

SSD 기반 디스크 시스템과 RAM 버퍼 기반 원격 디스크 시스템 성능평가

구본근*

*한국교통대학교 컴퓨터공학과

bggooo@ut.ac.kr

Disk Performance Evaluation of SSD and RAM Buffer-based Remote Disk System

Bon-Gen Gu*

*Dept. of Computer Engineering, Korea National University of Transportation

요약

본 논문은 원격 서버에 저장 내용을 백업하는 RAM 기반 디스크 시스템인 NBRD의 프로토타입 구현을 위한 블록 장치 구동기와 이를 이용한 디스크 입출력 성능을 평가한다. 또, SSD의 성능 평가를 실시하여 향후 NBRD의 이점 및 향후 성능 향상 목표값을 설정한다.

에서 성능을 평가한다. 또, 고성능 SSD의 성능을 평가하여 향후 NBRD의 성능 목표를 설정한다.

1. 서론

기계학습, 빅데이터 처리 등 대용량의 데이터에 신속하게 접근, 처리해야 하는 응용 분야가 확대됨에 따라 고성능의 저장 장치에 대한 요구가 증가하고 있다. 분산 파일 시스템은 네트워크에 연결된 노드를 이용하는 것으로 디스크 IO와 네트워크 IO가 동시에 발생하기 때문에 이에 따른 지연시간 감소를 위해 디스크와 네트워크 IO가 프로세서 친화도 특성을 활용하고 있다[1]. POPA 시스템은 디스크 시스템 풀을 구성하여 필요에 따라 노드에 디스크를 할당하여 디스크 IO 시간을 감소시키고 있다[2]. 자기 디스크 기반의 시스템의 탐색 시간 및 회전 지연 시간으로 인한 성능 한계를 극복하기 위해 RAM의 일부 영역을 디스크로 사용하는 연구도 있다[3,4].

RAM 디스크의 디스크 IO 성능은 높은 반면 전원이 단절되면 저장 내용을 잃어버리는 문제가 발생한다. 이러한 문제를 해결하기 위해 RAM을 기반으로 한 다중 디스크 IO 시스템이 제안되었다[4].

본 논문에서는 RAM 디스크를 이용하여 디스크 접근 시간을 감소시킴과 동시에 디스크 쓰기 요청은 원격 파일 서버로 전송하여 전원 단절 등으로 RAM 디스크의 내용이 소실되면 복구할 수 있는 NBRD(Network backup-based RAM Disk)의 프로토타입을 구현하고, SBC(single board computer) 상

2. NBRD 프로토타입 구현

NBRD는 RAM을 기반으로 한 디스크 시스템으로 전원 단절 등으로 인해 저장 내용의 소실을 방지하기 위해 디스크 쓰기 요청은 원격지의 파일 서버에 전송한다. 이를 처리하기 위해 리눅스 블록 장치 구동기의 디스크 입출력 요구를 처리하는 부분의 기본 알고리즘은 그림 1과 같다.

```
algorithm: blk_req_proc(req, blk)
  if req.type = read :
    copy RAM block to blk
  else if req.type = write :
    copy blk to RAM block
    transfer blk to remote server
```

(그림 1) 디스크 IO 요청 알고리즘.

서버에 저장된 RAM 디스크의 저장 내용은 NBRD 시스템 모듈이 시스템에 적재될 때 RAM 디스크 생성과 함께 그 내용을 읽어 복구한다. 그림 2는 NBRD 모듈이 적재될 때 초기화 과정에서 수행되는 것은 나타낸 것이다.

```

algorithm: module_init
    send id, n of blk to server
    if NBRD blk exist
        recover process from server
(그림 2) 디스크 IO 요청 알고리즘.

```

본 논문에서 NBRD 프로토타입은 6개의 ARM 코어 프로세서와 4GB 메모리의 SBC(하드커널사의 N2)와 리눅스 커널 4.9 기반 시스템에서 구현하였다. 이 시스템의 스토리지는 64GB의 eMMC 모듈이며, 비교 대상이 되는 SSD는 다른 고성능 시스템에 장착된 M.2 PCIe Gen3 기반 SSD(250GB)를 사용하였다.

성능평가는 리눅스 시스템의 dd 명령어를 이용하였으며, 그림 3에 나타낸 스크립트를 이용하여 RAM 디스크, NBRD 프로토타입, eMMC, SSD의 읽기, 쓰기 성능을 평가하였다. 성능은 최대, 최소 성능을 제외한 나머지의 평균을 이용하여 분석하였다.

```

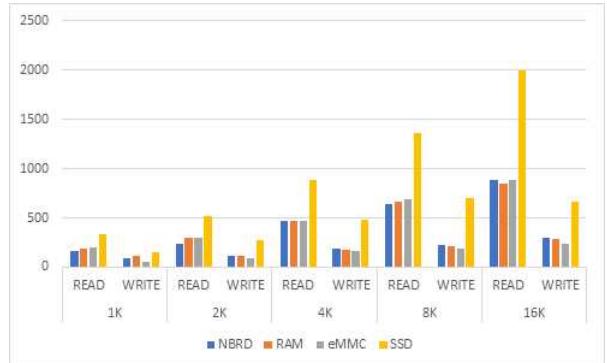
#!/bin/bash
iter="1 2 3 4 5"
rep="1 2 4 8 16 32 64 128 256 512 1024"
for j in ${iter}
do
    for i in ${rep}
    do
        dd if=/dev/zero of=file bs=${i}k count=1000
        dd if=file of=/dev/null bs=${i}k count=1000
    done
done
(그림 3) 디스크 IO 성능 평가를 위한 스크립트

