韓國컴퓨터情報學會 論文誌
第14卷 第4號, 2009. 4.

2009-14-4-1-2

에너지 및 성능 효율적인 이종 모바일 저장 장치용 동적 부하 분산

김 영 진*, 김 지 홍**

Energy and Performance-Efficient Dynamic Load Distribution for Mobile Heterogeneous Storage Devices

Young-Jin Kim*, Jihong Kim**

요 약

본 논문에서는 운영체제 수준에서 에너지 절감과 함께 I/O 성능 개선을 목적으로 하여 소형 하드 디스크와 플래시 메모리를 이종의 저장 장치로 가지는 모바일 시스템에 대해 동적 부하 분산 기법을 제안한다. 제안 기법은 부하가 에너지 및 성능 효율적인 방법으로 하드 디스크와 플래시 메모리의 이종성의 저장 장치 구성에 대해서 어떻게 효율적으로 분산될 수 있을 것인지를 발견하기 위하여 파일 배치 기법과 버퍼 캐시 관리 기법을 결합하는 접근법을 취한다. 제안한 기법은 폭넓은 시뮬레이션을 통해서 기존의 기법들과 비교하여 이종의 모바일 저장장치들에 대해서 더 개선된 실험 결과를 보이는 것으로 나타났다.

Abstract

In this paper, we propose a dynamic load distribution technique at the operating system level in mobile storage systems with a heterogeneous storage pair of a small form-factor hard disk and a flash memory, which aims at saving energy consumption as well as enhancing I/O performance. Our proposed technique takes a combinatory approach of file placement and buffer cache management techniques to find how the load can be distributed in an energy and performance-aware way for a heterogeneous mobile storage pair of a hard disk and a flash memory. We demonstrate that the proposed technique provides better experimental results with heterogeneous mobile storage devices compared with the existing techniques through extensive simulations.

▸ Keyword : 부하 분산(load distribution), 에너지(energy), 성능(performance), 이종 모바일 저장 장치 (heterogeneous mobile storage), 파일 배치(file placement), 버퍼 캐시 관리 (buffer cache management)

• 제1저자, 교신저자: 김영진
• 투고일 : 2009. 03. 09, 심사일 : 2009. 03. 26, 게재확정일 : 2009. 04. 19.
* 선문대학교 컴퓨터공학부 전임강사 ** 서울대학교 컴퓨터공학부 교수
※ 본 연구는 정보통신부 및 정보통신연구진흥원의 대학 IT연구센터 지원사업의 연구결과로 수행되었음. IITA-2008-C1090-0801-0020). 또한 이 논문은 2009년도 정부(교육과학기술부)의 재원으로 한국과학재단의 지원을 받아 수행된 연구임 (No. R0A-2007-000-20116-0).

Ⅰ. 서 론

근래에 이르러, PMP (portable media player)와 같은 모바일 정보기기에서 게임, 영화재생 등과 다양한 멀티미디어 응용 프로그램들이 최종 사용자(end user)간에 널리 사용되어 오고 있다. 그러한 기기 상에서 대규모의 멀티미디어 데이터를 처리하는 것은 빠르고 신뢰성 있는 모바일 저장 장치를 요구하게 된다. 이러한 목적으로 작은 크기 (2.5″ 또는 이보다 작은) 하드 디스크 드라이브들이 널리 채택되고 있다.

그러나 하드 디스크 드라이브들은 비트당 매력적인 저비용을 가지지만 상당한 전력을 소모하는 것으로 알려져 있으며 (전체 컴퓨팅 전력 소모의 20~30% 정도로 보고되고 있음) 임의 (random) I/O 요구들에 대해서 좋지 않은 성능을 보인다. 이러한 높은 소모 전력은, 일반적으로 배터리 기반으로 동작하는 모바일 정보 기기에 있어서 큰 문제가 될 수 있다. 또한 하드 디스크들은 내부의 기계 메커니즘으로 인해 외부 충격에 상당히 약하다.

하드 디스크 드라이브들의 이러한 약점들을 극복하기 위하여 이종의 저장 장치들을 사용하는 방법이 많이 연구되어 오고 있다. 서로 다른 이종의 저장 장치들을 결합하여 사용함으로써 비용 대비 개선된 성능과 절감된 전력 소모가 얻어 지기 때문이다. 작은 하드 디스크와 플래시 메모리의 쌍 [1], 하드 디스크와 MEMS (micro-electro mechanical system)기반의 저장 장치의 결합 [2], 그리고 다중 수준 셀 (multi-level cell, MLC)과 단일 수준 셀 (single-level cell, SLC)의 결합 플래시 메모리 [3]은 이러한 이종 저장 장치 시스템의 좋은 예들이다.

가장 대표적인 이종 저장 장치 시스템은 하드 디스크와 NAND 플래시 메모리를 결합한 것이다. NAND 플래시 메모리는 하드 디스크의 자리를 위협하는, 가장 상업적으로 성공한 저장장치로 여겨지고 있다 [4]. 표 1에서 보이는 것처럼, NAND 플래시 메모리 (정확히는 단일 수준 셀 NAND 플래시)는 하드 디스크보다 적어도 42배 정도 적은 동작(active) 모드 전력 소모를 보이고 있으며 66배 빠른 쓰기 응답 시간을 보인다.

하지만 NAND 플래시는 최근에 소형의 디스크 드라이브보다 적어도 2.5배 이상 기가바이트 (GB)당 가격이 높다. 플래시 메모리는, 표 1에서는 볼 수 없지만, 또한 쓰기(write)전에 소거(erase)가 되어야 하며 블록당 제한된 소거 횟수를 가진다. NANA 플래시의 빠른 응답 시간과 하드디스크의 비트 당 낮은 가격은 각각 모바일 저장 장치 시스템을 구성하는데 매력적이 될 수 있다. 그러나 하드 디스크의 높은 전력 소모와 플래시 메모리의 높은 비트당 가격은 비록 완화되고 있는 추세에 있지만 문제가 되는 정도이다. 그러므로 이러한 장치들을 함께 사용하는 각 저장 정치 장점을 이용함으로써 상승적으로 유용한 저장 장치 해법을 제공해줄 것으로 기대된다. 본 논문에서는 바로 하드 디스크와 NAND 플래시 메모리 장치를 쌍으로 사용하는 이종 저장 장치 시스템에 대한 연구 내용을 다룬다.

