

시뮬레이션을 이용한 완전 비동기 디스크 어레이 시스템의 성능 평가

오유영^o, 김성수

아주대학교 정보통신대학 정보및컴퓨터공학부

A Performance Evaluation of a Fully Asynchronous Disk Array System Using Simulation

Yuyoung Oh, Sungsoo Kim

Division of Information and Computer Engineering, Ajou University

요 약

대용량 데이터의 실시간 처리를 요구하는 멀티미디어 시대에 고성능 입출력을 제공할 수 있는 저장 시스템으로서 디스크 어레이가 보편적으로 사용되고 있다. 비용 효율적인 디스크 어레이가 설계될 수 있도록 디스크 어레이의 성능을 분석할 수 있는 성능 모델의 개발은 중요하다. 큐잉 모델링을 통해서 성능 평가를 할 수 있는 방법으로는 큐잉 이론이나 시뮬레이션을 이용할 수 있다. 디스크 어레이의 병렬 및 병행 처리 특성상 큐잉 이론의 분석적인 방법의 한계성을 인식하고, 본 논문에서는 낮은 수준에서 디스크 어레이 시스템을 추상화한 시뮬레이션 기법을 이용하여 디스크 어레이의 성능 평가를 수행한다. 시뮬레이션을 통해서 산출된 디스크 어레이 요구에 대한 평균 응답 시간, 평균 큐잉 지연, 평균 서비스 시간, 평균 큐 길이 및 디스크 어레이 시스템의 이용률, 처리율 등은 최적화된 디스크 어레이 설계를 위한 시스템의 용량 산정에 활용될 수 있다.

1. 서론

초고속 정보 통신망에서 대용량 데이터의 실시간 처리를 요구하는 멀티미디어 시대의 도래와 함께 계산 처리 능력에 견줄 수 있도록 입출력 성능을 개선시키기 위한 연구가 활발히 진행중이다. 그 중에서도 두드러진 연구 성과는 고성능과 고신뢰도를 동시에 제공할 수 있는 저장 서브시스템으로 디스크 어레이(disk array) 시스템을 사용하는 것이다. 이에 디스크 어레이에 관한 연구가 많이 수행되었지만, 비용 효율적인 디스크 어레이 설계 및 용량 산정(capacity planning)을 위한 디스크 어레이 시스템에 대한 성능을 평가할 수 있는 모델링에 관한 연구는 미진한 실정이다. 디스크 어레이의 성능 평가 모델링 방법으로 큐잉 모델(queueing model)을 사용할 경우에, 큐잉 이론(queueing theory)을 적용하여 모델을 수학적으로 해석하거나 시뮬레이션 기법(simulation method)을 이용해서 평균적인 성능 평가 매트릭(metric)을 산출할 수 있다. 특히, 디스크 어레이 시스템의 디스크 어레이 요구(disk array request)에 대한 병렬 및 병행 서비스 처리의 특성상 큐잉 이론으로 디스크 어레이의 큐잉 모델에 대한 분석적인 성능 평가가 불가능하다. 큐잉 이론의 분석적인 방법의 한계성을 인식하고, 본 논문에서는 낮은 수준에서 디스크 어레이 시스템을 추상화하고 디스크 어레이 요구가 다수의 디스크 요구(disk request)로 분리(fork)되어서 해당 디스크 서버에서 병렬 및 병행 서비스되고 다시 디스크 어레이 요구로 합체(join)되는 특성을 반영할 수 있도록 개발된 시뮬레이터를 사용하여 디스크 어레이 시스템의 성능을 평가한다.

본 논문의 구성은 다음과 같다. 2장에서는 관련 연구 결과를 살펴보고, 3장에서는 시뮬레이션하고자 하는 완전 비동기 디스크 어레이 시스템의 구조와 서비스 메커니즘을 설명하고, 4장에서는 완전 비동기 디스크 어레이 시스템의 시뮬레이션 모델을 제시하고, 5장에서는 시뮬레이션을 통해서 산출된 결과를 분석하고, 마지막으로 6장에서는 본 논문에 대한 결론을 맺는다.

