

# 하이브리드 폴링 기법의 I/O 응답 시간 예측 성능 분석

정지현<sup>o</sup> 윤여환 서동주 주용수 임성수 임은진

국민대학교 소프트웨어학부

{jungjh, johnnyhgmg, commisori, ysjo, sslim, ejim}@kookmin.ac.kr

## Analyzing the Accuracy of Hybrid Polling in Predicting I/O Response Time

Jihyeon Jung<sup>o</sup>, Yeohwan Yoon, Dongjoo Seo, Yongsoo Joo, Sung-Soo Lim, Eun-Jin Im

School of Computer Science, Kookmin University

### 요약

저지연 저장장치를 대상으로 인터럽트 대비 I/O 응답 시간을 개선하면서 폴링 대비 낮은 CPU 점유율을 가지는 하이브리드 폴링 기법이 제안되었다. I/O 응답 시간 및 CPU 점유율 분석을 통한 하이브리드 폴링의 성능평가 연구는 활발하게 수행된 반면 하이브리드 폴링의 핵심 요소인 I/O 응답 시간 예측 알고리즘 자체에 대한 성능 분석은 아직 미비하다. 본 연구에서는 하이브리드 폴링의 I/O 응답 시간 예측 성능 분석을 위한 측정도구를 구현하고 백그라운드 I/O가 과도하게 발생하는 상황에서 I/O 응답 시간 예측 정확도가 떨어지는 비효율성을 확인하였다.

## 1. 서론

저장장치의 느린 I/O 응답 시간은 컴퓨팅 시스템의 성능을 떨어뜨리는 주요 원인이었다. 하지만 최근 들어 수십 마이크로초 이하의 지연시간을 가지는 저지연 SSD가 등장하는 등 저장장치의 응답 속도가 빠르게 향상되고 있다. 한편 NVMe 기반 SSD에 대해 인터럽트 기법을 사용했을 때 문맥 교환 오버헤드의 비중이 지나치게 커지는 문제가 지적되었다. 따라서 이를 극복하기 위해 인터럽트 대신 폴링을 사용할 것을 제안한 연구 결과[1]가 발표되었다. 하지만 폴링 기법을 사용하는 경우 CPU 점유율이 100%에 달하는 문제가 있다. 이를 완화시키기 위해 인터럽트 기법과 폴링 기법의 장점을 통합한 하이브리드 폴링 기법이 제안되었다[2].

하이브리드 폴링은 직전에 발생한 I/O들의 응답 시간을 크기별로 분류하고 각 크기별 평균값으로 현재 처리하는 I/O의 응답 시간을 예측한다. 실제 구현은 I/O 요청 직후 예측한 응답 시간의 50%만큼 sleep한 후 깨어나 I/O 완료 시점까지 폴링을 수행한다. 따라서 폴링 기법에 비해 sleep한 시간만큼 CPU 점유율을 낮추면서도 인터럽트 대비 빠른 I/O 처리속도를 기대할 수 있다.

하지만 하이브리드 폴링이 성능을 발휘하기 위해서는 I/O 응답 시간 예측이 정확해야 한다. I/O 응답 시간을 실제보다 짧게 예측할 경우 CPU 점유율을 충분히 낮출 수 없다. 반면 I/O 응답 시간을 실제보다 크게 예측할 경우 I/O가 완료되었음에도 불구하고 여전히 sleep에서 깨어나지 않아 I/O 응답 시간이 실제보다 늘어난다.

\* 이 논문은 2021년도 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원의 지원을 받아 수행된 연구임 (No. 2014-0-00035, 매니코어 기반 초고성능 스케일러블 OS 기초연구)

\* 이 성과는 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(No.2018R1D1A1B05044558)

하이브리드 폴링의 성능 평가를 위한 기존 연구[3][4][5][6][7]에서는 하이브리드 폴링의 I/O 응답 시간 및 CPU 사용률을 주요 평가 지표로 삼았다. 여기서 I/O 응답 시간은 저장장치에서 실제로 소요된 I/O 처리 시간이 아니라 I/O를 요청한 프로세스의 입장에서 I/O를 요청한 후 완료될 때까지 걸린 전체 시간이다. 이를 통해 하이브리드 폴링의 성능을 평가할 경우 I/O 응답 시간 예측 실패로 인한 시간 지연과 저장장치 내부에서의 I/O 처리 지연을 구분할 수 없다. 즉 하이브리드 폴링의 I/O 응답 시간 예측 알고리즘 성능을 직접 평가하는 데 한계가 있다.