표 1. 랩탑 디스크, 소형 디스크 및 NAND 플래시 메모리의 특성 [9]-[13]
Table 1. Characteristics of a Laptop Disk, a Small Form-Factor Disk, and a NAND Flash Memory [9]-[13]

Device		Hard disk		NAND flash
		2.5″	1.8″	
Latency (us / 512 B)	Read	19100	22100	35.8
	Write	19100	22100	288
Power	Active	2300	1400	33
	Idle	950	400	0.13
	Standby	250	200	N/A
Cost per GB ($)		0.82	1.82	4.53

에너지 및 성능은 현대 컴퓨팅 시스템에서 두 중요한 설계 제약 요건들이며 많은 연구들이 이 요건들을 개선하고자 수행되어 오고 있다. Pinheiro 등은 클러스터 기반 시스템을 대상으로 성능을 개선하면서도 에너지를 절감하기 위한 부하 균등화 (load balancing) 및 비균등화 (unbalancing) 알고리즘을 제안하였다 [5]. 이종 저장 장치들을 기반으로 하는 모바일 컴퓨팅 시스템에서는 이러한 부하 균등화를 적절히 제어하는 것이 아주 중요하다. 에너지 효율적 측면이나[1] 성능 측면에서의[6] 부하 균등화 기법에 대한 개별 연구들은 많이 수행되어 왔으나 에너지와 성능을 동시에 고려하는 관점에서 소프트웨어 특히 운영 체제가 이종의 모바일 저장 장치들을 어떻게 잘 운용할 수 있는지에 대한 깊은 고찰이 없는 상태이다.

본 논문에서는 운영체제 수준에서 에너지를 절감하면서도 성능을 개선하는 목적으로 동적인 부하 분산 기법을 제안한다. 이러한 목적으로 본 논문의 연구 접근법은 2 가지의 독립적인 부하 분산 기법들을 상승적으로 운영체제 수준에서 결합하는 것이다. 첫째는 PB-PDC라고 불리는 이종 저장 장치에 대한 파일 분산 기법이며 [1] 두 번째는 DAC라고 불리는 장치인지 버퍼 캐시 관리 기법이다 [6]. PB-PDC는 하드 디스크와 플래시 메모리로 구성되는 모바일 저장 장치 시스템에 대해 전체 시스템 에너지 소모를 절감하는 것에 초점을 맞추고 있는 반면에 DAC는 동일한 저장 장치 구조에 대해서 요청당 평균 응답 시간을 개선하는 것에 초점을 맞추고 있다.

본 논문의 나머지는 다음과 같이 구성되어 있다: II장은 관

런 연구를 살펴보고 III장에서는 이 논문의 동기에 대해 언급을 한다. IV장과 V장은 에너지 및 성능 개선 목적의 부하 분산 기법에 대해서 각각 설명한다. VI장은 실험 환경 및 실험 결과를 제시한다. 마지막으로 VII장에서는 요약과 함께 결론을 기술한다.

II. 관련 연구

지금까지 모바일 하드 디스크와 플래시 메모리를 결합하는 많은 연구가 수행되어 오고 있다. March 등 [7], Bisson 등 [8], Chen 등 [14], 그리고 Kgil 등은 [15] 플래시 메모리를 비휘발성 캐시(non-volatile cache, NVC)로 사용하여 가까운 미래에 사용될 가능성이 높은 데이터 블록들을 여기에 저장함으로써 하드 디스크가 더 오래 저전력 모드에 머무를 수 있도록 하는 기법들을 연구하였다. Bisson 등은 하드 디스크 대신에 플래시 메모리로 쓰기 요청들을 전송하게 하는 기법에 초점을 맞춘 반면에 Chen 등은 최근에 플래시 메모리를 캐시 (cache)와 선출입 버퍼 (prefetch buffer) 및 쓰기 버퍼를 분할하고 에너지를 절감하는 연구를 수행하였다. Kgil 등은 플래시 메모리를 2차 수준의 버퍼 캐시로 사용함으로써 메인 메모리의 전력 소모를 절감하는 것에 초점을 두었다. 하지만 이러한 연구들은 모두 에너지 절감 측면에서의 기법 연구로 부하 분산 관점에서 에너지 소모와 성능 개선 측면에서의 기법에 대해서는 주의 깊게 연구되지 못하였다.

최근에 Bisson 등은 [16], 플래시 메모리와 하드 디스크가 결합된 하이브리드 하드디스크를 대상으로 하는 스핀 다운 (spin-down) 알고리즘들을 개발하였다. 특히, 이 연구에서는 플래시 메모리를 분리된 읽기 및 쓰기 캐시로 나누어 사용하는 기법, I/O 세부 시스템의 개선 사항 및 I/O 요청들의 플래시 메모리로의 재전송 기법 들을 연구하였다. 이 연구는 주로 에너지 절감의 효율성 개선과 스핀 업 (spin-up)의 지연 개선에 대해서 언급하고 있는데 버퍼 캐시나 파일 재배치 등과 같은 부하 분산에 대한 고려는 없다.

삼성과 마이크로소프트 (Microsoft)는 상업적으로 하이브리드 하드 디스크 기술을 개발하였는데 이는 하드 디스크에 NAND 플래시 메모리를 NVC로 연결함으로써 전반적인 성능을 개선하고 전력 소모를 절감하며 모바일 컴퓨터의 신뢰성을 증가시키고자 하였다 [17, 18].

유사하게 인텔 (Intel)에서는 Robson 기술을 개발하였는데 플래시 메모리를 주기판 (main board) 수준에서 NVC로 사용하여 시스템 전반의 응답성을 높이고 다중 작업 (multi-tasking)을 개선하며 배터리 수명을 늘이고자 하였다. 이러한 연구들은 에너지 및 성능을 경험적으로 (heuristically) 개선하는 기법들을 제시하였으며 부하 분산이라는 측면을 고려하지 않았다 [19].

