대규모 정보처리를 위한 병렬 화일시스템 설계에 관한 연구

장 시 웅[†] · 정 기 동[††]

요 약

본 연구에서는 Workstation Cluster 환경에서 전통적인 디스크들을 디스크 배열처럼 사용할 수 있는 병렬 화일시스템(N-PFS)의 성능을 해석적 방법과 실측 결과를 사용하여 분석하였다. N-PFS는 소규모 서버 시스템에서 고성능 화일 서버로 사용될 수 있으며, 멀티미디어 데이타나 과학 계산용 데이타와 같은 대용량 데이타를 효율적으로 처리할 수 있다. 본 논문에서는 N-PFS의 성능 분석을 위한 해석적 모델을 제시하였으며, 제시된 해석적 모델의 정확성을 시스템에서의 실측값과 비교함으로써 검증하였다. 해석적 방법과 실측을 통하여 성능을 분석한 결과, 워크스테이션 클러스터 환경에서 대용량 데이타 처리에 적합한 스트라이핑 단위는 64~128Kbytes이며, 8개의 디스크에서 최대 대역폭은 15.8Mbytes/sec로 나타났다. 그리고 대용량 데이타 처리시의 병목 현상은 버퍼 간의 데이타 복사시간으로 나타났다.

A Study of Designing Parallel File System for Massive Information Processing

Si Woong Jang[†] · Ki Dong Chung[††]

ABSTRACT

In this study, the performance of a parallel file system (N-PFS), which is implemented using conventional disks as disk arrays on a Workstation Cluster, is analyzed by using analytical method and actual values in experiments. N-PFS can be used as high-performance file server in small-scale server systems and efficiently process massive data I/Os such as multimedia and scientific data. In this paper, an analytical model was suggested and the correctness of the suggested was verified by analyzing the experimental values on a system. The result of the performance analysis, which is achieved by joining the analytical model and the experimental values, shows that the appropriate striping unit for processing massive data on the Workstation Cluster with 8 disks is 64~128Kbytes and the maximum throughput on it is 15.8Mbytes/sec. In addition, the performance of parallel file system on massive data is bounded by the time required to copy data between buffers.

1. 서 론

최근 통신망의 급속한 발전과 컴퓨터 처리 능력의 향상으로 컴퓨터는 단순 문자나 숫자 처리의 응용에 국한되지 않고 음성, 화상, 비디오, 그래픽스와 같은 멀티미디어 정보의 처리로 응용이 급속히 확대되고 있다. 반도체의 발달로 처리기는 멀티미디어나 과학 계산용 정보를 처리하기에 충분한 능력을 가지고 있으나 디스크와 같은 저장장치의 경우 기계적 특성에

※이 논문은 1995년도 한국학술진흥재단의 공모과제 연구비에 의하여 연구되었음.

† 종신회원:동의대학교 전산통계학과 전임강사
†† 종신회원:부산대학교 전자계산학과 교수

논문접수:1997년 3월 10일, 심사완료:1997년 4월 29일

의해 성능 개선이 효율적으로 이루어 지지 않아 전체 컴퓨터 시스템의 병목현상을 일으키는 요소가 되어 왔다.[1]

멀티미디어 정보를 서비스하기 위하여는 고성능 화일서버가 필요하므로 이에 대한 연구는 여러 곳에서 진행되어 왔으나 대부분의 연구가 CPU 성능, 디스크 개수, 스트라이핑 단위에 따른 성능 분석에 관한 체계적인 연구를 수행하지 않아서 병렬 화일시스템의 성능 예측이 어려운 실정이다[2, 3, 4]. 또한 병렬 화일 시스템의 성능을 분석할 해석적 모델이 제시되지 않아서 이에 대한 체계적 연구가 필요한 실정이다.

따라서 본 논문의 연구 목적은 다음과 같다.

첫째, 병렬 화일시스템의 성능을 해석적 방법과 실측을 통해 체계적으로 분석한다. 실측을 통한 성능 분석을 통해 CPU의 계산 능력과 네트워크 대역폭 등을 고려한 종합적인 화일 입출력의 성능을 분석하고 디스크 수의 변화에 따른 화일 입출력의 병목현상을 분석한다. 그리고 다양한 워크로드에 대한 성능 분석을 위해 해석적 모델을 제시하고 해석과 실측의 결과를 비교함으로써 해석적 모델의 타당성을 검증한다.

둘째, 현재까지 구축된 병렬 화일시스템은 고성능, 대규모 시스템을 중심으로 설계되고 구현되었기 때문에 CPU 자원의 병목현상을 고려하지 않았다. 그러나 소규모 서버의 경우에는 CPU 자원의 병목현상이 중요한 요소를 차지하기 때문에 본 논문에서는 화일 입출력시 CPU 자원의 병목 현상을 분석하고 이에 대한 해결 방안을 제시한다.

이상과 같은 목적을 위해 본 논문에서는 로컬 네트워크를 통하여 다수의 Workstation들이 연결되어 있는 분산 UNIX 환경 하에서 로컬 네트워크 내에 있는 다수의 디스크 자원들을 디스크 배열처럼 사용할 수 있는 네트워크 단위 병렬 화일시스템(Network-wide Parallel File System : N-PFS)을 구성하여 성능을 평가하고 분석한다.

2. 병렬 화일시스템의 성능

병렬 화일시스템의 성능 연구는 연구의 초창기에는 다중 디스크 중심의 시뮬레이션으로 진행되었고 최근의 많은 연구에서는 자신의 연구를 통하여 구축한 병렬 화일시스템의 최대 성능을 제시하였다.

그러나, 대부분의 연구에서는 시스템의 성능을 실측을 통하여만 평가하고 해석적 모델을 제시하지 않았으며, 구체적인 병목 현상 분석과 이의 해결 방안에 대한 연구가 부족하다. N-PFS와 유사한 병렬 화일 시스템의 성능 분석 사례를 살펴보면 다음과 같다.