본 연구에서는 하이브리드 폴링 기법의 I/O 응답 시간 예측 정확도를 분석하기 위한 측정 도구를 구현하였다. 구현한 도구를 사용하여 백그라운드 I/O 프로세스가 많은 양의 I/O를 발생시키는 상황에서 하이브리드 폴링의 I/O 응답 시간 예측 실패율이 30% 수준으로 높아지는 비효율성을 확인하였다.

## 2. I/O 응답 시간 예측 성능 분석 도구

리눅스 커널의 block/blk-mq.c 파일을 수정하여 두 곳의 측정 지점을 추가하였다. 첫 번째 지점에서는 하이브리드 폴링의 응답 시간 예측에 따른 sleep 함수의 실행이 끝나기 전에 I/O가 먼저 완료되는지의 여부를 확인하는 기능을 구현하였다. 두 번째 지점에서는 하이브리드 폴링 알고리즘이 I/O 응답 시간을 예측하여 실제로 적용한 유효 sleep 시간을 측정하도록 구현하였다.

첫 번째 측정지점에서는 sleep이 종료된 후 blk\_poll 함수의 폴링을 개시하기 전에 이미 I/O가 완료되었을 경우 하이브리드 폴링 기법의 예측 알고리즘 miss로 간주하였다. 따라서 sleep 종료 시 이미 완료된 I/O의 개수와 총 I/O 개수 사이의 비율을 miss rate로 정의하였다.

두 번째 측정지점에서는 blk\_mq\_poll\_hybrid\_sleep 함수에서 blk\_mq\_poll\_nsecs 함수 호출을 통해 얻은 결과 혹은 미

리 정해진 nsecs 값을 모니터링하여 누적하였다. 또한 누적된 값을 해당 측정 구간 동안 하이브리드 폴링이 처리한 I/O의 총 개수로 나누어 sleep 시간의 평균값을 계산하였다.

또한 측정 단위 구간을 정의하고 매 단위 구간이 종료될 때마다 각 측정지점의 누적 값을 초기화하여 시간의 흐름에 따른 miss rate 및 유효 sleep 시간의 변화를 추적할 수 있도록 하였다.

### 3. I/O 응답 시간 예측 성능 분석

본 연구에서는 백그라운드 I/O가 하이브리드 폴링의 I/O 응답 시간 예측 성능을 저하시킬 것이라는 가설을 세우고 이를 검증하기 위하여 표 1의 실험 환경을 준비하였다. 하이브리드 폴링 기법에 적합한 저지연 SSD인 983ZET SSD를 사용하였으며 실험용 I/O 발생을 위해 fio 벤치마크 도구를 사용하였다.

OS	Ubuntu 18.04
Kernel	Linux 5.11
CPU	Intel Xeon Gold 6230
Memory	192GB
NVMe SSD	Samsung 983 ZET 480GB
I/O generator	fio[8] 3.26

표 1: 실험 환경

백그라운드 I/O의 유무 및 강도에 따른 하이브리드 폴링의 응답 시간 예측 정확도 평가를 위하여 표 2와 같이 세 종류의 워크로드를 준비하였다. 포그라운드 I/O는 pvsync2 엔진을 이용하여 하이브리드 폴링으로 처리하고, 백그라운드 I/O는 libaio 엔진을 이용하여 하이브리드 폴링을 거치지 않도록 설정하였다. 본 실험에서 사용한 I/O의 유형은 4KB 임의 읽기로 고정하였다.

Job	Time (sec)	포그라운드 I/O	백그라운드 I/O
Job1	0~20	pvsync2 1core	X
Job2	20~40	pvsync2 1core	libaio 1core
Job3	40~60	pvsync2 1core	libaio 4core

표 2: 포그라운드 I/O와 백그라운드 I/O로 구성된 워크로드 명세

첫 번째 실험의 결과는 그림 1과 같다. Job1을 실행한 첫 20초 구간에서는 miss rate가 초반에 순간적으로 올라갔다가 곧바로 안정화되어 계속 0을 유지하는 것을 확인하였다. 백그라운드 I/O가 추가된 Job2를 실행한 20~40초 구간에서는 IOPS의 저하는 심하지 않으나 miss rate가 계속 10% 이상으로 측정되었다. 백그라운드 I/O의 강도가 높아진 Job3를 실행한 40~60초 구간에서는 miss rate가 80% 이상으로 치솟은 후 안정화된 후에도 약 30% 선을 유지하