한편, Kim 등은 하드디스크와 플래시 메모리의 이종 저장 장치 구성에 대해서 개별적인 연구에서 파일 배치에 따른 에너지 효율적인 부하 분산 효과를 연구와[1] 저장 장치의 특성과 작업 부하의 특성을 고려하여 버퍼 캐시의 성능 효율을 높임으로써 성능 중심의 부하 분산 연구[6]를 수행하였다. 하지만, 이 연구들은 각각 에너지와 성능 지표를 최소화하는 목적만을 가지고 있으므로 본 연구에서의 목표인 에너지와 성능 동시 효율적인 부하 분산 방법과는 달리 제한된 연구 범위를 가지고 있다.

III. 연구 동기

1. 연구 동기 개요

본 논문의 연구 동기는, 이종의 저장 장치들을 사용하는 모바일 저장 장치 시스템을 대상으로 에너지 및 성능 동시 개선을 이루기 위해 부하 분산에 대해서 새로이 요구되는 사항들로부터 발생한다. 모바일 정보 기기에서는 제한 배터리 기반의 동작으로 인해 에너지 소모가 아주 중요한 제약 조건이다. 게다가 시스템 전체 성능 역시 최종 사용자(end user)가 멀티미디어 응용 프로그램들을 수행할 때 긴 지연을 느낄 수 없도록 하기 위하여 중요하게 고려가 되어야 한다. 이러한 이유로 이종의 저장 장치 시스템에서 부하 분산에 대한 연구가 보다 엄밀히 수행되어야한다.

2. 부하 균등 및 접근법

부하 (일) 균등화는, 사용가능한 저장 장치들 간에 부하가 균등히 분배되어 접근되는 것을 의미하며 따라서 일정한 주기 내에 각 저장 장치에 대한 누적 지연이 같은 것을 의미한다 [5, 22]. 부하 균등화는 장치들 간에 I/O 요청 분포에 매우 밀접하게 연관되어 있으며 그 목표는 골고루 분산되어진 I/O 요청 처리를 통해서 전반적인 성능을 개선하는 것에 있다. 이와 반대로 부하 비균등화 또는 집중화는 I/O 요청들을 일부의 장치들에 편중(skew)시키려고 노력하는 것이다. 이를 통해 모든 유휴의 (idle) 장치들을 저전력 모드로 진입시킴으로써 전력 소모를 절감할 수 있다. 그러나 다시 요청이 도달하 는 경우 빠른 시간에 응답을 하지 못함으로 인해서 (정상 동작 상태로 다시 돌아가는 wake-up 시간이 필요함) 성능상의 문제를 초래할 수 있다. 따라서 부하는 에너지와 성능 측면에서 동일하

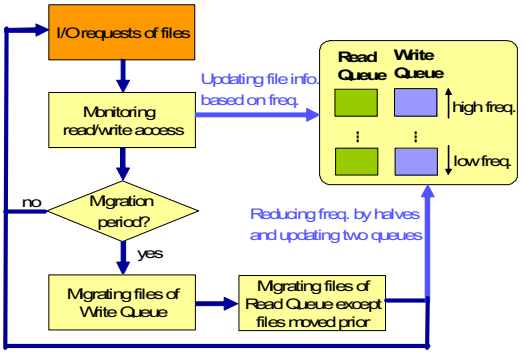

그림 1. PB-PDC의 전체 수행 흐름
Fig. 1. Overall Execution Flow of PB-PDC

게 중요하게 분포되도록 고려되어야 한다.

I/O 요청 분포는 파일 시스템 수준에서 저장 장치들 상의 각 파일의 분포뿐만 아니라 버퍼 캐시 관리에 의존한다. 결과적으로 부하 균등화는 파일 배치와 버퍼 캐시 관리 방법에 직접적으로 관련되어 있다고 말할 수 있다. 파일 배치는, 이로 인해서 부하가 모든 장치들 간에 균등히 분산되어 있거나 또는 일부의 저장 장치들에 편중될 수 있기 때문에 전반적인 I/O 성능과 에너지 소모에 크게 영향을 미치므로 매우 중요하다. 한편, 버퍼 캐시 관리는 요청된 I/O들이 어떤 장치로 전달되어야 하며 얼마나 많은 I/O 요청들이 일정한 I/O로 전달되는지에 대한 필터링 효과 (filtering effect)를 가진다. 따라서 서로 다른 버퍼 캐시 관리 알고리즘은 서로 다른 I/O 요청 분산 형태를 취하게 된다.

본 논문에서는 파일 배치 기법과 버퍼 캐시 관리 기법을 결합하여 사용함으로써 시스템 전체의 부하가 어떻게 에너지와 성능 동시 개선을 위해 효율적으로 분산될 수 있는지를, 하드 디스크와 플래시 메모리를 가지는 이종의 모바일 저장 장치 시스템에 대해서 살펴보고자 한다.

IV. 에너지 효율적 부하 분산

I/O 요청의 접근 패턴 기반으로 동작하는 PB-PDC (pattern-based PDC) 기법은 빈번 접근 파일 집중 기법 (popular data concentration, PDC)을 확장 및 개선한 기법으로, 파일의 배치가 하드 디스크와 플래시 메모리로 구성된 이종의 저장 장치 시스템에 적합하도록 고안되었다 [1]. PDC는 Pinherio 등 [20]에 의해서 네트워크 서버들 상의 작업 부하에서 매우 편중되어 나타나는 파일별 접근 횟수들을 다루기 위해서 제안되었다. PDC의 기본 발상은 가장 빈번한, 즉 가장 자주 접근되는 파일들을 저장 장치들 중의 일부 장치로 집중시킴으로써 그 나머지 저장 장치들이 저전력 모드로 들어가서 에너지를 절감할 수 있도록 하는 것이다.