2.1 iPSC/2 CFS

Intel의 iPSC/2 상에서 구현된 CFS[4]는 큰 데이타 화일 접근시 병렬 입출력 기능을 제공함으로써 입출력 성능 향상을 지원하기 위해 구성된 패키지 형태의 S/W로서 부가적인 H/W 장치를 요구하지 않는다.

Intel의 CFS는 I/O의 노드 수가 8이고 I/O 노드당 1대의 디스크를 장착하였을 때 최대 대역폭이 6~7 MBytes/sec로 나타나서 다른 화일시스템에 비해 입출력 대역폭이 크게 낮다. 그 이유는 Intel CFS에서는 블럭의 크기를 16 KBytes 이하로 작게하여서 디스크의 탐색시간과 회전지연 시간이 차지하는 비율이 크기 때문이다. 따라서 Intel CFS는 최초의 병행 화일시스템이라는 큰 의의를 가지지만 화일시스템의 성능 측면에서는 매우 빈약한 시스템이다.

2.2 sfs 화일시스템

sfs(scalable file system)[10]는 확장 가능한 디스크 배열을 갖는 Thinking Machine사의 CM-5에서 UNIX와 호환을 유지하면서 고도의 확장성을 갖도록 설계된 병렬 화일시스템이다. CM-5는 수십 개에서 수천 개의 처리기 노드를 가질 수 있도록 설계되어 있으며 각 노드는 SIMD 또는 MIMD 형으로 작성된 프로그램을 수행할 수 있다.

sfs에서는 디스크 1개당 성능, 디스크 개수에 따른 누적 대역폭과 시스템의 최대 대역폭은 제시하였으나 시스템의 병목 현상을 분석하지 않았고, 시스템의 성능 평가를 위한 해석적 모델을 제시하지 않았다. sfs의 디스크당 읽기 대역폭은 1.5MBytes이며 시스템의 최대 성능은 디스크의 개수가 118개일 때 185Mbytes/sec이다.

2.3 RAID-II

RAID-II[2]는 UCB에서 RAID-I의 경험을 바탕으로 멀티미디어, CAD, 객체지향 데이타 베이스와 같이 높은 대역폭을 요구하는 응용을 지원하기 위해 개

발된 시스템이다. RAID-II는 화일서버와 클라이언트에게 디스크 배열의 높은 대역폭을 제공하기 위해 H/W와 S/W를 설계하였다.

RAID-II의 S/W는 디스크 배열의 대역폭을 최대로 이용하기 위하여 설계되었으며 LFS(Log-Structured File System) 구조를 수용한다. LFS 로그는 디스크에 적당한 단위(예, 64 KBytes)로 striping된다. RAID-II에서 수행되는 LFS의 성능은 큰 읽기와 쓰기 연산에 대해서는 디스크 H/W 성능의 65% 정도 성능을 나타낸다. 24개의 디스크를 갖는 하나의 XBUS 보드에서 화일시스템이 수행되지 않는 H/W의 성능은 임의(random) 읽기 연산에 대해서는 20 MBytes/sec의 성능을 나타내고 순차적인 읽기 연산에 대해서는 31 MBytes/sec 성능을 나타낸다.

RAID-II는 XBUS 보드와 같은 별도의 H/W에 기반을 두었으므로 융통성과 이식성이 결여되어 있고 각 보드에 대한 최대 성능이 기술되어 있으나 각종 성능에 대한 체계적인 분석이 이루어지지 않았다.

2.4 Vesta 병렬 화일시스템

Vesta 병렬 화일시스템[1, 5]은 IBM T.J. Watson 연구소에서 개발 중인 초대규모 병렬 처리 시스템인 Vulcan에서 사용하기 위하여 개발된 병렬 화일시스템으로 수치연산이 많은 과학계산 응용에서 MPP의 I/O 문제를 해결하는 것을 목표로 삼고 있다.

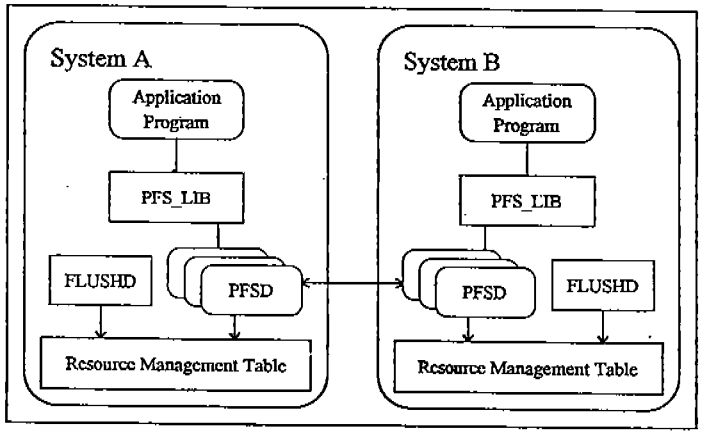
화일은 사용자의 지정에 따라 여러 개의 I/O 노드로 물리적으로 분할되어 배치된다. 화일이 사용자에 의해 기술된 I/O 노드로 분산할 수 있다는 점이 화일시스템이 암시적으로 데이타를 여러 노드로 분산하는 병행 화일시스템과 다른 점이다.

Vesta의 성능 분석에서는 데이타의 접근 크기와 I/O 노드의 개수에 따른 화일시스템의 대역폭, 계산 노드와 I/O 노드의 개수에 따른 화일시스템의 대역폭 등을 분석하였다. 계산노드가 4개이고 I/O 노드의 개수가 12개일때 최대 대역폭은 약 27Mbytes/sec의 대역폭을 보였다.

Vesta에서는 다른 연구보다 다양한 성능분석 결과를 제시하였지만 CPU의 성능에 따른 화일시스템 전체의 성능을 분석하지 않았으며 화일시스템의 성능을 일반화할 수 있는 해석적 모델의 제시와 해석적 성능 분석을 제시하지 않았다.

