다는 단점을 가지고 있다. 이후에 나온 FAST 기법은 위의 단점을 개선하기 위해서 임의 쓰기의 갱신 블록을 추가하였다. 그러나 임의 쓰기의 갱신 블록은 한 개의 갱신 블록을 소거할 때 유효한 페이지의 복사가 많이 일어난다. 그래서 우리는 특정 영역을 자주 사용하는 데이터 패턴에 다른 기법을 사용하여 관리하고 최신의 변경된 데이터 패턴의 정보를 반영하여 동적으로 업데이트 블록을 할당하고자 한다. 다양한 데이터 패턴에 따라 동적으로 블록을 할당한 결과 쓰기와 소거 작업의 수를 감소시키는 성능 개선을 보였다.

2. 본 론

2.1 전체적인 구조

동적 블록 할당 기법의 기본적인 아이디어는 두 가지의 데이터패턴 즉, 핫 블록과 콜드 블록 안의 핫 페이지로 나뉜다. 핫 블록은 블록내의 전체 페이지의 절반 이상이 업데이트 된 경우이다. 또한, 처음 입력된 데이터는 아직 블록의 종류가 정해지지 않아 핫 블록처럼 수행된다. 콜드 블록 안의 핫 페이지는 이와 반대로 전체 페이지의 절반 이하만이 업데이트 되는 상태이다. 핫 블록은 순차적인 데이터 패턴과 블록 전체가 패턴 없이 업데이트가 일어날 때, 한 개의 원천 블록에 한 개의 갱신 블록이 할당된다. 콜드 블록 안의 핫 페이지에서 갱신 블록은 몇 개의 원천 블록을 공유한다. 이는 유효한 데이터 복사로 인해 발생하는 오버헤드를 줄이기 위함이다. 블록은 재배열 과정에서 블록의 최신 상태에 따라 핫 블록과 콜드 블록내의 핫 페이지로 구분된다.

2.1 블록 재배열

● 핫 블록을 위한 합병 연산

이 합병 연산은 핫 블록으로 구분된 경우에 이전의 방식과 유사하다. 핫 블록으로 구분되어서 순차적으로 갱신 블록에 쓰여 졌을 경우에는 매우 단순하다. 이 과정에서 갱신 블록에 더 이상의 남은 페이지가 없게 되면, 원천 블록을 삭제하고 갱신 블록을 원천 블록으로 교체하면 된다. 위의 과정은 스위치 연산이라고 부른다. 이 과정은 소거 연산과 약간의 사상 정보 변경으로 실행하기 때문에 블록 재배열로 인한 오버헤드가 없다. 순차적이지 않게 수정이 된 갱신 블록에서는 스위치 연산과 다른 방법이 사용된다. 원천 블록에 할당된 갱신 블록이 모두 소모되었을 때 합병 연산이 실행된다. 먼저 한 개의 새로운 자유 블록을 선정하고, (새로운 자유 블록 선정 방법은 소거 횡수 평준화 기법 (wear-leveling)을 고려하여 선정한다.) 새로운 블록에 원천 블록에서 유효한 페이지와 갱신 블록에서 유효한 페이지를 각각 페이지 번호순으로 기록한다. 이 새로운 블록을 원천 블록으로 사용하고 이전의 원천 블록과 갱신 블록을 삭제한다.

● 콜드 블록 안의 핫 페이지의 합병 연산

콜드 블록 안의 핫 페이지로 분류된 블록의 갱신 블록은 몇 개의 원천 블록을 공유한다. 먼저, 갱신 블록 중에서 유효한 페이지를 가장 작게 가진 희생 블록을 선정한다. 하나의 새로운 블록을 할당하여 갱신 블록에서 유효한 페이지를 복사한다. 갱신 블록의 유효한 페이지가 모두 복사되면, 새로운 블록의 사상 정보를 변경하고 희생 블록을 삭제한다. 이 합병 연산은 약간의 소거 연산의 실행과 사상 정보의 변경이 필요하다.

마지막으로 더 이상의 여유 공간이 없을 때 수행되는 과정에 대해 설명한다. 희생 블록으로 선정된 블록은 합병 연산을 통해 자유 블록으로 만든다. 그래도, 자유 블록이 부족하면 미리 정해진 최소 갱신 블록의 수가 될 때까지 합병한다. 이렇게 우리의 블록 재배열은 최악의 경우에만 갱신 블록의 수가 제한된다.

3. 결론

본 논문에서 우리가 제시한 동적 블록 할당은 순차적인 데이터 패턴과 블록 전체가 자주 업데이트 되는 패턴 그리고 특정 부분이 자주 갱신되는 패턴으로 나누어 관리한다. 이 전 논문과는 다르게 특정 부분이 자주 갱신되는 데이터 패턴에 대해 따로 관리한다. 데이터 패턴에 따라 효율적으로 블록을 할당함으로써 이전 보다 나은 성능을 보인다. 이 기법은 다양한 패턴의 쓰기에 높은 성능을 나타내므로 다양한 기기에서 사용 가능하다. 특히, 자주 갱신되는 패턴에 성능이 높은 것을 알 수 있으므로 이러한 패턴의 데이터 타입에 특성을 가진 시스템으로는 운영체제로 들 수 있는데, 이는 디스크 대체용과 같은 대용량과 고속화에 적합하다.

향후 연구로는 데이터 패턴을 판단하는데 초기시간이 걸린다. 이 초기 시간에는 약간의 데이터 공간 활용률이 떨어지게 된다. 그리고 이 기법은 플래시 메모리가 대용량의 여유로운 공간을 두고 사용할 때 좋은 성능을 보인다. 위와 같은 단점을 개선하는 것이 앞으로 해결해야 할 과제이다.

참 고 문 헌

- [1] F. Douglass, R. Ciceris, F. Kaashoek, K. Li, B. Marsh, and J. A. Tauber, "Storage alternatives for mobile computers," in Proceedings of the 1st Symposium on Operating Systems Design and Implementation (OSDI), 1994, pp. 25-37.
- [2] J. Kim, J. M. Kim, S. Noh, S. L. Min, and Y. Cho. A space-efficient flash translation layer for compactflash systems. IEEE Transactions on Consumer Electronics, 48(2):366-375, May 2002.
- [3] S.-W. Lee, D.-J. Park, T.-S. Chung, D.-H. Lee, S.-W. Park, and H.-J. Songe. FAST: A log-buffer based ftl scheme with fully associative sector translation. In The 2005 US-Korea Conference on Science, Technology, & Entrepreneurship, August 2005.