PDC는 동종의 (homogeneous) 저장 장치들 (즉, 디스크들)을 대상으로 파일 접근 횟수에만 기반하여 파일을 배치하는 기법인 반면에 PB-PDC는 각 장치별 동작 특성뿐만 아니라 접근 횟수에 함께 기반하여 파일을 이동 및 배치시키는 기법이다. 파일의 접근 횟수가 시간에 따라 변화하는 경우 PDC는 굉장히 많은 수의 파일 이동을 야기할 수 있으며 이것은 유휴의 디스크들을 동작 모드로 전환시킴으로써 에너지 소모를 많이 증가시키는 결과를 초래한다. PDC는 파일 접근 횟수에만 의존하여 파일을 이동시키므로 이종성 (heterogeneity)에 대한 고려가 없으며 결과적으로 자주 장치간 많은 파일 이동을 유발시킨다. PB-PDC는 PDC의 이러한 단점들을 극복하기 위해서 개발되었다 [1].

플래시 메모리 장치는 낮은 쓰기 처리량 (throughput)과 제한된 소거 횟수를 가지고 있으므로 PB-PDC는 이를 고려하여 빈번한 쓰기 횟수를 보이는 파일들을 디스크로 빈번한 읽기 접근의 파일들은 플래시 메모리로 이동시킨다. 같은 장치 상에서 빈번한 읽기와 쓰기 데이터간의 충돌이 발생하면 (즉, 같은 파일이 쓰기도 빈번하게 발생하고 읽기도 빈번하게 요청되면) 쓰기 데이터가 우선순위를 가진다. 이런 식으로 PB-PDC는 모다 효율적으로 부하 분산을 처리하게 된다.

PB-PDC는 각 파일별로 읽기 및 쓰기 계수기 (counter)를 유지하고 있으며 각 파일이 읽기 및 쓰기 접근이 될 때마다 접근 횟수를 갱신한다. 파일 접근이 읽기 (쓰기) 이면 읽기 (쓰기) 계수기가 1씩 증가된다. 읽기/쓰기 계수기는 파일 아이디 (ID), 파일 크기, 장치 식별자 (디스크인지 플래시인지를 구별하는)와 같은 파일 정보와 삼께 다중 큐 (multi-queue)에 의해 파일 접근 횟수의 감소 순서로 관리된다.

그림 1은 PB-PDC의 전반적으로 수행 흐름을 보여준다. 요청수가 파일 재배치 주기에 일치하게 되면 파일 이동 조건을 검사하여 이에 적합한 파일들을 장치 간에 이동하게 된다. 매 파일 재배치 주기마다 PB-PDC는 파일 이동 조건 확인을 위해서 2개의 다중 큐(읽기 큐와 쓰기 큐)들을 모두 가로질러 검사한다. 먼저, 쓰기 큐의 파일들을 검사하여 쓰기 계수가 큰 파일들부터 플래시에서 디스크로 이동시키는 시도를 하게 된다. 이때, 디스크의 여유 공간이 부족하면 옮길 파일의 크기보다 큰 파일이면서 쓰기 계수 값이 작은 파일이 있는지를 파악하고 플래시에 이 파일을 옮길 공간이 있는지를 확인한다. 이것이 가능하면, 디스크의 희생 파일을 먼저 플래시로 옮기고 원래 옮겨야 할 파일을 플래시에서 디스크로 옮기게 된다. 이러한 조건에 만족되지 않으면 아무런 파일 이동이 일어나지

않으며 다음의 작은 쓰기 계수를 가지는 파일들에 대해서 파일 이동을 시도하게 된다.

쓰기 큐에 대한 파일 재배치가 모두 끝나면 읽기 큐에 대한 파일 재배치에 대한 확인 및 파일 이동이 일어나게 된다. 읽기 파일 이동시에는 앞의 쓰기 파일 이동시와 동일한 알고리즘이 적용되는데 이에 더하여 잦은 쓰기 접근 때문에 이미 이동된 파일들을 제외하게 된다. 앞에서 언급한 바와 같이, PB-PDC는 자주 발생하는 쓰기 I/O 요청의 파일들에 우선순위를 주고 있다.

모든 파일 재배치가 끝나면, PB-PDC는 두 큐에 존재하는 모든 파일들에 대해서 읽기 및 쓰기 계수를 반으로 줄이는 작업을 수행한다. 이것은 모바일 작업 부하에서 파일들에 대한 읽 기 및 쓰기 접근이 시간에 따라 변화하므로 매번 파일 재배치 주기의 시작마다 계수들을 감소시킴으로써 해당 주기에서 파일 접근 횟수가 커지는 것을 반영하기 위함이다. 즉, 파일 접근 횟수의 변화에 따라서 더 최근의 접근 횟수들에 더 높은 가중치를 주어야 하기 때문이다.

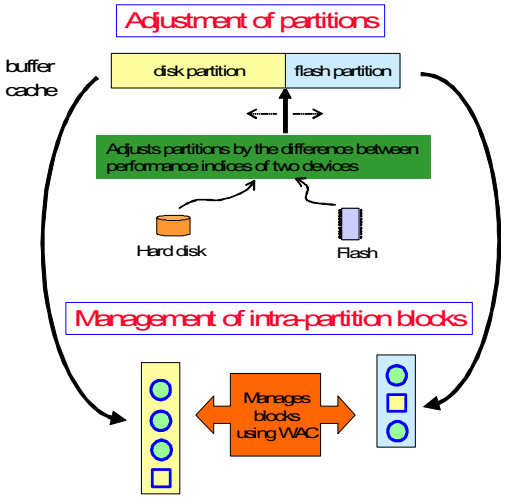

그림 2. DAC 알고리즘의 구조. DAC는 2 수준으로 구성됨: 1) 구획 조정 부분 2) 구획 내부의 블록 관리 부분
Fig. 2. Overview of DAC algorithm. DAC consists of two levels: 1) adjustment of partitions 2) management of intra-partition blocks

V. 성능 효율적 부하 분산

이종의 저장 장치들을 사용하는 경우, 파일 시스템은 파일 블록들이 속해 있는 장치들에 의존하여 블록에 대한 I/O 접근 시 달라지는 캐시 실패 지연 (cache miss penalties)을 고려하는 캐시 알고리즘이 필수적으로 요구된다. 하지만 기존 운영체제에서 널리 사용되어 오고 있는 LRU기법은 그러한 파일 접근간의 접근 비용 차이에 대한 고려가 없으며 모든 캐시내의 블록들에 대해서 같은 교체 (replacement) 비용을 가지는 것처럼 관리를 한다. GreedyDual-Size 알고리즘은 [21] LRU 알고리즘을 일반화하여 개선하기 위하여 지역성 (locality)을 캐시 실패 지연과 파일 크기 정보를 결합하고 있다.