3. N-PFS의 구조

본 연구에서 성능분석 모델로 구성한 N-PFS는 트랜잭션과 같은 작은 데이타를 병행 처리하기 위하여 설계되었던 S-CFS(Shared-Concurrent File System)를 대용량 데이타 처리를 위한 화일시스템으로 발전시키면서 이름을 개칭한 것이다. S-CFS 구현시에는 시스템의 성능을 실측을 이용하여 부분적으로 평가하였으나 N-PFS에서는 실측과 해석적 모델을 결합하여 다양한 성능을 분석하였다. N-PFS의 설계와 구현



(그림 1) N-PFS의 소프트웨어 블럭 구성도

에 관한 내용은 S-CFS[11]를 참조하면 되므로 본 논문에서는 N-PFS의 구조와 설계에 대한 내용을 간략히 언급한다.

본 연구에서는 시스템 구성(System Configuration) 정보에 따라 로컬 시스템에 있는 전통적인 디스크들을 디스크 배열 형태로 사용할 수도 있고, 네트워크 상에 분산되어 있는 디스크들을 디스크 배열 형태로 사용할 수 있도록 융통성 있게 설계하였다.

N-PFS는 (그림 1)에 보여진 것처럼, 사용자 모델에 함께 링크되는 라이브러리인 PFS_LIB(Parallel File System Library), 라이브러리로부터의 요구를 받아 실제 입출력을 수행하고 자원을 관리하는 서버 데몬인 PFSD(Parallel File System Daemon), inode 등과 같은 메타 데이타들을 주기억장치에서 디스크로 주기적으로 flush하는 FLUSHD(Flush Daemon)의 3개 소프트웨어 블럭으로 구성되며 각 블럭의 기능은 다음과 같다.

3.1 PFS_LIB

PFS_LIB는 사용자 프로그램과 서버 데몬 사이의 인터페이스 역할을 담당한다. N-PFS에서 서버 데몬은 한 번에 하나의 블럭에 대한 입출력만을 처리하고 라이브러리에서 사용자의 데이타를 여러 개의 단위 블럭으로 분할하여 각각의 블럭을 서로 다른 서버 데몬에게 입출력을 수행하도록 함으로써 화일 입출력의 병렬성을 제어한다. 이 때, 라이브러리는 사용자가 명시한 요구 병렬성에 따라 해당 입출력을 서비스할 데몬의 수를 결정한다.

3.2 PFSD

PFSD는 클라이언트(라이브러리) 및 다른 시스템의 서버로부터 오는 요청을 처리하기 위해 동작하는 서버 데몬으로서 시스템에 하나 이상의 PFSD가 대기 상태로 존재한다. PFSD는 처음 구동될 때 클라이언트로부터의 요구를 서비스하기 위해 소켓을 초기화하고 클라이언트로부터의 병렬 요구를 동시에 처리하기 위해 다수의 데몬 프로세스를 생성한다. 이 때 생성할 프로세스의 수는 운용자가 지정한 수만큼 생성한다. 각 PFSD는 특정 포트로 바인드되며 N-PFS를 접근하는 응용 프로그램은 바인드된 포트를 통해 로컬 시스템에서 동작하는 PFSD에게 원하는 기능을 요청한다. PFSD는 각 요청을 분석하여 로컬 시스템에 있는 서브디스크에 대한 요청이면 스스로 처리하고 그 외의 서브디스크에 대한 요청은 해당 서브디스크를 관리하는 리모트 시스템의 PFSD에게 요청하여 처리하도록 한다.

3.3 FLUSHD

FLUSHD는 N-PFS에서 데몬들 사이에 공유하는 자원들을 관리하는 역할을 수행하는 데몬으로서 각 시스템에 하나씩 동작하며 다음과 같은 두 가지 기능을 가지도록 설계되었다.

첫째, 버퍼나 inode 리스트 및 화일 테이블과 화일 디스크립터 테이블 등 FLUSHD와 PFSD에 의해 공유되는 자원들을 공유 메모리 상에서 초기화시킨다.

둘째, 버퍼나 inode와 같이 메모리와 디스크 간에 동기화시켜야 하는 자원들을 주기적으로 디스크에 플러쉬시킨다.

4. 해석적 방법에 의한 성능분석

해석적 방법에 의한 성능 분석을 효율적으로 수행하기 위하여 화일 입출력에 영향을 미치는 주요 요소인 로컬 CPU 수행시간, 네트워크 전송시간, 리모트 CPU 수행시간, 입출력 버스(SCSI bus) 전송시간, 디스크 접근시간 등의 5가지 요소를 사용하여 해석적 방법에 의해 단위 데이타 처리를 위한 평균반응시간을 계산한다.

4.1 해석적 모델의 구성요소 정의

N-PFS의 성능을 해석적으로 평가하기 위해서는 시스템 구성요소들 각각의 수행시간을 알아야 한다. 이 때 사용되는 기호들은 〈표 1〉과 같다. 여기서 ct_1, nt, ct_2, st는 큐에서의 대기 시간을 제외하면 항상 일정하지만 $\overline{dt}$는 사용자 수에 따라 실린더의 평균 이동거리가 달라지므로 사용자 수에 따라 $\overline{dt}$가 결정되어야 한다.