```

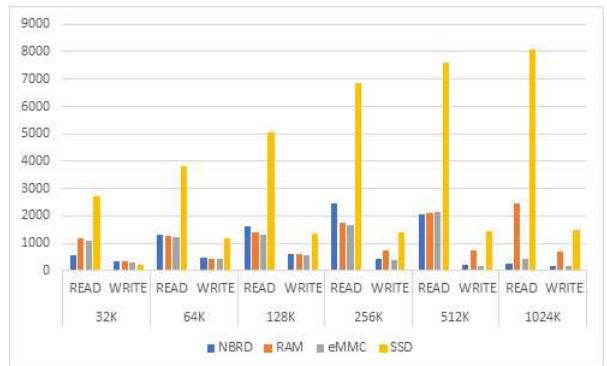
3. 성능 평가 결과

그림 4는 한 번에 읽기 또는 쓰기 데이터의 크기를 1K부터 16K까지 변화를 줄 때 전송률을 MB/S 단위로 나타낸 것이며, 그림 5는 32K부터 1024K까지 변화를 주었을 때 전송률을 나타낸 것이다. NBRD는 디스크 쓰기 요청시 네트워크 전송이 발생하므로 전반적으로 읽기 요청보다 전송률이 떨어진다. 하지만 RAM을 이용한 디스크 서비스 특성으로 인해 한 번에 읽기 또는 쓰기 데이터 크기에 따른 성능 차이가 상대적으로 적지 않음을 알 수 있다. 따라서, 디스크 서비스 성능이 응용 시스템의 디스크 읽기/쓰기 비율에 상대적으로 적게 영향을 받을 것으로 예상된다. 반면 SSD는 전반적으로 매우 높은 성능을 보이고 있지만 읽기 또는 쓰기 데이터 크

기가 증가할수록 그 성능 차이가 급격하게 증가함을 알 수 있다. 따라서, SSD는 응용 프로그램의 읽기/쓰기 비율에 따라 기대할 수 있는 성능은 다소 차이가 있을 것으로 판단된다.



(그림 4) 디스크 I/O성능(블록크기: 1K ~ 16K)



(그림 5) 디스크 I/O성능(블록크기: 32K ~ 1024K)

NBRD, RAM 디스크, eMMC 및 SSD의 성능평가 결과는 본 논문에서 제안하는 NBRD가 고성능 디스크 서비스를 위해 활용될 수 있음을 보이고 있다.

4. 결론

본 논문에서 구현한 NBRD는 시스템 주기억장치의 일부 영역을 RAM 디스크로 활용하여 서비스하고, 다양한 사고상황에서 손실되는 RAM 디스크 블록의 복구를 위해 쓰기 요청을 서비스할 때 별도의 서버에 그 블록을 전송한다. 비교적 저성능의 SBC에서 구현한 NBRD 시스템의 성능평가 결과는 아직 최적화가 되어 있지 않음에도 그 가능성을 확인할 수 있었다.

현재 NBRD의 구현은 최적화가 되어 있지 않으며, 시스템에 설치되어 있는 RAM의 일부 영역을 사용하기 때문에 디스크 용량의 확장에 한계있으며,

최초 NBRD 구성시 서버로부터 모든 블록을 읽어와야 하는 결정적 단점을 갖고 있다. 구현의 최적화는 지속적으로 개선을 하면 해결될 것으로 예상되며, 추후 연구과는 용량 확장 문제 및 NBRD 구성시 모든 블록을 읽어 와야 하는 문제를 해결하기 위해 RAM 디스크를 일종의 캐시로 사용하도록 블록 장치 구동기를 재설계하는 것이다.

감사의 글

2020년 한국교통대학교 지원을 받아 수행하였음.

참고문헌

- [1] J. Y. Cho and H. W. Jin, "Impact of Processor Affinity on Disk and Network I/O Performance", The 40th Fall Conference Proceeding of The Korean Institute of Information Scientists and Engineers, pp. 1197–1199, Nov. 2013.
- [2] S. C. Kim, H. D. Park, J. Y. Lee, C. J. Park, C. S. Jeong, J. H. Han, and J. Y. Lee, "An Implementation of a Disk Access Server In the POPA Parallel Computer", Fall Conference Proceeding of The Korean Institute of Information Scientists and Engineers, pp. 507–510, Oct. 1989.
- [3] T. Y. Son, "Design and Implementation of Block Device Driver for Accessing the NRD", Thesis of Master's Degree, The Graduate School of Hoseo University, Dec. 2012.
- [4] B. G. Gu, "Multi Disk I/O Model using Duplicated Disk I/O Packet Transferring," The Proceedings of the 2020 KIIT Conference, Vol.15, NO.1, pp.19–21, Doc. 2020