디스크 기반의 저장 장치 시스템에 대해서, Forney 등은 동종의 디스크들을 대상으로 노후화 (aging) 등에 따른 이종성의 여지를 발견하고 저장 장치인지 (storage-aware) 기반의 캐시 관리 알고리즘을 연구하였다 [22]. 이 연구에서는 디스크별로 캐시 내에 구획 (partition)을 나누고 구획의 크기를 주기적으로 각 디스크별로 부하 수행에 따라 수행 시간에 기반하여 가변시키도록 하고 있다. 각 구획내의 관리는 GreedyDual-Size 알고리즘과 유사하게 수행한다. 이 연구의 목표는 모든 장치들 간에 일 또는 부하를 균등하게 분산시키는 것이다. 하지만, 저자들은 많은 수의 순차 접근 (sequential accesses)이 있는 경우에 대해서 제대로 살펴보고 대처하지 않고 있는데 멀티미디어 파일 재생 등으로 인해 순차 접근이 많은 모바일 부하에서는 문제를 야기하게 된다.

DAC는 하드 디스크와 플래시 메모리의 쌍과 같은 이종 저장 장치들을 가지는 모바일 컴퓨팅 시스템을 대상으로 제안된 동적 캐시 분할 기법이다 [6]. DAC의 목표는 1) 경험 기반의 동적인 분할 기법을 사용하여 캐시 구획을 조정하며 2) 각 구획 내에서는 디스크를 위한 순차 블록과 플래시 메모리를 위한 읽기 블록들을 빨리 축출(evict)함으로써 평균 I/O 응답 시간을 최소화하고자 하는 것이다.

DAC은 크게 2개의 부분으로 구성되어 있다. 첫째는 디스크와 플래시를 위한 캐시 구획의 크기를 조절하는 캐시 구획 조정 부분이며 둘째는 각 구획 내에서 작업 부하 패턴과 블록들의 I/O 타입 및 지역성을 함께 고려하여 블록을 축출 및 관리하는 구획 내부 캐시 관리 부분이다. 전자에서는 고정된 I/O 요청주기마다 누적된(접근 시간에 의해 가중화된) 캐시 참조 실패 횟수가 각 장치별로 어떻게 변화하는 지를 추적하며 현재 시스템이 어떤 양상(phase)에 있는지를 파악한다.

DAC의 두 번째 부분의 알고리즘을 설계하는데 있어서는 2개의 캐시 관리 정책이 도입되었다. 1) 각 구획의 크기는 순차 접근들이 임의 접근 블록 보다는 더 일찍 두 장치중의 하나로 향해져야 한다는 것이다. 2) 플래시 메모리 측의 캐시 구획내의 캐시 블록들에 대한 쓰기 I/O 요청들은 읽기 요청보다 더 늦게 플래시 메모리로 전달되어야 한다는 것이다. 이 두 정책에 기반하여 제안된 알고리즘이 구획 관리 알고리즘(intra-partition management algorithm)을 제시한다.

이 알고리즘은, GreedyDual-Size 알고리즘 [21]과 유사하

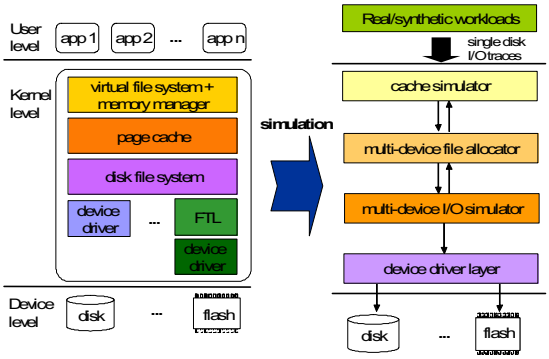

그림 3. 하드 디스크와 플래시 메모리의 쌍으로 구성된 이종 저장장치를 위한 시뮬레이션 구조
Fig. 3. Simulation architecture for a heterogeneous storage pair of a hard disk and a flash memory

지만 이를 보다 확장하여 디스크에 대해서는 임의성과 높은 시간 지역성을 지니는 블록들을, 플래시에 대해서는 역시 임의성과 높은 시간 지역성 및 쓰기 I/O 타입을 지니는 블록들을 각 캐시 구획 내에 더 오래 머무르도록 함으로써 모바일 작업 부하에서 자주 발견되는 많은 수의 순차 요청들이 적절히 다루어질 수 있도록 하고 있다. 또한, 플래시 메모리에 대해 지연된 쓰기 블록의 동작은 쓰기/소거 횟수를 줄임으로써 수명 주기를 더 늘이고 나아가 전체 저장 장치 시스템의 신뢰성을 높이는 것으로 기대된다. (자세한 알고리즘 내용은 [6]을 참고하기 바람.)

DAC의 강점은 이 알고리즘이 작업 부하의 특성뿐만 아니라 저장 장치의 특성을 고려하여 캐시를 관리하는데서 온다. 그림 2에서 보는 바와 같이, DAC은 캐시 구획 조정 부분에서 캐시 실패 및 캐시 실패 변화율과 장치별 I/O 접근 시간에 기반하는 성능 지표의 변화에 따라서 캐시 구획을 각 장치에 대해서 변화한다. 또, 구획 내부 캐시 관리 부분에서는 작업부하 인지 캐기 관리 (workload-aware cache management, WAC)를 시간 및 공간 지역성과 I/O 타입에 기반하여 각 구획 내에서 캐시 블록들을 관리한다. 이러한 메커니즘을 기반으로 DAC는 LRU와 비교하여 이종 저장 장치 시스템에 대하여 효율적으로 부하 분산을 다룸으로써 시스템 성능을 개선하는 것으로 나타나고 있다 [6].