디스크 접근시간에는 탐색시간, 회전지연시간, 전송시간이 포함되므로 탐색시간을 α, 회전지연시간을 β, 전송시간을 γ라 하면 $\overline{dt}$는 다음의 수식으로 표현할 수 있다.

$$\overline{dt} = \overline{\alpha} + \overline{\beta} + \overline{\gamma} \quad (1)$$

〈표 1〉 해석적 모델에 사용되는 기호
〈Table 1〉 Symbols in analytical model

기호명	의 미	기호명	의 미
ct_1	로컬 CPU 수행시간	nt	네트워크 전송시간
ct_2	리모트 CPU 수행시간	st	SCSI 버스 전송시간
$\overline{dt}$	디스크 평균접근시간	N	노드 수
d	노드당 디스크 수	n	사용자 수

탐색시간은 $a+b\sqrt{x}$[12]의 형태로 표현되므로 평균 탐색시간은 $a+b\sqrt{\bar{x}}$의 형태가 된다. 여기서 a와 b는 디스크 특성에 따른 상수를 나타내며 $\bar{x}$는 실린더의 평균 이동거리를 나타낸다. 본 연구에서 사용한 디스크의 평균 탐색시간 $\bar{\alpha}$는 다음의 수식에 접근하는 것으로 실험 결과 나타났다.

$$\bar{\alpha}=\begin{cases}4.0+0.50\sqrt{\bar{x}} & (0\le\bar{x}\le 7)\\ 6.3+0.21\sqrt{\bar{x}} & (8\le\bar{x}<48)\\ 6.7+0.45\sqrt{\bar{x}} & (\bar{x}\ge 48)\end{cases} \tag{2}$$

시스템 실측시 하나의 화일을 $N\times d$ 개의 디스크에 스트라이핑시켜 저장하였으므로 탐색거리는 디스크 개수와 사용자 수에 의하여 결정된다. 따라서 탐색거리 $\bar{x}$는 수식 (3)에 의하여 구할 수 있다. 수식 (3)에서 1/3은 전체 거리가 1일 때의 평균 탐색거리가 약 1/3에 근사하다는 사실에 근거한다[12].

$$\bar{x}=\frac{file\_size(cylinder\_no)}{N\times d}\times user\_no\times\frac{1}{3} \tag{3}$$

4.2 해석적 모델

해석적 모델은 동시 사용자의 수가 1, 2인 경우에 대해 유도한 후 이를 n명의 사용자인 경우로 일반화하는 귀납적 방식을 이용하여 유도한다.

4.2.1 동시 사용자 수가 1인 경우

앞 절의 시스템 구성요소들에 대해 큐 지연시간이 전혀 없는 단일 사용자의 평균반응시간의 기대값 $E(Y)$를 수식으로 표현하면 다음과 같다.

$$E(Y)=\frac{1}{N}\left(ct_1+st+\overline{dt}\right)+\frac{N-1}{N}\left(ct_1+nt+ct_2+st+\overline{dt}\right)$$
$$=ct_1+\frac{N-1}{N}\left(nt+ct_2\right)+st+\overline{dt}$$

4.2.2 동시 사용자 수가 2인 경우

입출력에 대한 평균반응시간의 기대값을 구하기 위해, 사용자의 수가 1명인 경우에는 전체 수행시간 중에서 각 요소가 차지하는 비율이 필요 없지만, 사용자가 2명 이상인 경우에는 각 자원을 점유하려는 사용자 간에 충돌이 발생하게 되므로 전체 수행시간에 대한 각 자원의 수행시간 비율이 필요하다. 전체 자원 중 어떤 입출력 요구가 로컬 CPU에 있을 확률을 P_1, 네트워크에 있을 확률을 P_2, 리모트 CPU에 있을 확률을 P_3, SCSI 버스에 있을 확률을 P_4, 특정 디스크에 있을 확률을 P_5라 하면 P_1, P_2, P_3, P_4, P_5의 값은 다음과 같이 구할 수 있다.

$$P_1=\frac{ct_1}{ct_1+\frac{N-1}{N}\cdot nt+\frac{ct_2}{N}+\frac{st}{N}+\frac{\overline{dt}}{N\times d}} \tag{5}$$

$$P_2=\frac{\frac{N-1}{N}\cdot nt}{ct_1+\frac{N-1}{N}\cdot nt+\frac{ct_2}{N}+\frac{st}{N}+\frac{\overline{dt}}{N\times d}} \tag{6}$$

$$P_3=\frac{\frac{ct_2}{N}}{ct_1+\frac{N-1}{N}\cdot nt+\frac{ct_2}{N}+\frac{st}{N}+\frac{\overline{dt}}{N\times d}}\times\frac{1}{N-1} \tag{7}$$

$$P_4=\frac{\frac{st}{N}}{ct_1+\frac{N-1}{N}\cdot nt+\frac{ct_2}{N}+\frac{st}{N}+\frac{\overline{dt}}{N\times d}}\times\frac{1}{N} \tag{8}$$

$$P_5=\frac{\frac{\overline{dt}}{N\times d}}{ct_1+\frac{N-1}{N}\cdot nt+\frac{ct_2}{N}+\frac{st}{N}+\frac{\overline{dt}}{N\times d}}\times\frac{1}{N\times d} \tag{9}$$

어떤 태스크가 전체 시스템 구성요소 중 시스템의 각 구성요소들에 있을 확률이 수식 (5)~(9)와 같을 때 사용자가 2명인 경우에 단일 입출력의 평균반응시간에 대한 기대치 $E(Y)$는 수식 (10)과 같이 나타낼 수 있다. 수식 (10)의 구성요소 중 $\frac{ct_1}{2}$, $\frac{nt}{2}$, $\frac{ct_2}{2}$, $\frac{st}{2}$, $\frac{\overline{dt}}{2}$는

자신을 제외한 다른 사용자가 해당 요소를 사용하고 있을 경우에 기다려야 하는 평균시간을 의미한다.

$$E(Y) = (ct_1 + P_1 \cdot \frac{ct_1}{2}) + \frac{N-1}{N}(nt + P_2 \cdot \frac{nt}{2}) + \frac{N-1}{N}(ct_2 + P_3 \cdot \frac{ct_2}{2}) + (st + P_4 \cdot \frac{st}{2}) + (\overline{dt} + P_5 \cdot \frac{\overline{dt}}{2}) \quad (10)$$