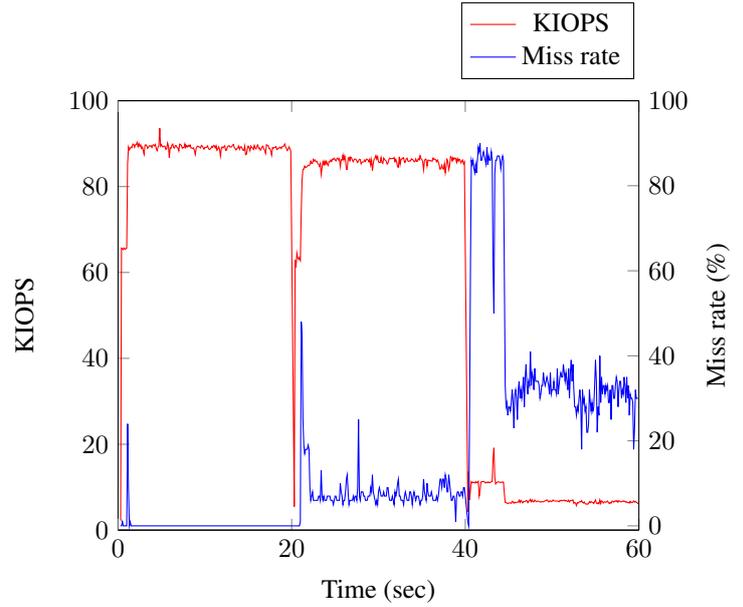


그림 1: 백그라운드 I/O에 따른 포그라운드 프로세스의 IOPS 및 하이브리드 폴링 예측 실패율 (miss rate) 변화

였다. 또한 IOPS도 Job1 대비 1/10 정도로 떨어지는 것으로 측정되었다.

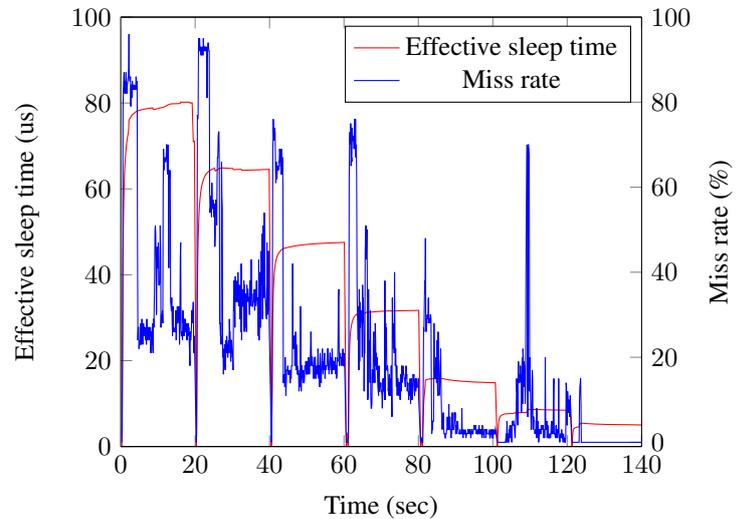


그림 2: 하이브리드 폴링의 sleep 비율에 따른 유효 sleep 시간 및 예측 실패율 변화

두 번째 실험으로 백그라운드 I/O의 강도가 높은 상황에서 sleep 비율을 커널 기본 설정치인 50%에서 시작하여 점점 줄여가면서 miss rate 및 실제로 적용된 유효 sleep 시간의 변화를 관찰하였다. 그림 2는 좌측부터 50%, 40%, 30%, 20%, 10%, 5%, 3%의 sleep 비율이 각 20초씩 적용된 결과를 보여준다. Sleep 비율을 줄임에 따라 유효 sleep 시간 및 miss rate가 점차 낮아졌지만 sleep 비율을 5%까지 낮추었음에도 불구하고(100-120초 구간) 상당한 수준의 miss가 발생하였다. Miss rate를 Job1 수준에 가깝게 억제하기 위해서는 sleep 비율을 3%까지 줄여야 하는데(120-140초 구간), 이

경우 하이브리드 폴링의 장점인 CPU 점유율 감소 효과가 크지 않을 것으로 예상되었다.