VI. 실험 결과

1. 실험 환경

본 논문에서는 에너지 및 성능 동시 고려의 부하 분산 기법을 실험으로 평가하기 위하여, [1]의 다중 저장 장치 I/O 시뮬레이터를 기반으로 하여 트레이스 입력 동작 방식의 캐시 시뮬레이터를 구축하였다. 본 캐시 시뮬레이터에서는 LRU와 DAC을 수행할 수 있으며 이 버퍼 캐시 관리 기법들은 하드 디스크와 플래시 메모리 장치의 쌍에 대해서 파일 할당 및 이동을 수행하는 PB-PDC 모듈의 수행과 연계 동작하도록 하였다.

그림 3에서는, 왼쪽에서 대상으로 하는 모바일 시스템의 구조와 오른쪽에는 이를 모사하는 시뮬레이터 전체 구성을 보여주고 있다. 시뮬레이터에서 다중 장치 파일 할당자 모듈은 PB-DAC 알고리즘이 포함되어 수행되는 모듈로, 트레이스 들이 다중의 이종 저장 장치 시스템으로 전달될 때 이 장치들에 대해 정적인 파일 배치와 동적인 파일 이동을 모사한다. 이때, 정적인 파일 배치는 파일들이 디스크 접근이 일어나기 전에 디스크들 상에 분포되어 있도록 하는 방식을 언급한다.

또, 동적 파일 이동유발 조건이 만족하는 경우에 PB-PDC 알고리즘을 호출하여 파일 이동을 수행한다. 아래의 다중 장치 I/O 시뮬레이터는 각 장치에 대해 I/O 동작들이 분배가 되는 것을 모사한다. 이 모듈은 디스크 및 플래시 메모리에 대한 전력 및 성능 모델을 사용하여 각 저장 장치의 에너지 소모 및 성능을 추정한다. 본 실험에서의 측정 요소는 시스템 전체 에너지 소모 값과 평균 I/O 응답 시간이다. 사용된 하드 디스크의 모델은 1.8 ″, 4,200 RPM의 MK4004GAH [1]이며 플래시 메모리의 모델은 K9K1208U [13]이다.

모바일 트레이스를 생성하기 위하여 본 논문에서는 합성 트레이스 생성기[6]를 사용하였다. 이 생성기는 3가지 종류의 트레이스를 생성할 수 있는데 각각 파일들에 대한 순차 접근 패턴과 높은 시간 지역성을 가지는 접근 패턴, 그리고 앞의 두 패턴이 혼재한 접근 패턴을 가지고 있다 (각각 SEQ, TEMP, 그리고 COMPOUND임). 이 생성기는 또한 다양한 변수, 예를 들면 요청 발생율, 읽기/쓰기 비율, 파일 크기, 그리고 요청 크기 등을 조정할 수 있다. 사용된 기본 I/O 접근 종류는 COMPOUND로 사용 트레이스의 명칭은 trace1과 trace2로 trace2가 좀더 수행 시간이 길고 시간 지역성이 높다. DAC를 위해서 구획 조정 주기와 같은 가변 변수들은 [6]에 있는 값들을 사용하였다. 또, DAC의 구획 재조정 및 구획 내 블록 관리에 대한 오버헤드는 LRU에 비해서 적절한 수준이라고 가정하였다.

2. 실험 결과

PB-PDC와 DAC는 둘다 부하 분산을 수행하는 기법들이나 처음 고안 단계에서 그 목적이 다르고 완전히 독립적인 성능을 보이고 있다. 따라서 본 연구에서는 이 두 독립적인 부하 분산 기법을 결합하여 시스템 전체 부하를 에너지와 성능 측

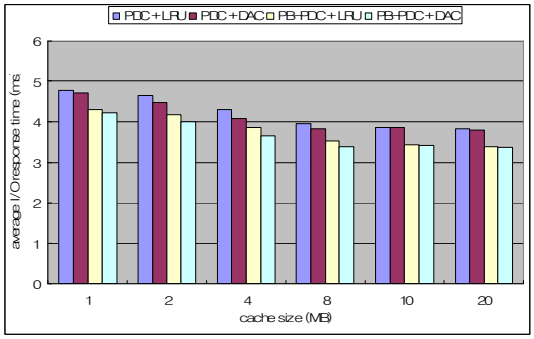

그림 4. 4가지 조합의 부하 분산 알고리즘의 평균 I/O응답 시간 (trace1)
Fig. 4. Average I/O response times of 4 combinatory pairs of load distribution algorithms (trace1)

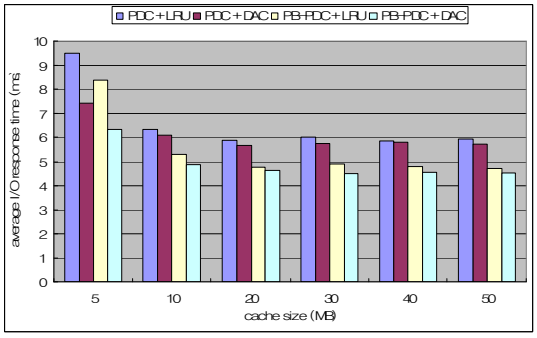

그림 6. 4가지 조합의 부하 분산 알고리즘의 평균 I/O응답 시간 (trace2)
Fig. 6. Average I/O response times of 4 combinatory pairs of load distribution algorithms (trace2)

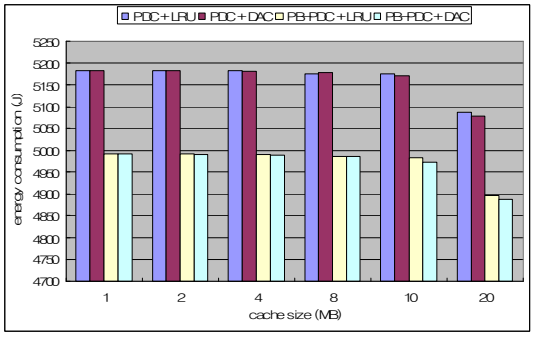

그림 5. 4가지 조합의 부하 분산 알고리즘의 에너지 소모 (trace1)
Fig. 5. Energy consumptions of 4 combinatory pairs of load distribution algorithms (trace1)

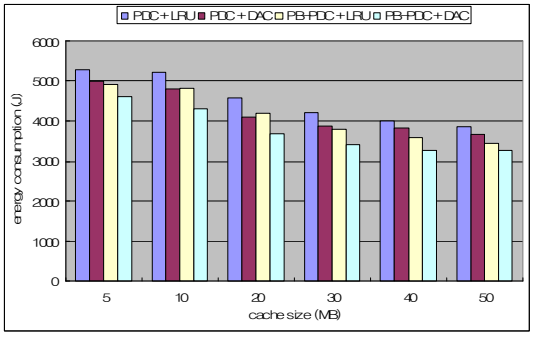

그림 7. 4가지 조합의 부하 분산 알고리즘의 에너지 소모 (trace2)
Fig. 7. Energy consumptions of 4 combinatory pairs of load distribution algorithms (trace2)

면에서 동시에 분산 및 배치하는 것을 실험을 통해서 살펴보고자 한다.

