I/O 반응시간 모델을 활용한 자동 블록 I/O 트레이스 변환기의 구현 및 검정

진용석 천명준 김윤아 김지홍
서울대학교 컴퓨터공학부
(ysjin, mjchun, yoonakim, jihong)@davinci.snu.ac.kr

Design and Implementation of an Automatic Block I/O Trace Converter using an I/O Latency Model

Yongseok Jin Myoungjun Chun Yoona Kim Jihong Kim
Seoul National University

요 약

블록 I/O 트레이스는 시스템의 성능을 평가하기 위해 흔히 활용된다. 하지만 실제로 사용되고 있는 대부분의 블록 I/O 트레이스는 하드디스크 기반의 시스템에서 기록된 것으로, SSD 등의 최신 저장 장치를 활용하는 시스템에서 성능을 평가하기 위해서는 저장 장치의 특성에 맞게 적절하게 변환하여 사용해야 한다. 이 때, 저장 장치의 성능을 고려하여 각각의 I/O 요청의 발생 시간과 반응 시간을 예측할 수 있어야 실제 결과와 유사하게 변환할 수 있다. 본 논문에서는 저장 장치의 반응 시간 모델링 자동화 툴과 이를 활용한 블록 I/O 트레이스 변환기를 소개하고, 실제 저장 장치에서의 결과물과 유사한 블록 I/O 트레이스로 변환이 가능함을 실험적으로 검증한다.

1. 서 론

블록 I/O 트레이스는 실제 시스템에서 수행된 블록 I/O 요청 패턴을 수집하여 만들어진다. 일반적으로 스토리지 시스템 연구에 있어 성능을 측정하기 위해서는 평가하고자 하는 어플리케이션을 실행하기 위해 직접 환경 설정 및 입력 등을 설정해야 하는 불편함이 있다. 블록 I/O 트레이스를 활용하면 별도의 설정 없이 트레이스에 기록된 블록 I/O를 재실행하는 것만으로 실제 I/O를 수행한 것과 유사한 효과를 내어, 시스템 성능을 평가하는 데에 큰 편리함을 제공한다. 이로 인해, 스토리지 시스템의 성능 측정에는 마이크로소프트 사에서 수집하여 배포하는 MSR-Cambridge 트레이스[1]와 같은 블록 I/O 트레이스가 흔히 사용되고 있다.

하지만 성능 평가에 자주 사용되는 블록 I/O 트레이스의 대부분은 약 10년 전에 하드디스크 기반의 시스템에서 수집된 것이다. 실제로 시스템에 I/O 요청이 들어오게 되면 저장 장치의 대역폭과 처리 시간에 따라 I/O 요청들의 발생 시간과 반응 시간이 크게 달라지게 된다. 오늘날 활용되는 SSD와 같은 저장 장치는 하드디스크와 크게 달라, 기존의 트레이스를 그대로 재생해서는 유의미한 성능 평가가 불가능하다[2].

따라서, 블록 I/O 트레이스를 통해 성능을 평가하고자 한다면 분석하고자 하는 저장 장치의 특성에 알맞게 변환할 필요가 있다. 블록 I/O 트레이스를 변환하기 위해서는 각각의 I/O 요청의 발생 시간과 반응 시간을 예측해야 한다. I/O 요청의 발생 시간은 저장 장치의 성능에 따라 크게 좌우되며, 각각의 요청의 반응 시간은 I/O 요청의 크기, 읽기/쓰기 등의 패턴 등에 따라 크게 갈릴 수 있다. 따라서 이러한 저장 장치의 특성을 고려한 I/O 요청의 발생 시간과 반응 시간을 예측할 수 있어야 블록 I/O 트레이스의 변환이 가능하다.

본 논문에서는 이전에 제시한 블록 I/O 트레이스 자동 변환 기법을 확장하여[3], 저장 장치의 반응 시간 예측을 위한 모델링 자동화 툴과, 이를 활용한 블록 I/O 트레이스를 변환하는 기법을 제시한다. 저장 장치의 반응 시간은 여러 요인에 따라 크게 갈릴 수 있어, 다양한 변수를 고려한 모델링 자동화 툴을 활용하여 실제 I/O의 반응 시간과 유사한 값을 예측할 수 있도록 구현하였다. 이 기법을 활용하여 블록 I/O 트레이스를 변환했을 때, 총 수행 시간과 I/O 반응 시간의 분포가 실제 동일한 응용을 수행했을 때의 결과와 상당히 유사하게 나타났음을 확인하였다.

2. 블록 I/O 트레이스 변환 기법

블록 I/O 트레이스는 일반적으로 I/O 발생 시간, CPU 번호, 크기, 타입(읽기 또는 쓰기), 블록 번호, 반응 시간 등으로 구성된다. 이 중에서 변환하고자 하는 정보는 I/O 발생 시간과 반응 시간이다. 이 절에서는 이 두 가지를 변환하는 과정에 대해 설명한다.