2. 관련 연구

디스크 어레이 시스템의 설계 및 용량 산정을 위한 성능 평가 모델은 분석적인 수학적 모델, 시뮬레이션 모델 그리고 측정 모델 등이 있다. 비용 측면에서 가장 유리한 분석적인 수학적 모델로 산출된 결과가 정확하다면 문제가 없지만, 디스크 어레이 시스템에서 디스크 어레이 요구가 다수의 디스크 요구로 분리되어 병렬 및 병행 서비스되고 다시 디스크 어레이 요구로 합체되는 메커니즘을 고려할 경우에 디스크 어레이 큐잉 모델에 대해서 큐잉 이론을 적용하여 성능 모델을 해석하는 것이 현재로서는 불가능하다. 3개 이상의 서버로 구성되는 시스템에서 일(job)이 포아송(Poisson) 분포로 도착해서 서버 수만큼의 태스크(task)로 분리되어 병렬로 지수(exponential) 분포로 서비스되고 다

시 합체되어 서비스 받는다고 단순화된 수학적 가정을 하더라도, 이 큐잉 모델에 대해서 근접된(approximate) 결과를 얻을 수밖에 없다. 그렇다고 프로토타입(prototype)을 만들어서 실제로 성능을 측정한다는 것은 소비용이 들고 유용적이지 못한 단점이 있다[1].

디스크 어레이 시스템을 일반적인 큐잉 시스템으로 간주하고 큐잉 모델링을 통해서 큐잉 이론을 적용하여 성능 평가를 수행한 연구에 대해서 살펴보면, [2]에서는 데이터 스트라이핑을 하지 않은 n 개의 독립적인 디스크로 구성된 디스크 어레이와 데이터 스트라이핑을 적용한 n 개의 디스크가 디스크 어레이 요구를 서비스 시에 개별 디스크 요구 사이에 동기화(synchronization)되어서 서비스가 이루어진다고 가정 한 후, 이 두가지 경우의 디스크 어레이에 대해서 성능 비교를 했다. 여기에서 n 개의 디스크로 구성된 디스크 어레이를 하나의 논리적인 디스크로 간주하고 $M/G/1$ 큐잉 이론을 적용해서 성능을 평가했다. 그리고 [3]에서는 비동기(asynchronous) 디스크 어레이에 대해서 디스크 어레이 요구에 대한 최소 응답 시간(minimum response time)을 산출하는 식을 유도했다. 하지만 디스크 어레이 요구가 디스크 요구로 분리되어서 병렬 및 병행 서비스 받는다는 사실을 고려하지 않았다. [1]에서는 RAID의 최대 처리율(maximum throughput)에 대한 수학적 식을 유도했다. 디스크 어레이 시스템에 대한 성능 모델이 큐잉 모델링을 통해서 큐잉 이론으로 해석된 식이 의미를 갖기 위해서는, 디스크 어레이 요구가 디스크 요구로 분리되어서 병렬 및 병행 서비스 받고 다시 디스크 어레이 요구로 합체된다는 사실이 모델링에 포함되어야만 의미가 있다. 하나의 일(job)이 다수의 태스크로 분리되어서 서비스 받는 후에 다시 하나의 일로 합체되는 서비스 메커니즘을 주로 병렬 컴퓨팅과 다중 프로그래밍 분야에서 큐잉 이론으로 해석하고자 했었다[4]. 하지만 서버의 수가 2개일 경우에만 정확한 Closed 형태의 성능 분석 식을 유도할 수 있지만 아직까지 3개 이상일 경우에는 정확한 결과를 갖는 식을 유도할 수 없다[5].

3. 완전 비동기 디스크 어레이 시스템

본 장에서는 본 논문의 대상이 되는 일반적인 디스크 어레이 시스템의 구조, 완전 비동기 디스크 어레이 시스템의 특징을 설명한다. 그리고 완전 비동기 디스크 어레이 시스템의 서비스 메커니즘, 디스크 어레이 요구의 병렬 및 병행 서비스의 특성을 반영한 큐잉 모델을 살펴본다.

3.1 디스크 어레이 시스템 구조

본 논문에서 고려하고 있는 디스크 어레이 시스템은 그림 1과 같이 N 개의 동일한 디스크, 디스크 어레이 제어기, 디스크 어레이 로직, SCSI 버스, 그리고 인터페이스로 구성된다. 디스크는 고정된 대역폭을 갖는 SCSI 버스와 인터페이스에 위해서 디스크 어레이 제어기에 연결된다. 디스크 어레이 제어기는 선인출한 데이터를 저장하거나 나중에

본 연구는 1999년도 정보통신부 정보통신 우수시범학교 지원사업에 의한 결과임.