본 논문에서는 다양한 실험을 통하여 제안한 에너지 및 성능 동시 제어 부하 분산 방법이 얼마나 효율성을 보이는지를 평가하였다. 실험에서는 앞에서 기술한 시뮬레이션 환경을 사용하여 trace1과 trace2 2가지에 대해서 버퍼 캐시 기법 2가지, LRU와 DAC를 파일 배치 기법 2가지, PDC와 PB-PDC를 각각 결합하여 총 4가지의 부하 분산 방법에 대해서 평가를 수행하였다. 각각의 방법들은 다음과 같이 명명한다: PDC+LRU, PDC+DAC, PB-PDC+LRU, PB-PDC+DAC.

그림 4와 6은 블록당 평균 I/O 응답 시간을 위 4가지 부하 분산 조합에 대해서 trace1과 trace2에 대해서 각각 보여주고 있다. 그리고 그림 5와 7은 마찬가지로 trace1과 trace2를 각각 사용하였을 때 4가지 조합에 대해서 전체 에너지 소모 값을 보여주고 있다.

그림 4에서 DAC는 각 캐시 크기에 대해서 LRU보다 성능, 즉 평균 I/O 응답 시간에서 더 빠름을 확인할 수 있다. 파일 배치가 PDC나 PB-PDC로 고정된 경우에도 최고 약 5.4%정도까지 개선하는 것으로 파악되었다. 즉, 이러한 성능 개선은 파일 배치와 관계없는 것으로 나타났으며 에너지 절감과는 관계없이 성능 개선에 독립적인 효과를 나타낸 것으로 분석된다. 게다가 PB-PDC+DAC는 가장 빠른 평균 I/O 응답 시간을 나타내고 있는데 기존 기법들의 결합 방법인 PDC+LRU에 대해서 12-15%나 성능 개선이 있는 것으로 파악되었다.

유사하게, 그림 6에서도 PB-PDC+DAC가 trace2에 대해서 가장 빠른 평균 I/O 응답 시간을 보이고 있다. 특히 PDC+LRU에 대해서는 22-33%의 성능 개선을 보이고 있다.

주목할 사항은, DAC가 기존의 파일 배치 기법인 PDC와 결합 될 때보다 파일 접근 타입 및 접근 횟수를 기반으로 하는 배치기법인 PB-PDC와 결합되었을 때가 훨씬 더 많은 성능 개선이 얻어진다는 사실이다. 그림 4에서는 PB-PDC+DAC가 PDC+DAC에 대해서 모든 캐시 크기에 대해서 평균적으로 10.5%의 성능 개선이 있으며 최고 12%의 개선을 보였다. 그림 5에서는 PB-PDC+DAC가 PDC+DAC에 대해서 평

균적으로 약 26%와 최고 35%의 성능 개선을 보이는 것으로 나타났다.

이 결과들로부터, 본 연구에서는 DAC가 가지는 독립적인 성능 개선 작용이 PB-PDC의 결합으로 인해 부하 분산의 측면에서 보다 상승된 개선 효과를 보이고 있음을 확인할 수 있었다. 즉, PB-PDC가 에너지 측면이 아닌 성능 측면에서도 지능적인 파일 배치를 통해서 뛰어난 역할을 하고 있음을 알아챌 수 있다.

비슷한 결과가 그림 5와 그림 7에서의 에너지 소모 값에서도 발견되어 졌다. 그림 5에서 PB-PDC+DAC는 모든 캐시 크기에 대해서 모든 부하 분산 조합 중에서 가장 낮은 에너지 소모를 나타내고 있음을 확인할 수 있었다. PDC+LRU와 비교하면 최고 19%의 에너지 절감을 보이고 있다. 따라서 DAC가 성능 측면뿐만 아니라 에너지 측면에서 LRU와 비교하여 PB-PDC와 결합됨으로써 에너지 소모 절감에 더 좋은 영향을 끼치는 것으로 파악되었다.

요약하면 이러한 결과들로부터, 본 논문에서 제안한 방법은 기존의 기법들이 독립적인 성능을 가지던 것에 비해 부하 분산의 관점에서 결합된 형태를 통해 보다 상승적인 에너지와 성능 측면의 개선 효과를 얻는 다는 것을 알 수 있었다. 또, 제안 기법은 점점 더 사용이 많아지고 있는 하드 디스크와 플래시 메모리의 이종 저장 장치 시스템에서의 실용적인 부하 분산을 다루고 있다는 점이 중요하다고 말할 수 있다.

VII. 결 론

본 논문에서는 운영체제 수준에서 에너지와 성능을 동시에 개선하고자 하는 목적으로 소형 하드 디스크와 플래시 메모리를 이종의 저장 장치로 가지는 모바일 시스템에 적합한 동적 부하 분산 기법을 제안하였다. 이러한 목적으로 본 논문에서는 에너지 효율적인 파일 배치 기법과 성능 효율적인 버퍼 캐시 관리 기법을 결합하는 접근법을 취하였으며 시뮬레이션을 통해서 기존의 기법들과 비교하여 이종의 모바일 저장장치들에 대해서 에너지와 성능에 대해서 동시에 개선된 결과를 보이는 것을 확인하였다.