여기서 로컬 CPU의 평균반응시간에 대한 기대값을 $E(Y_1)$, 네트워크의 평균반응시간에 대한 기대값을 $E(Y_2)$, 리모트 CPU의 평균반응시간에 대한 기대값을 $E(Y_3)$, SCSI 버스의 평균반응시간에 대한 기대값을 $E(Y_4)$, 디스크의 평균반응시간에 대한 기대값을 $E(Y_5)$라 한다. 그러면 하나의 입출력에 대한 평균반응시간의 기대값 $E(Y)$는 로컬 CPU, 네트워크, 리모트 CPU, SCSI 버스 및 디스크 등 5개 요소들의 평균반응시간에 대한 기대값을 더한 것과 같다. 즉, $E(Y)$는 수식 (11)과 같이 나타낼 수 있다.

$$E(Y) = E(Y_1) + E(Y_2) + E(Y_3) + E(Y_4) + E(Y_5) \quad (11)$$

4.2.3 동시 사용자 수가 n인 경우

수식의 일반화를 위해 사용자 수가 3인 경우에 CPU를 사용 중이거나 기다리고 있는 사용자 수를 임의변수(random variable) X라 두고 CPU의 평균반응시간에 대한 기대값 $E(Y_1)$을 구하면 수식 (12)와 같다.

$$E(Y_1) = E(Y_1|X=0)P(X=0) + E(Y_1|X=1)P(X=1) + E(Y_2|X=2)P(X=2) = \binom{2}{0}(1-P_1)^2 \cdot ct_1 + \binom{2}{1}P_1(1-P_1)\left(ct_1 + \frac{ct_1}{2}\right) + \binom{2}{2}P_1^2(2ct_1 + \frac{ct_1}{2}) \quad (12)$$

여기서 사용자 수가 n일 경우로 일반화하여 $E(Y_1)$을 전개하면 수식 (13)과 같다.

$$E(Y_1) = (1-P_1)^{n-1} \cdot ct_1 + \sum_{x=1}^{n-1}\binom{n-1}{x}P_1^x(1-P_1)^{n-1-x}\left(x \cdot ct_1 + \frac{ct_1}{2}\right) \quad (13)$$

마찬가지로 $E(Y_2)$, $E(Y_3)$, $E(Y_4)$, $E(Y_5)$를 $E(Y_1)$에서와 같은 방법으로 수식을 전개하면 각각 수식 (14), (15), (16), (17)이 된다.

$$E(Y_2) = \frac{N-1}{N}\left\{(1-P_2)^{n-1} \cdot nt + \sum_{x=1}^{n-1}\binom{n-1}{x}P_2^x(1-P_2)^{n-1-x}\left(x \cdot nt + \frac{nt}{2}\right)\right\} \quad (14)$$

$$E(Y_3) = \frac{N-1}{N}\left\{(1-P_3)^{n-1} \cdot ct_2 + \sum_{x=1}^{n-1}\binom{n-1}{x}P_3^x(1-P_3)^{n-1-x}\left(x \cdot ct_2 + \frac{ct_2}{2}\right)\right\} \quad (15)$$

$$E(Y_4) = (1-P_4)^{n-1} \cdot st + \sum_{x=1}^{n-1}\binom{n-1}{x}P_4^x(1-P_4)^{n-1-x}\left(x \cdot st + \frac{st}{2}\right) \quad (16)$$

$$E(Y_5) = (1-P_5)^{n-1} \cdot \overline{dt} + \sum_{x=1}^{n-1}\binom{n-1}{x}P_5^x(1-P_5)^{n-1-x}\left(x \cdot \overline{dt} + \frac{\overline{dt}}{2}\right) \quad (17)$$

따라서 단일 입출력의 평균반응시간에 대한 기대값 $E(Y)$는 수식 (11)에 수식 (13)~(17)을 대입하여 구하면 수식 (18)과 같다.

$$E(Y) = (1-P_1)^{n-1} \cdot ct1 + \sum_{x=1}^{n-1}\binom{n-1}{x}P_1^x(1-P_1)^{n-1-x}\left(x \cdot ct_1 + \frac{ct_1}{2}\right) + \frac{N-1}{N}\left\{(1-P_2)^{n-1} \cdot nt + \sum_{x=1}^{n-1}\binom{n-1}{x}P_2^x(1-P_2)^{n-1-x}\left(x \cdot nt + \frac{nt}{2}\right)\right\} + \frac{N-1}{N}\left\{(1-P_3)^{n-1} \cdot ct_2 + \sum_{x=1}^{n-1}\binom{n-1}{x}P_3^x(1-P_3)^{n-1-x}\left(x \cdot ct_2 + \frac{ct_2}{2}\right)\right\} + (1-P_4)^{n-1} \cdot st + \sum_{x=1}^{n-1}\binom{n-1}{x}P_4^x(1-P_4)^{n-1-x}\left(x \cdot st + \frac{st}{2}\right) + (1-P_5)^{n-1} \cdot \overline{dt} + \sum_{x=1}^{n-1}\binom{n-1}{x}P_5^x(1-P_5)^{n-1-x}\left(x \cdot \overline{dt} + \frac{\overline{dt}}{2}\right) \quad (18)$$

따라서 단위 시간당 입출력 처리 수는 시스템 구성 요소들의 수행시간과 사용자 수 및 디스크 수가 주어지면 수식 (19)에 의하여 구할 수 있다.

$$IOs = \frac{1000ms}{E(Y)} \times user\_no \quad (19)$$

4.2.4 대용량 데이타를 위한 해석적 모델

이상은 한 번에 입출력하는 데이타의 크기가 작은 경우에 단일 입출력의 평균 반응 시간이다. 한 번에 입출력하는 데이타의 양이 크면 데이타의 전체 입출력시간 중 전송시간이 차지하는 비율이 크다. 데이타 전송시간은 서로 다른 장치 사이의 데이타 전송과 주기억장치 내부에서의 데이타 전송으로 구분되며 서로 다른 장치 사이의 데이타 전송은 전송시간이 단순히 데이타의 크기에 비례하므로 데이타의 크기와 관계 없이 일정하나 주기억장치 내부에서의 데이타 전송은 메모리 복사기능(memcpy)을 사용하므로 전송 데이타의 크기에 따라 초당 전송율이 달라진다.