요청된 디스크 어레이 요구에 대한 응답 시간을 줄임으로써 성능을 향상시키기 위해서 캐쉬로 사용되는 지역 메모리를 가지고, 고신뢰도를 보장하기 위해서 결함허용을 위한 패리티 정보를 사용할 경우 XOR 결합으로 인해서 액세스가 불가능한 데이터를 생성하는 XOR Engine 을 갖는다. 디스크 어레이 로직은 개별적인 디스크들이 논리적으로 디스크 어레이가 될 수 있도록 하는 소프트웨어 모듈로서 각 디스크와 디스크 제어기에 존재한다. 디스크 어레이 버퍼는 각 디스크에서 디스크 요구에 대한 데이터를 인터페이스를 통해서 디스크 제어기로 보내기 위한 완충 역할을 수행한다.

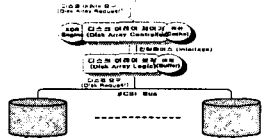


그림 1 디스크 어레이 시스템의 구조

3.2 완전 비동기 디스크 어레이 시스템

다수의 디스크를 하드웨어 혹은 소프트웨어에 의해서 어떻게 하나의 디스크 어레이로 만들 것인가에 의해서 디스크 어레이 시스템을 다음과 같이 4가지로 분류할 수 있다[6].

- 완전 동기 디스크 어레이(Fully Synchronous Disk Array): 모든 디스크의 헤드 탐색과 회전이 하드웨어에 의해서 동기화가 맞춰진다. 이 때에 인터페이스 대역폭이 충분히 크다면 단일 디스크의 전송 대역폭보다 디스크 어레이를 구성하는 디스크 수만큼 확장된 대역폭을 갖지만, 동일한 탐색 시간(seek time) 및 회전 지연(rotational latency)을 초래한다.
- 부분적 동기 디스크 어레이(Partially Synchronous Disk Array): 모든 디스크의 헤드 탐색은 하드웨어에 의해서 동기화가 맞춰지지만, 회전은 각 디스크에서 독립적으로 수행된다. 데이터는 모든 디스크의 동일한 물리적 위치에 스트라이핑된다.
- 완전 비동기 디스크 어레이(Fully Asynchronous Disk Array): 각 디스크의 헤드 탐색 및 회전이 독립적으로 수행된다. 모든 디스크가 한 디스크 어레이 요구에 대해서 병렬적으로 서비스에 참여하는 동기 디스크 어레이와 다르게 디스크 어레이 요구들에 대해서 데이터를 갖는 디스크들이 병렬 및 병행적으로 서비스를 한다.
- 완전 비결집 디스크 어레이(Fully Decoupled Disk Array): 모든 디스크가 완전히 독립적으로 동작하지만 데이터가 스트라이핑되어 저장되지 않는다. 일반적으로 데이터를 스트라이핑하는 디스크 어레이와 비교를 목적으로 모델링되는 디스크 어레이이다. 특히, 디스크 어레이 요구가 동일한 비율로 각 디스크에 의해서 서비스 받는다는 비현실적인 가정을 한다.

동기 디스크 어레이의 성능 평가 모델은 M/G/1 큐잉 모델링을 통해서 큐잉 이론을 적용하여 수학적으로 정확한 성능 평가 결과를 얻을 수 있지만, 완전 비동기 디스크 어레이는 디스크 어레이 요구의 병렬 및 병행 서비스 특성으로 인해서 분석적인 방법으로는 성능을 평가하기가 어렵다.

4. 시뮬레이션 모델

본 장에서는 단순화된 수학적 가정을 배제한 완전 비동기 디스크 어레이 큐잉 시스템에 대한 시뮬레이션 모델을 제시하고 완전 비동기 디스크 어레이 시뮬레이터를 구성하는 모듈들을 살펴본다. 그리고 디스크 서버에서 디스크 요구의 서비스 시간 계산 시에 어떻게 디스크의 물리적/기계적 특성이 반영되는 지를 설명한다.