이종의 저장 장치 시스템에서의 부하 분산은 때때로 최적의 전체 I/O 성능을 얻기 힘든 경우가 발생하는 것을 확인하였다. 이것은 다양한 이종 저장 장치들이 서로 다른 I/O 동작 특성을 가지기 때문이다. 따라서 향후 연구로는, 부하 분산이 에너지 절감과 관련하여 전체 시스템 성능에 어떠한 영향을 주는지를 더 깊이 조사할 계획이다. 더불어 실제 운영체제 시스템에서의 구현에 따른 평가 또한 중요한 향후 연구 계획에 포함된다.

참고문헌

[1] Y.-J. Kim, K.-T. Kwon, and J. Kim, "Energy-efficient file placement techniques for heterogeneous mobile storage systems," in Proc. of the 6th ACM & IEEE Conference on Embedded Software (EMSOFT), Seoul, Korea, October 22-25 2006.

[2] F. Wang, B. Hong, S. A. Brandt, and D. D. E. Long, "Using MEMS-based storage to boost disk performance," in Proc. of 22nd IEEE / 13th NASA Goddard Conference on Mass Storage Systems and Technologies (MSST 2005), Monterey, CA, USA, April 2005.

[3] New Toshiba mobileLBA-NAND memory chips for mobile phones support both SLC and MLC memory areas. *http://www.toshiba.com/taec/news/press_releases/2007/memy_07_482.jsp*

[4] G. Lawton, "Improved Flash Memory Grows in Popularity," IEEE Computer, vol. 39, no. 1, pp. 16-18, Jan. 2006.

[5] E. Pinheiro, R. Bianchini, E. V. Carrera, and T. Heath, "Load balancing and unbalancing for power and performance in cluster-based systems," in Proc. of the International Workshop on Compilers and Operating Systems for Low Power, September 2001.

[6] Y.-J. Kim and J. Kim, "Device-aware cache replacement algorithm for heterogeneous mobile storage devices," in Proc. of the 3rd International Conference on Embedded Software and Systems (ICESS), Daegu, Korea, May 14-16, 2007. Lecture Notes in Computer Science (LNCS), Vol. 4523, pp. 13-24, May 2007.

[7] B. Marsh, F. Douglis, and P. Krishnan, "Flash memory file caching for mobile computers," in Proc. of the 27th Hawaii International Conference on System Sciences, Hawaii, USA, pp. 451-460, January. 1994.

[8] T. Bisson and S. Brandt, "Reducing energy consumption with a non-volatile storage cache," in Proc. of International Work-shop on Software Support for Portable Storage (IWSSPS), held in conjunction with the IEEE Real-Time and Embedded Systems and Applications Symposium (RTAS 2005), San Francisco, California, USA, March, 2005.

[9] *http://shopping.msn.com.*

〔10〕 *http://www.inspectrumtech.com.*
〔11〕 Hitachi GST, Travelstar 80GN. *http://www.hitachigst.com/tech/techlib.nsf/products/Travelstar_80GN.*
〔12〕 Toshiba, MK4004GAH. *http://www3.toshiba.co.jp/storage/english/spec/hdd/mk4004gs.htm.*
〔13〕 H. G. Lee and N. Chang, "Low-energy heterogeneous non-volatile memory systems for mobile systems," Journal of Low Power Electronics, Vol. 1, Number 1, pp. 52-62, April, 2005.
〔14〕 F. Chen, S. Jiang, and X. Zhang, "SmartSaver: turning flash drive into a disk energy saver for mobile computers," in Proc. of 11th ACM/IEEE International Symposium on Low Power Electronics and Design (ISLPED'06), Tegernsee, Germany, October 4-6, 2006.
〔15〕 T. Kgil and T. Mudge, "FlashCache: A NAND flash memory file cache for low power web servers," in Proc. of the 2006 International Conference on Compilers, Architecture and Synthesis for Embedded Systems (CASES '06), Seoul, Korea, October 22-25, 2006.
〔16〕 T. Bisson, S. Brandt, and D. Long, "A hybrid disk-aware spin-down algorithm with I/O subsystem support," in Proc. of the 26th IEEE International Performance Computing and Communications Conference (IPCCC), New Orleans, Louisiana, USA, April 11-13, 2007.
〔17〕 Microsoft, ReadyDrive and Hybrid Disk. *http://www.microsoft.com/whdc/device/storage/hybrid.mspx.*
〔18〕 R. Panabaker, "Hybrid hard disk & ReadyDriveTM technology: improving performance and power for Windows Vista mobile PCs," in Proc. of Microsoft WinHEC 2006. *http://www.microsoft.com/whdc/winhec/pres06.mspx.*
〔19〕 M. Trainor, "Overcoming disk drive access bottlenecks with Intel Robson technology," Technology®Intel Magazine, December, 2006. *http://www.intel.com/technology/magazine/computing/robson-1206.htm.*
〔20〕 E. Pinheiro and R. Bianchini, "Energy conservation techniques for disk array-based servers," in Proc. of the 18th International Conference on Supercomputing (ICS'04), June 2004.
〔21〕 P. Cao and S. Irani, "Cost-aware WWW proxy caching algorithms," in Proc. of USENIX Symposium on Internet Technology and Systems, December, 1997.
〔22〕 B. Forney, A. C. Arpaci-Dusseau, and R. H. Arpaci-Dusseau, "Storage-aware caching: revisiting caching for heterogeneous storage systems," in Proc. of the 1st.USENIX Conference on File and Storage Technologies (FAST), Jan. 2002.

저 자 소 개

김 영 진

1999년 서울대학교 전기공학부 석사
1999년-2003년 한국전자통신연구원 연구원
2003년-2008년 서울대학교 전기·컴퓨터공학부 박사
2008년-현재 선문대학교 컴퓨터공학부 전임강사
관심분야 : 임베디드 시스템 및 소프트웨어, 저전력 소프트웨어 기법, 모바일 저장 장치 시스템 및 기법, 성능 및 전력 분석 도구

김 지 홍

1988년 University of Washington 컴퓨터과학과 석사
1995년 University of Washington 컴퓨터과학 및 공학과 박사.
1995년-1997년 미국 Texas Instruments 선임연구원.
1997년-현재 서울대학교 전기·컴퓨터공학부 교수.
관심분야 : 임베디드 소프트웨어, 저전력 시스템, 멀티미디어 시스템, 컴퓨터 구조