2.1 I/O 요청의 반응 시간 모델링 자동화

I/O 요청의 반응 시간은 여러 요인에 따라 분포가 크게 변한다. 우선 I/O 요청의 크기에 의해서도 반응 시간이 달라진다. I/O 요청의 크기가 클수록 저장 장치가 처리하는 시간이 오래 걸리게 된다. 또한, 같은

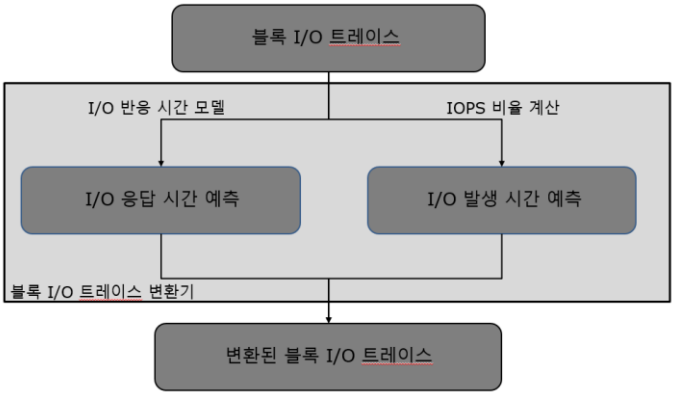

그림 1 블록 I/O 트레이스 변환기의 변환 과정

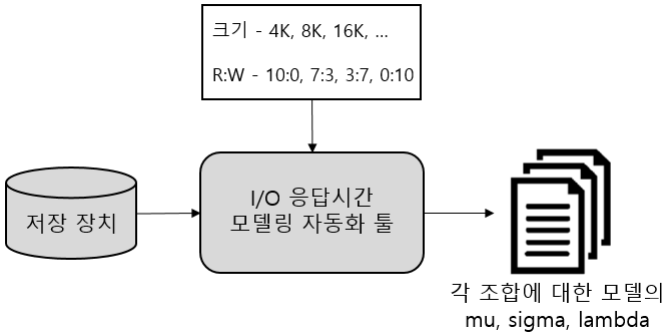

그림 2 I/O 반응 시간 모델링 자동화 툴의 동작 방식

크기인 상황에서도 I/O 패턴에 따라서 반응 시간의 분포가 크게 달라진다. 그림 3은 읽기 요청만 있을 때의 반응 시간의 분포로, 비교적 한 구간에 반응 시간이 몰려 있는 분포가 나타난다. 반면 그림 4는 읽기와 쓰기 요청의 비율이 3:7일 때의 읽기 요청의 반응 시간 분포로, 그림 3과 다르게 반응 시간이 여러 구간에 몰려 있는 분포가 나타난다.

이러한 모든 경우를 단일 모델만으로 예측할 수가 없기 때문에, 여러 경우의 수의 조합에 대해서 그림 2와 같이 I/O 반응 시간을 각각 자동으로 모델링하여 제공하는 모델링 툴을 구현하였다. 현재 적용하고 있는 변수는 I/O의 크기와 읽기와 쓰기 요청의 비율이다. 크기는 4K, 8K, 16K, 32K, 64K, 128K, 256K 총 7종류로, 읽기와 쓰기 요청의 비율은 읽기 100%, 쓰기 100%, 그리고 읽기와 쓰기의 비율이 각각 7:3, 3:7인 총 4종류로 구분한다. 각각의 조합에 대한 모델링을 자동으로 수행한 뒤, 전체 결과를 저장한다.

저장 장치의 I/O 반응 시간은 SSD의 경우 그림 2와 같이 내부 쓰레기 수집 등의 부가적인 동작으로 인해 꼬리가 길게 늘어지며, 여러 개의 피크를 나타낸다. 따라서 이를 효과적으로 모델링하기 위해 여러 개의 정규분포의 합으로 나타내어지는 가우시안 혼합 모델(Gaussian Mixture Model)을 사용한다[3].