주기억장치에서의 데이타 이동은 대부분 버퍼 간의 데이타 이동이며 사용자의 요구 크기와 스트라이핑 단위에 큰 영향을 받는다. 따라서 대용량 데이타의 성능 분석을 위해 입출력 모델에서 정의한 수식 (18)의 해석적 방법을 사용하되 버퍼간 데이타 이동을 위한 memcpy(memory copy)의 수행시간을 정밀하게 분석한다.

수식 (18)에서 memcpy를 제외한 전송에 관련된 모든 시간들은 데이타의 크기에 비례하므로 작은 데이타 처리의 성능 분석에서 사용했던 수식을 그대로 이용하면 된다. 그러나 memcpy의 경우는 CPU의 수행시간과 기억장치 접근시간에 대한 의존도가 높으므로 memcpy의 실행모듈과 전송대상 데이타가 CPU 캐쉬에 있는지의 여부에 따라 memcpy의 수행시간은 현격한 차이를 보인다.

따라서 수식 (18)의 구성요소 중 로컬 CPU 수행시간인 ct_1은 작은 데이타 처리의 경우에 데이타 복사시간이 무시할 만큼 적어서 일정한 상수값으로 표현했으나 대용량 데이타의 경우 데이타의 복사시간이 화일시스템의 성능을 크게 좌우하므로 ct_1을 수식 (20)과 같이 표현한다.

$$ct_1 = ct + k \times memcpy\_time / 4KBytes \times rs / 4KBytes \qquad (20)$$

수식 (20)에서 ct는 순수 프로그램 수행시간을 의미하고, memcpy_time은 memcpy 실행모듈과 복사 대상 데이타 모두가 CPU 캐쉬에 있지 않은 경우에 버퍼 간에 데이타를 복사하는 시간을 나타내며, rs는 요청 데이타의 크기(request size)를 나타낸다.

[정의 1] 임의의 단위 블럭의 데이타를 복사하는데 소요되는 시간을 S라하고, memcpy의 실행모듈과 대상 데이타 모두가 CPU 캐쉬 외부에 있는 경우에 단위 블럭을 복사하는데 소요되는 시간을 t라 할때 S를 t로 나눈 상수값을 복사상수 k라 정의 한다. 복사상수는 $0 \le k \le 1$의 값을 가진다.

복사상수는 버퍼 간의 데이타 복사시에 두 개의 버퍼가 모두 CPU 캐쉬에 있지 않은 경우는 복사상수의 값이 1이고 그 외에 대상 버퍼나 복사 프로그램이 CPU 캐쉬에 있는 경우에는 복사상수의 값이 작아진다.

〈표 2〉 Memcpy의 복사시간과 복사상수
〈Table 2〉 Copying time and copying constant in memcpy

분류	4 Kbytes 복사시간	복사율	복사상수(k)
case 1	0.83ms	4.8 MBytes/sec	1.00
case 2	0.52ms	7.69 MBytes/sec	0.62
case 3	0.30ms	13.3 MBytes/sec	0.36
case 4	0.07ms	57.1 MBytes/sec	0.08

〈표 2〉는 memcpy의 실행모듈과 전송 대상 데이타의 캐쉬 여부에 따라 4 KBytes의 데이타를 복사하는 경우의 복사시간과 단위시간당 복사율, 복사상수를 보여준다. case-1은 memcpy 실행모듈과 대상 버퍼 모두가 CPU 캐쉬에 있지 않는 경우를 나타내며, case-2는 memcpy 실행모듈만 CPU 캐쉬 내부에 있는 경우이고, case-3는 memcpy 실행모듈과 대상 버퍼 중 하나가 CPU 캐쉬에 있는 경우이며 case-4는 memcpy 실행모듈과 대상 버퍼 모두가 CPU 캐쉬에 있는 경우를 나타낸다.

따라서 버퍼 간의 복사에 소요되는 시간은 복사상수 k에 의존하며 복사상수 k는 CPU 캐쉬의 크기와 스트라이핑 단위에 의존한다.

5. 해석과 실측에 의한 성능 분석

본 절에서는 해석적 결과와 실측의 결과를 비교 분석하여 해석적 방법의 타당성을 입증하며 해석적 방법과 실측을 결합하여 N-PFS의 병목현상을 일으키는 요소를 분석한다.

디스크 수에 따른 성능을 분석하기 위하여는 노드당 디스크 개수를 표현해야 하는데 노드당 디스크 개수는 다음과 같이 정의한다.

[정의 2] Workstation Cluster의 전체 디스크의 수는 $d(wd_1, wd_2, ..., wd_n)$의 형태로 정의하며 wd_1은 첫 번째 Workstation의 디스크 수를 의미하고 wd_n은 n번째 Workstation의 디스크 수를 의미한다.

5.1 해석에 의한 성능 평가 방법

해석적 방법을 사용하기 위하여 ct_1 및 ct_2의 값을 워크스테이션에서 실측하였으며 나머지 구성요소의 값은 [11]의 시스템 사양과 디스크 사양을 참조하여 계산하였다. 각 구성요소의 수행시간을 정리하면 〈표 3〉과 같다. 〈표 3〉에 있는 각 구성 요소의 수행시간을 수식 (19)에 대입하여 단위시간당 입출력 처리 개수(IOs/sec)를 구하였다. 수식 (19)에 의한 결과는 〈표 4〉의 해석 방법에 나타나 있다.

〈표 3〉 시스템 구성요소의 수행시간
〈Table 3〉 Execution time of system components

구성요소	수행시간	구성요소	수행시간
ct_1	1.5ms	nt	0.4ms/0.5 KBytes
ct_2	0.3ms	nt-100	0.04ms/0.5 KBytes
$\bar{\alpha}$	수식(2)	st	0.05ms/0.5 KBytes
$\bar{\beta}$	6.67ms	*file_size*	139.8 cylinder
$\bar{\gamma}$	0.12ms/0.5 Kbytes	$\bar{x}$	수식(3)

5.2 실측에 의한 성능 평가 방법

화일 입출력은 화일 읽기와 쓰기로 구분되며 대개의 경우 두 가지 모두 유사한 접근 형태를 나타내지만 디스크 특성에 따라 서로 다른 형태를 나타내는 경우도 있다. 본 연구에서는 일반적인 디스크 입출력의 성능 분석에 초점을 두고 있으므로 실험의 단순화를 위해 화일 읽기에 대한 실험만을 수행하였다.