4.1 시뮬레이션 모델

실제적인 완전 비동기 디스크 어레이 시스템에 대한 시뮬레이션은 위해서 그림 2와 같이, 디스크 어레이 요구가 Fork Node에서 몇 개의 디스크 요구로 분리될 것인가는 디스크 수, 디스크의 특성 파라미터, 디스크 어레이 요구 크기 등에 의해서 서비스에 참여하는 디스크 요구 수가 결정될 것이고, 또한 서비스가 시작되는 디스크도 이산 행동(discrete uniform) 분포에 의해서 선택되도록 해야 한다. 그리고 각 디스크 요구가 디스크 서버에서 서비스 받는 시간은 디스크 헤드에 원하는 데이터가 저장된 실린더의 트랙까지 이동하는 탐색 시간, RPM 속도에 따른 디스크 회전에 의해서 데이터 블록이 저장된 섹터를 찾는 데 걸리는 지연 시간, 데이터를 디스크 전송 속도로 디스크 어레이 로직의 일시적인 캐쉬로 사용되는 트랙 버퍼로 전송하는 시간, 그리고 인터페이스를 통해서 버퍼의 데이터를 디스크 어레이 제어기로 전송하는 시간을 합산한 것이다. 디스크 어레이 요구의 도착률, 각 디스크 서버의 큐 길이, 그리고 서비스 정책 등은 큐잉 모델과 같이 각각 포아송 분포에 의한 디스크 어레이 요구의 도착, 무한대 큐 길이, FCFS 정책에 의한 디스크 요구의 서비스로 시뮬레이션된다.

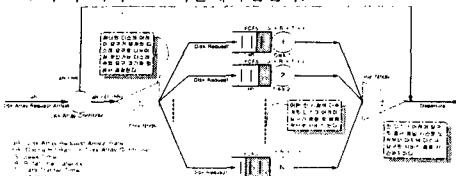


그림 2 완전 비동기 디스크 어레이 시스템의 시뮬레이션 모델

4.2 완전 비동기 디스크 어레이 시뮬레이터

완전 비동기 디스크 어레이에 대한 시뮬레이터는 C 프로그래밍 언어에 의해서 구현되었고, Event-driven 방식으로 시뮬레이션된다. 그림 3은 시뮬레이터를 구성하는 모듈 및 주요 관련 파라미터들을 보여준다. 완전 비동기 디스크 어레이 시스템은 Input Parameter Determination Module에 의해 입력되는 디스크 수, 디스크 어레이 요구 도착률, 디스크 어레이 요구 크기, 디스크 특성 파라미터, 인터페이스 전송 속도, 그리고 디스크 어레이 제어기의 캐쉬에 대한 적중률(hit ratio) 등에 의해서 시스템이 결정된다. Simulation Engine Module에 의해서 디스크 어레이 요구가 Fork Node에서 분리된 디스크 요구를 디스크 서버에서 병렬 및 병행적으로 서비스되는 것을 시뮬레이션하게 된다. 각 디스크 요구가 끝나는 시점에 Performance Metric Collection Module에 의해서 디스크 요구에 대한 응답 시간, 서비스 시간(service time), 큐잉 지연(queueing delay), 큐 길이(queue length)에 대한 정보가 수집된다.

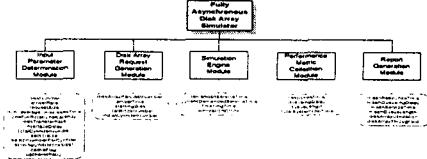


그림 3 완전 비동기 디스크 어레이의 시뮬레이터 모듈

디스크 서버에서 디스크 요구에 대한 서비스 시간을 계산할 때에, 그림 4와 같이 디스크 헤드의 스케줄링은 이전 디스크 요구에 대한 서비스가 종료될 때 디스크 헤드는 데이터 서비스가 끝난 실린더에 위치하고 현재 디스크 요구를 서비스하기 위해서 이산 행동 분포에 의해서 선택된 새로운 실린더로 이동하게 된다. 이 디스크 스케줄링에 의해서 실린더 거리가 계산됨으로써 탐색 시간이 계산된다. 이 때에 디스크 요구 크기에 따라 두 가지 서비스 시나리오를 고려해야 한다. 디스크 서버에 배정된 디스크 요구 크기가 한 실린더 범위 안에서 서비스 가능한 경우(Type 1)와 디스크 요구가 한 개 이상의 실린더에 걸쳐서 서비스되어야 하는 경우(Type 2)로 나누어진다.