2.2 I/O 요청의 발생 시간 및 반응 시간 예측

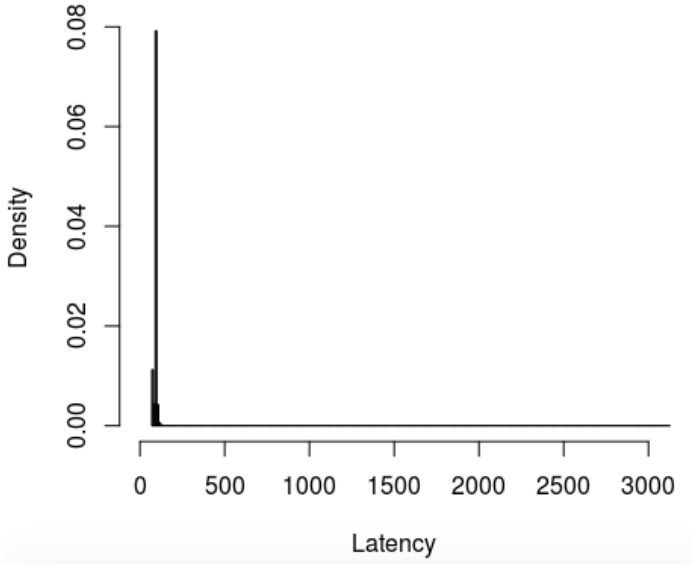

그림 3 읽기 요청만 있을 때 32K 읽기 반응 시간의 분포

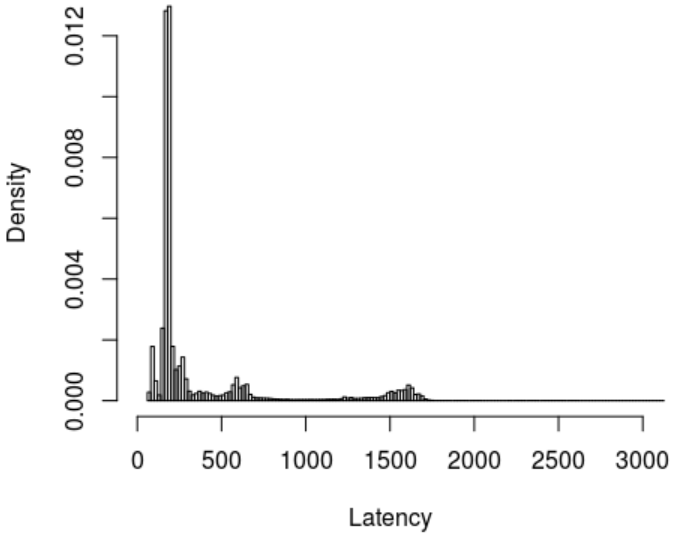

그림 4 쓰기 요청이 섞여 있을 때 32K 읽기 반응 시간의 분포

블록 I/O 트레이스 내에는 연속해서 발생되어 서로 연관이 있는 I/O 요청들의 묶음으로 나눌 수 있다. 트레이스 내에 I/O 연관성에 대한 모든 정보가 없기 때문에 정확한 분석은 사실상 불가능하지만, 저장 장치의 입장에서 서로 성능에 영향을 미치는 관계에 있으려면 비슷한 시기에 발생된 I/O여야 한다. 따라서 전체 트레이스의 I/O 발생 시간의 간격을 계산하여, 특정 임계값 이내에 있다면 인접한 I/O로 본다.

서로 연관이 있는 I/O에 대해서, I/O 발생 시간의 간격을 다시 조절하게 된다. 이 때, 앞선 I/O 요청과의 간격은 주어진 I/O 트레이스의 IOPS와 분석 대상인 저장 장치의 IOPS의 비율으로 계산한다. 기존의 I/O 요청간의 거리가 150μs, 기존 트레이스의 IOPS가 3000,

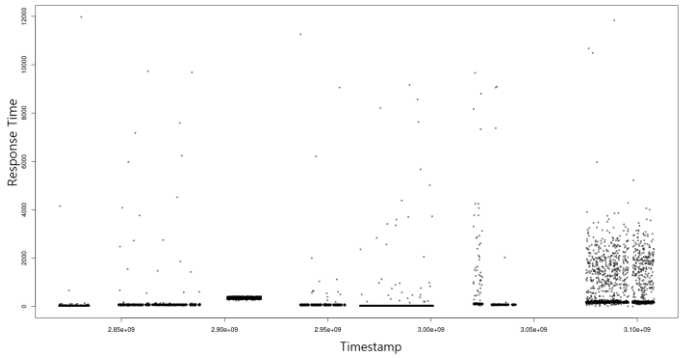

그림 5 변환기를 활용해 변환한 트레이스의 반응 시간

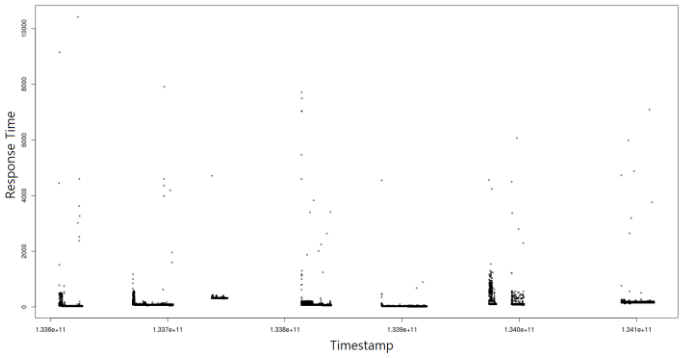

그림 6 SSD에서 추출한 트레이스의 반응 시간

분석 대상인 저장 장치의 IOPS가 9000인 경우 두 I/O의 새로운 간격은 50μs로 계산한다. 두 I/O의 발생 시간의 차이가 임계값(0.1초)을 넘은 경우 서로 연관이 없는 I/O 요청이므로, 간격을 새로 계산하지 않고 기존의 트레이스의 간격을 유지한다.