서로 다른 사용자가 서로 다른 화일을 랜덤하게 주어진 회수(1000회)만큼 입출력을 수행하도록 하여 단일 입출력에 대한 평균반응시간을 구한 후 사용자 수와 디스크 수에 따른 단위시간당 트랜잭션 처리 수를 계산하였다. 여기서 평균반응시간은 한 사용자의 입출력 중 앞의 작업이 끝난 후 다음 작업이 시작해서 마칠 때까지의 평균시간을 의미한다.

이러한 방법으로 일정 시간 동안 동일한 수의 사용자가 입출력을 요구하도록 함으로써 사용자수(concurrency)에 대한 입출력 시스템의 성능을 측정한다. 화일 입출력의 성능을 다양하고 세밀하게 분석한 Chen[9]도 이 방법을 사용하여 시뮬레이션을 수행하고 성능을 분석하였으므로 본 연구에서도 이 방법을 사용하여 성능을 분석하였다. 실측에 의한 결과는 〈표 4〉의 실측방법에 나타나 있다.

5.3 해석과 실측의 성능 비교

〈표 4〉는 단위 입출력 크기가 0.5 KBytes이고 디스크의 수가 d(2, 2, 2, 0)와 d(2, 2, 2, 2)인 경우에 해석적 방법과 실측에 의한 단위시간당 입출력 처리 수와, 실측과 해석의 오차를 나타낸다. 대부분의 경우 오차가 5%이하이나 한 경우에만 오차가 6.9%로 나타나 해석적 모델이 정확함을 입증하고 있다.

〈표 4〉 해석적 모델과 실측의 비교
〈Table 4〉 Comparison of results in analytical model and experimentals

디스크 수	사용자 수	해석방법 (IOs/sec)	실측방법 (IOs/sec)	실측/해석 비율	오차
d(2, 2, 2, 0)	1	69.3	69.3	100.0%	+0.0%
	2	116.7	119.5	97.7%	−2.3%
	4	200.4	187.4	106.9%	+6.9%
	8	265.5	259.2	102.4%	+2.4%
	16	321.4	309.9	103.7%	+3.7%
d(2, 2, 2, 2)	1	70.2	69.1	101.6%	+1.6%
	2	118.8	119.4	99.5%	−0.5%
	4	208.1	208.1	100.0%	+0.0%
	8	324.4	320.0	101.4%	+1.4%
	16	389.5	388.2	100.3%	+0.3%

5.4 병목현상 분석 및 최적의 스트라이핑 단위

화일시스템의 처리율은 스트라이핑 단위에 의존하므로 스트라이핑 단위와 입출력 요구 크기를 동일한 크기로 실험을 수행하여 단일 입출력에 대한 평균반응시간을 실측으로 구한다. 그리고 전체 수행시간 중 CPU 수행시간과 디스크 수행시간에 대한 기대값을 수식 (18)에 의하여 구하고 전체 수행시간 중 CPU 수행시간과 디스크 수행시간이 차지하는 비율을 구한다.

〈표 5〉는 수식 (18)에 수식 (20)의 ct_1 변수를 적용했을 경우에 실측과 가장 가까운 복사상수 k값을 구하여 그 시점에서의 CPU 수행시간과 디스크 수행시간의 기대값과 전체 수행시간에 대한 각각의 비율을 구한 것이다.

〈표 5〉 단위 입출력시간 중 CPU 수행시간과 디스크 수행시간의 비율

〈Table 5〉 Ratio of CPU execution time and disk execution time

스트라이핑 단위	사용자 수	실측 시간	CPU 수행 시간 (비율)	디스크 수행 시간 (비율)	대역폭 (Mbytes/sec)
16 KBytes	8	30.4	5.0(16.4%)	23.1(76.0%)	4.2
	16	48.7	9.5(19.5%)	36.8(75.6%)	5.3
32 KBytes	8	38.0	6.1(16.1%)	27.6(72.6%)	6.7
	16	61.0	23.7(38.9%)	32.9(53.9%)	8.4
64 KBytes	8	53.0	7.4(14.0%)	36.7(69.2%)	9.7
	16	82.4	21.5(26.1%)	50.8(61.7%)	12.4
128 KBytes	8	86.1	13.8(16.0%)	54.5(63.3%)	11.9
	16	136.5	46.8(34.3%)	69.1(50.6%)	15.0
256 KBytes	8	309.9	195.6(63.1%)	82.2(26.5%)	6.6

해석적 분석 결과, 복사상수 k 값은 스트라이핑 단위가 32 KBytes 이하인 경우에는 0.2 이하로서 CPU 수행시간이 전체 수행시간에서 차지하는 비율이 26% 이하로 나타났다. 스트라이핑 단위가 64 KBytes, 128 KBytes인 경우, 사용자 수가 8명 이하이면 복사상수가 0.2 이하로 나타났지만 사용자 수가 16명인 경우 복사상수가 0.5 이상으로 증가하여서 전체 수행시간 중 CPU가 차지하는 비율이 급격히 증가하였다.

스트라이핑 단위가 256 KBytes일때에는 사용자 수와 관계 없이 모든 경우에 복사상수가 0.9 이상이 되어 CPU에 병목현상이 발생되어 최대 대역폭은 오히려 스트라이핑 단위가 64 KBytes인 경우보다도 좋지 않게 나타났다.