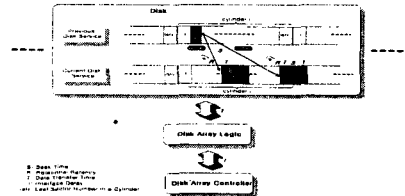


그림 4 디스크 요구에 대한 서비스 시나리오

5. 시뮬레이션을 통한 성능 평가

완전 비동기 디스크 어레이 시스템의 성능을 평가하기 위해서, 디스크 어레이는 동일한 디스크(IBM 0661 SCSI 디스크)로 구성되고 모든 디스크는 [1]과 같은 디스크 특성 파라미터를 갖는다고 가정한다. 그리고 시뮬레이터에 입력되는 시스템 결정 파라미터는 표 1, 출력되는 성능 파라미터는 표 2와 같다. 특히, IBM 0661 디스크에 대한 탐색 시간 함수는 식 (1)과 같다[1]. 식(1)에서 x: 탐색 거리(실린더 단위), a, b, c: 한 실린더에 적용되는 상수 값(a = 0.4623, b = 0.0092, c = 2)이다. 이 식에 의해서 최초 탐색 시간 및 인접 실린더간의 탐색 시간(최소 탐색 시간)이 포함되어 디스크 요구에 대한 탐색 시간이 계산된다.

$$\text{탐색시간}(seekTime) = \begin{cases} 0, & \text{if } x=0 \\ a(x-1)^{0.5} + b(x-1) + c, & \text{if } x>0 \end{cases} \quad (1)$$

표 1 성능 평가에 사용된 입력 파라미터

항목	값	설명
arrivalRate(aR)	5, 10, 15, 20, 25/sec	초당 도착하는 디스크 어레이 요구 수
diskArrayRequestSize(darS)	4, 64, 128, 256 KB	디스크 어레이 서비스 크기
diskNumber(dN)	1 ~ 15	디스크 어레이를 구성하는 디스크 수
cylinderNumber(cN)	949	디스크의 실린더 총 수
sectorNumberPerCylinder(sN)	686	실린더의 섹터 총 수
sectorSize(sS)	512 B	한 섹터의 크기
strippingUnitSize(sU)	512, 1024, 2048 B	스트라이핑 크기(한 블록)
min. average, maxSeekTime (msT, asT, xST)	2.0, 12.5, 25.0 msec	최소, 평균, 최대 탐색 시간
oneFullRotationalLatency(rL)	13.9 msec	회전 지연 시간
diskTransferRate(dtR)	1.7 MB/sec	디스크 전송 속도
interfaceSpeed(iS)	10.0 MB/sec	인터페이스 전송 속도
cacheHitRatio(hR)	0.2, 0.4	캐쉬 적중률

표 2 성능 평가에 사용된 출력 파라미터

항목	설명
meanResponseTime(mRT)	평균 응답 시간(msec)
meanQueueingDelay(mQD)	평균 큐잉 지연(msec)
meanServiceTime(mST)	평균 서비스 시간(msec)
meanQueueLength(mQL)	평균 큐 길이(디스크 어레이 요구 수)
diskArrayUtilization(duU)	디스크 어레이 시스템의 이용률(utilization)
diskArrayThroughput(dtT)	디스크 어레이 시스템의 처리율(KB/sec)

5.1 평균 응답 시간

디스크 어레이 제어기에서 캐쉬를 사용할 경우와 캐쉬를 사용하지 않을 경우로 나누어 시뮬레이션 실험을 수행함으로써, 완전 비동기 디

스크 어레이 시스템의 디스크 어레이 요구에 대한 평균 응답 시간을 획득했다. 캐시를 사용할 경우에 [7]에 의해서 LRU(Least Recently Used), RR(Random Replacement) 등의 캐쉬 제어 정책을 사용한다고 가정할 때 캐쉬 적중률(hR)에 대해서 캐쉬로 사용되는 지역 메모리의 크기는 결정된다. 그림 5는 디스크 수(dN)의 변화에 따른 디스크 어레이 요구 도착률(aR), 디스크 어레이 크기(darS)의 조합에 대해서 산출된 디스크 어레이 요구에 대한 평균 응답 시간(mRT)을 보여준다. aR이 크면 큐잉 지연이 발생할 확률이 커지므로 mRT가 증가하지만 변화의 폭은 그렇게 크지 않다. 반면에 darS가 4KB보다 128KB일 때, dN이 10 이하일 경우에 변화의 폭이 크다. 그림 6은 aR이 10/sec, darS가 64KB일 때 캐시를 사용할 경우와 사용하지 않을 경우에 mRT를 보여준다. 캐시를 사용하지 않을 경우보다 캐쉬 적중률(hR)이 0.2, 0.2보다 0.4일 경우에 mRT는 감소한다.