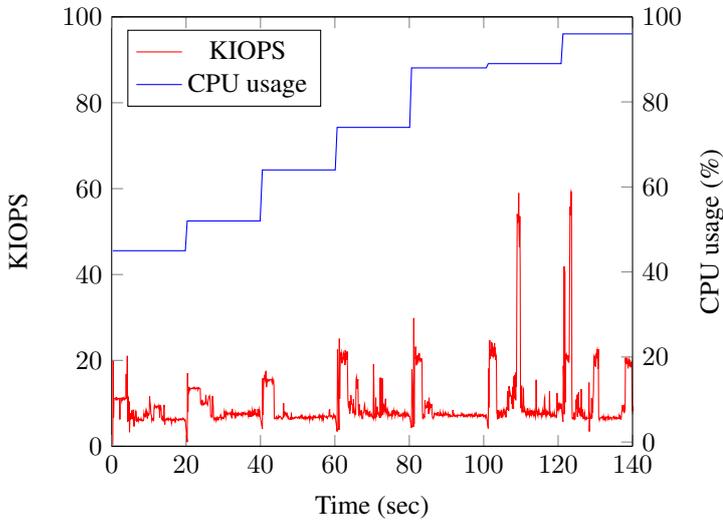


그림 3: 하이브리드 폴링의 sleep 비율에 따른 포그라운드 프로세스의 IOPS 및 평균 CPU 점유율 변화

두 번째 실험과 동일한 설정에서 IOPS와 CPU 점유율을 측정하여 그림 3에 제시하였다. Sleep 비율이 50%인 0-20초 구간에서는 CPU 점유율이 50% 미만으로 측정되었지만, sleep 비율이 3%인 120-140초 구간에서는 95% 이상의 높은 CPU 점유율을 보여 예상대로 하이브리드 폴링의 효과가 크지 않음을 확인하였다. 또한 sleep 비율을 3%까지 낮추었음에도 불구하고 IOPS의 순간 최댓값은 증가하지만 구간 평균값에는 큰 변화가 없는 현상 또한 확인하였다.

#### 4. 결론 및 향후 연구

본 논문에서는 하이브리드 폴링의 I/O 응답 시간 예측 성능을 상세하게 분석하기 위한 측정 도구를 구현하고 이를 이용하여 백그라운드 I/O의 간섭 하에서 하이브리드 폴링의 I/O 응답 시간 예측 정확도를 분석하였다. 실험 결과 과도한 백그라운드 I/O가 발생하는 상황에서는 하이브리드 폴링의 sleep 비율 설정을 어떻게 변경하더라도 miss rate와 CPU 점유율을 동시에 낮추는 것은 불가능함을 확인하였다. 향후 연구로 백그라운드 I/O 하에서도 miss rate와 CPU 점유율을 동시에 낮출 수 있도록 하이브리드 폴링의 I/O 응답 시간 예측 알고리즘을 개선하고자 한다.

#### 참고 문헌

[1] J. Yang, D. B. Minturn, and F. Hady, "When Poll is Better than Interrupt," in *10th USENIX Conference on File and Storage Technologies (FAST'12)*, vol. 12, pp. 3-3, 2012.

[2] D. L. Moal, "I/O Latency Optimization with Polling," in *Proc. of USENIX Conference of Linux Storage and Filesystems Conference (VAULT)*, pp. 1-25, 2017.

[3] Y. Song and Y. Eom, "HyPI: Reducing CPU Consumption of the I/O Completion Method in High-Performance Storage Systems," in *International Conference on Ubiquitous Information Management and Communication*, pp. 646-653, 2019.

[4] 천명준, 한상욱, 김지홍, "초저지연 저장장치를 위한 적응형 폴링 선택 기법," *한국정보과학회 학술발표논문집*, pp. 1690-1692, 2018.

[5] 이혜지, 이태형, 엄영익, "가상화 시스템에서의 하이브리드 폴링 방식 성능 분석," *한국정보과학회 학술발표논문집*, pp. 1528-1529, 2019.

[6] 송용주, 엄영익, "고성능 SSD 기반 시스템에서 I/O 간섭을 고려한 폴링 방식 분석," *한국정보과학회 학술발표논문집*, pp. 1163-1164, 2019.

[7] S. Koh, J. Jang, C. Lee, M. Kwon, J. Zhang, and M. Jung, "Faster than Flash: An In-Depth Study of System Challenges for Emerging Ultra-Low Latency SSDs," in *2019 IEEE International Symposium on Workload Characterization (IISWC)*, pp. 216-227, IEEE, 2019.

[8] J. Axboe, "FIO-Flexible IO Tester," URL <http://freecode.com/projects/fio>, 2014.