따라서 최대 대역폭에 도달하는 시점에서 CPU수행시간과 디스크 수행시간의 비율을 살펴보면 최대 대역폭에서의 병목현상을 일으키는 요소를 결정할 수 있다. 〈표 5〉에서 스트라이핑 단위가 64 KBytes 이하인 모든 경우에 전체 수행시간중 디스크 수행시간이 차지하는 비율이 60% 이상으로 화일시스템의 병목현상으로 나타나는 반면 스트라이핑 단위가 128 KBytes인 경우에는 디스크 수행시간이 전체 수행시간의 50% 정도로서 최적의 성능을 나타내는 것으로 분석된다.

그러나 스트라이핑 단위가 256 KBytes인 경우 전체 수행시간 중 CPU 수행시간이 차지하는 비율이 63.1%로서 CPU 수행시간에 병목현상이 발생하여 오히려 화일시스템의 성능이 저하되는 것으로 분석되었다.

6. 결 론

멀티미디어와 과학 계산용 데이타와 같은 대용량 데이타에 대한 수요가 급증함에 따라 소규모의 서버 시스템에서도 고성능 입출력 시스템을 필요로 하게 되었다. 현재까지의 병렬 화일시스템에 관한 연구는 MPP와 같은 대규모의 고성능 시스템을 중심으로 연구되어 왔으나 소규모 서버에서의 화일시스템 설계와 성능 분석에 관한 체계적인 연구는 미흡하였다.

따라서 본 연구에서는 소규모 서버로 사용될 수 있는 워크스테이션 클러스터 환경에서 범용의 UNIX 운영체제를 기반으로 하여 이식성과 융통성이 높은 병렬 화일시스템을 구성하여 성능을 분석하였다.

고성능 컴퓨터 시스템에서는 입출력 향상을 위한 병렬 화일시스템의 계산 오버헤드가 문제되지 않지만 소규모 서버로 사용되는 워크스테이션의 경우에는 병렬 화일시스템의 오버헤드가 높은 비중을 차지함이 본 연구의 결과 밝혀졌다.

본 연구에서는 해석적 방법과 실측의 두 가지 방법으로 성능을 분석하였으며, 본 연구에서는 제시한 해석적 방법과 실측에 의한 결과가 유사하게 나타나 해석적 모델이 정확함을 입증하였다. 또한 해석과 실측을 결합하여 워크스테이션 클러스터에서 대용량 데이타 입출력에 적합한 스트라이핑 단위가 64~128KBytes임을 입증하였다.

대규모 데이타를 입출력하는 경우, 화일시스템 전체의 성능이 메모리 복사의 오버헤드로 인해 제한을 받으므로 본 연구에서는 메모리 복사의 오버헤드를 최소로 하기 위하여 사용자와 병렬 화일시스템이 메모리를 공유하도록 하였다. 공유 메모리를 사용하여 8대의 디스크를 병렬로 구동하였을 때 최대 15.8 MBytes/sec의 처리율을 보였다.

참 고 문 헌

[1] Peter F. C., "Overview of the Vesta Parallel File System," Computer Architecture News, Vol. 21, No. 5, pp.7-14, 1993.

[2] A. L. Drapeau, et. al, "RAID-II: High-bandwidth Network File Server," Proceedings of 21st Annual International Symposium on Computer Architecture, pp.234-244, 1994.

[3] E. L. Miller, R. H. katz, "RAMA: A File system for Massively-Parallel Computers," 12th IEEE Symposium on Mass Storage Systems, pp.163-168, 1993.

[4] P. Pierce, "A Concurrent File System for a Highly Parallel Mass Storage Subsystem," 4th Conference on Hypercubes, Concurrent Computers and Applications, Monterey, CA, pp.155-160, March 1989.

[5] D. G. Feitelson, P. F. Corbett, J. P. Prost, "Performance of the Vesta Parallel File System," Proceedings of 9th International Parallel Processing Symposium, pp.150-158, 1995.

[6] T. W. Pratt, J. C. French, P. M. Dickens, S. A. Janet, jr, "A Comparison of the Architecture and Performance of Two Parallel File Systems," 4th Con. on Hypercubes, pp.161-166, Mar. 1989.

[7] C. S. Freedman, D. J. DeWitt, "The SPIFFI Scalable Video-on-Demand System," Proceedings of the 1995 ACM SIGMOD, pp.352-363, 1995.

[8] J. V. Huber, C. L. Elford, D. A. Reed, "PPFS: A High Performance Portable Parallel File System," International Conference on Supercomputing, pp. 385-394, 1995.

[9] P. M. Chen, D. A. Patterson, "Maximizing Performance in a Striped Disk Array," The 17th Annual International Symposium on Computer Architecture, pp.322-331, May. 1990.

[10] Susan J. L., Marchall Isman, Andy Nanopoulos, "sfs: A Parallel File System for the CM-5," Proceedings of 1993 Summer USENIX, pp.291-304, 1993.

[11] 장시웅, 정기동, "분산 UNIX 환경에서 Shared-Concurrent File System의 설계 및 구현," 한국 정보처리학회 논문지, 제3권 3호, pp.617-630, May 1996.

[12] 김정원, "다중 존 디스크 상에서 비디오 데이터의 효율적인 배치 기법," 부산대학교 석사학위 논문, 1997. 2.

장 시 웅

1984년 부산대학교 계산통계학과 졸업(학사)
1993년 부산대학교 대학원 전자계산학과 졸업(이학석사)
1996년 부산대학교 대학원 전자계산학과 졸업(이학박사)
1986년~1993년 대우통신종합연구소 주임연구원
1996년~현재 동의대학교 전산통계학과 전임강사
관심분야: 병렬 화일시스템, 멀티미디어

정 기 동

1973년 서울대학교 졸업(학사)
1975년 서울대학교 대학원 졸업(석사)
1986년 서울대학교 대학원 계산통계학과 졸업(박사)
1990년~1991년 MIT, South Carolina 대학 교환 교수
1978년~현재 부산대학교 전자계산학과 교수
1993년~현재 부산대학교 부설 컴퓨터 및 정보통신연구소 운영위원
1995년~현재 부산대학교 전자계산소장
관심분야: 병렬처리, 멀티미디어