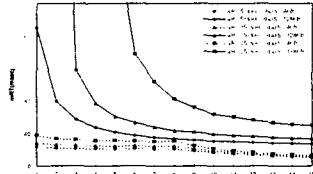


그림 5 디스크 어레이 요구에 대한 평균 응답 시간
(aR: 5, 15, 25/sec, darS: 4, 128KB, N: 변화, sU: 512B, No Caching)

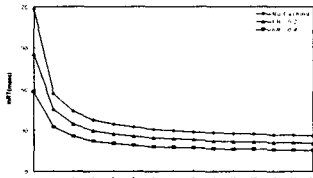


그림 6 캐쉬 적중률 변화에 따른 디스크 어레이의 평균 응답 시간
(aR: 10/sec, darS: 64KB, sU: 512B, Caching)

5.2 평균 큐잉 지연 및 평균 서비스 시간

완전 비동기 디스크 어레이 시스템에서 디스크 어레이 요구에 대한 평균 큐잉 시간(mQD)은 분리된 디스크 요구 중에서 가장 긴 큐잉 지연을 디스크 어레이 요구에 대한 큐잉 지연으로 간주하고 모든 디스크 어레이 요구에 대해서 평균값을 취한 것이다. 디스크 어레이 요구에 대한 평균 서비스 시간(mST)은 앞서 산출된 mRT에서 mQD를 뺀 값이다. 그림 7과 그림 8은 각각 mQD와 mST에 대한 결과를 보여준다. mQD는 dN이 증가할수록 감소하고, aR의 변화에 대해서는 변화의 폭이 크지 않지만 darS의 변화에 대해서는 상당히 크게 변화하는 양상을 보인다. mST는 darS가 작을 경우에는 dN에 무관하게 비슷한 값을 보이지만, darS가 크고 dN이 작은 경우에는 증가한다.

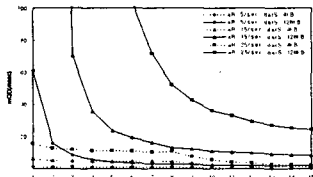


그림 7 디스크 어레이 요구에 대한 평균 큐잉 지연
(aR: 5, 15, 25/sec, darS: 4, 128KB, N: 변화, sU: 512B, No Caching)

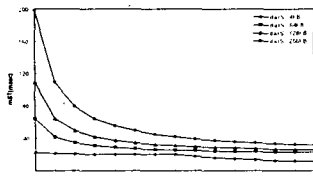


그림 8 디스크 어레이 요구 크기의 변화에 따른 평균 서비스 시간
(aR: 5/msec, darS: 4, 64, 128, 256KB, N: 변화, sU: 512B, No Caching)

5.3 평균 큐 길이

완전 비동기 디스크 어레이 시스템에 대한 시뮬레이션 모델에서 각 디스크 서버의 큐 길이는 무한대로 가정했다. 여기에서 큐 길이의 기본 단위는 디스크 요구이다. 한 디스크 어레이 요구에 대한 큐 길이는 분리된 디스크 요구 중에서 가장 긴 큐 길이가 디스크 어레이 요구에 대한 큐 길이가 되기 때문에, 디스크 어레이 요구에 대한 평균 큐 길이(mQL)는 모든 디스크 어레이 요구에 대해서 평균값을 취한 것이다. 시뮬레이터에서 한 디스크 요구에 대한 큐 길이는 디스크 요구가 디스크 서버에서 서비스가 종료되는 시점에 서버의 큐에 대기 중인 디스크 요구의 수를 확인함으로써 산출된다. 그림 9는 dN의 변화에 따른 aR, darS의 조합에 대해서 얻어진 디스크 어레이 요구에 대한 mQL를 보여준다. darS가 작을 경우에는 aR의 변화에 민감하지 않지만, darS

가 크고 aR이 15, 25/sec일 경우에 N이 10이하에 대해서는 mQL이 길고 dN의 변화에 민감하게 변한다.