I/O의 반응 시간은 자동화 툴을 통해 계산한 모델 중 가장 적합한 모델에서 난수 발생을 통해 예측한다. 앞서 나눈 서로 연관성 있는 I/O 요청들 사이에서 읽기와 쓰기 요청의 비율을 계산한 뒤, 트레이스 내에 기재된 I/O의 크기를 참조하여 적합한 모델을 선정한다.

3. 실험 및 결과

실험은 우분투 16.04 LTS, 리눅스 커널 4.10 버전, 그리고 FIO 벤치마크를 활용하여 진행했다. I/O 패턴은 쉬는 구간과 I/O 요청 구간이 반복되며, 여러 I/O 패턴을 섞으며 수행하도록 구성하였다. 변환된 트레이스가 유의미함을 검증하기 위해 같은 패턴을 각각 SSD, HDD에서 수행한 뒤 결과를 비교했다. 실험에 사용된 SSD는 삼성 840 PRO, 하드디스크는 웨스턴디지털의 WD10EALX를 사용하였다.

하드디스크에서 추출한 트레이스를 변환한 결과물과 SSD에서의 원본 트레이스는 각각 그림 4, 그림 5에 시간의 흐름에 따른 각각의 I/O 요청의 반응 시간으로 나타내었다. 이 두 트레이스의 I/O 간격이 적절하게 변환되었는지 확인하기 위해 두 트레이스의 총 수행 시간을 비교하였다. 총 수행 시간은 SSD 원본 트레이스는 약 582.073초, 변환된 트레이스는 약 531.318초로 측정되었다. 하드디스크에서 추출한 원본 트레이스의 총 수행 시간이 SSD의 약 4배임을 감안하면 상당히 유사하게 변환했음을 확인할 수 있다.

또한, I/O 반응 시간이 유사하게 변환되었는지 확인하기 위해 변환된 트레이스와 SSD의 트레이스의 반응 시간을 두 집단의 누적분포를 비교하는 KS-Test로 검증하였다. 계산 결과 D=0.115832 정도로 약 11% 정도의 오차가 있지만 상당히 유사하게 변환된 것으로 확인되었다.

4. 결론 및 향후 연구

본 논문은 블록 I/O 트레이스를 분석하고자 하는 저장 장치에 맞추어 자동으로 변환하는 기법과, 이를 위해 자동으로 여러 가지 변수에 대응할 수 있도록 모델링을 자동화하는 툴을 소개하였다. 블록 I/O 트레이스 변환 시에 고려해야 하는 I/O 응답 시간과 I/O 발생 시간 예측에 대해 각각 모델링 자동화 툴과 IOPS의 비율을 통해 해결하였으며, 실험 결과 두 가지를 상당히 유사하게 변환하였음을 확인하였다.

일반적으로 블록 I/O는 여러 계층으로 구성된 운영체제의 버퍼 및 스케줄러를 거치는데, 이 과정에서 각종 지연이 발생해 반응 시간 예측이 어려워진다. 운영체제의 I/O 스택을 분석하여 정확한 블록 I/O의 지연 시간을 예측하는 것이 앞으로 연구할 과제이다.

5. 감사의 글

이 연구를 위해 연구장치를 지원하고 공간을 제공한 서울대학교 컴퓨터연구소에 감사드립니다. 이 논문은 2018년도 정부(과학기술정보통신부)의 재원으로 한국연구재단의 지원을 받아 수행된 연구임(NRF-2018R1A2B6006878, NRF-2015M3C4A7065645).
(교신저자: 김지홍)

참 고 문 헌

[1] Dushyanth Narayanan et al., “Write off-loading: Practical power management for enterprise storage,” ACM Transactions on Storage, vol. 4, issue 3, article 10, pp. 1-23, 2008.

[2] Michael P. Mesnier et al., “Relative fitness modeling,” Communications of the ACM, vol. 52, issue 4, pp. 91-96, 2007.

[3] Junyub On and Jihong Kim, "Design and Implementation of an SSD-Aware Automatic Block I/O Trace Converter," 한국정보과학회 학술발표논문집, pp. 1299-1301, Dec, 2016.