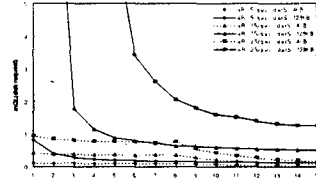


그림 9 디스크 어레이 요구에 대한 평균 큐 길이
(aR: 5, 15, 25/sec, darS: 4, 128KB, N: 변화, sU: 512B, No Caching)

5.4 완전 비동기 디스크 어레이 시스템의 이용률 및 처리율

완전 비동기 디스크 어레이 시스템의 이용률(daU)은 어떤 디스크 서버도 서비스하지 않은(idle) 시간을 Performance Metric Collection Module에서 시뮬레이션이 종료될 때까지 totalSystemIdleTime이라는 파라미터에 유지함으로써 시스템의 비이용률이 계산됨으로써 구해진다. 그림 10은 dN의 변화에 따른 aR, darS의 조합에 대해서 산출된 디스크 어레이 시스템의 daU를 보여준다. dN이 작을수록, aR과 darS가 클수록 daU는 높다. 또한 darS가 클수록 dN의 변화에 따른 daU의 변화의 폭도 크다. 완전 비동기 디스크 어레이 시스템의 처리율(daT)은 조당 얼마만큼의 데이터를 처리할 수 있는가를 의미한다. 그림 11은 dN의 변화에 따른 aR, darS의 조합에 대해서 산출된 완전 비동기 디스크 어레이 시스템의 daT를 보여준다. darS가 작으면 aR의 변화에 민감하지 않지만, darS가 크면 aR이 5, 15, 25/sec일 경우에 각각 N이 1, 2, 4 부터 급격히 높아진다.

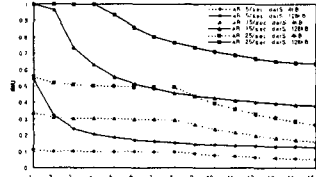


그림 10 완전 비동기 디스크 어레이 시스템의 이용률
(aR: 5, 15, 25/sec, darS: 4, 128KB, N: 변화, sU: 512B, No Caching)

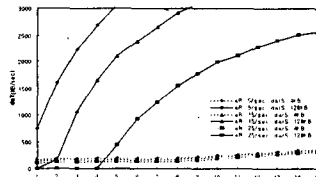


그림 11 완전 비동기 디스크 어레이 시스템의 처리율
(aR: 5, 15, 25/sec, darS: 4, 128KB, N: 변화, sU: 512B, No Caching)

6. 결론 및 활용 방안

완전 비동기 디스크 어레이 시스템의 경우에 디스크 어레이 요구가 디스크 요구로 분리되어 디스크 서버에서 병렬 및 병행적으로 서비스 받은 후에 다시 합체되는 특성을 갖기 때문에 큐잉 이론으로 해석하는 것이 불가능하기 때문에, 분리된 디스크 요구가 병렬 및 병행적으로 서비스되는 것을 반영할 수 있는 시뮬레이션 기법을 이용해서 완전 비동기 디스크 어레이 시스템에 대한 성능을 평가했다. 본 논문에서 시뮬레이션을 통해서 산출된 디스크 어레이 요구 및 디스크 어레이 시스템에 대한 성능 매트릭들은 디스크 어레이 설계를 위한 시스템의 용량 산정에 활용될 수 있다.

참고문헌

- [1] G. Gibson, *Redundant Disk Arrays: Reliable, Parallel Secondary Storage*. PhD thesis, Univ. of California at Berkeley, December 1991.
- [2] M. Y. Kim, "Synchronized Disk Interleaving," *IEEE Transactions on Computers*, pp. 978-988, November 1986.
- [3] M. Y. Kim and A. N. Tantawi, "Asynchronous Disk Interleaving: Approximating Access Delays," *IEEE Transactions on Computers*, pp. 801-810, July 1991.
- [4] R. Nelson and A. N. Tantawi, "Approximate Analysis of Fork/join Synchronization in Parallel Queues," *IEEE Transactions on Computers*, pp. 739-743, June 1988.
- [5] C. Kim and A. K. Agrawala, "Analysis of the Fork-join Queue," *IEEE Transactions on Computers*, pp. 250-255, February 1989.
- [6] Christopher L. Elford and Daniel A. Reed, "Technology Trends and Disk Array Performance," *Journal of Parallel and Distributed Computing* 46, pp. 136 - 147, November 1997.
- [7] R. Karedia, J.S. Love and B.G. Wherry, "Caching Strategies to Improve Disk System Performance," *IEEE Computer*, pp. 38-46, March 1994.