

그리드 기반 분산시스템을 통한 시스템 성능 평가

장수환^o 박수연 박주현 방영철 공기석

한국산업기술대학교 컴퓨터공학과

{nwangel^o, anisoo, blueii79, ybang, kskong}@kpu.ac.kr

On Evaluation of System Performance for Distributed System using Grid

Soo-Hwan Jang^o, Soo-Yeon Park, Ju-Hyun Park, Young-Cheol Bang, ki-Sok Kong
Dept. of Computer Engineering, Korea Polytechnic University

요 약

그리드는 국가 전략산업 뿐만 아니라, 여러 응용분야에서도 실용성이 많이 대두 되면서, 국내를 비롯한 유럽과 미주에서 많은 연구가 진행 중이다. 본 실험에서 분산된 고성능의 컴퓨팅자원을 활용하여 자체 제작한 파이 프로그램의 실행을 통하여, 클러스터의 작업능력과 클러스터수에 따른 처리 시간의 차이를 분석하여, 그리드 분산시스템의 필요성과 효율성을 확인하고자 한다.

1. 서 론

그리드는 미국의 Ian Foster 가 웹의 차세대 개념으로 제안한 지역적 분산 컴퓨팅을 활용하려는 것이다.[1] 현재 유럽을 선두로 미국과 여러 나라의 연구 기관 및 기업체에서도 자체적인 그리드를 시험 및 운영을 하고 있는 상태이며, 한국도 그리드 기술을 벤치마킹을 통해 기술을 축적한 상태이다. 연구 분야로는 그리드 중간 프로그램인 미들웨어부분과, 분산 알고리즘, 보안, MPI (Message Passing Interface) 부분에서도 연구가 활발이 이루어지고 있다.[2]

국가에서도 미래를 주도할 6개의 첨단 산업인, 정보 기술(IT), 바이오기술(BT), 나노기술(NT), 환경기술(ET), 우주항공분야 (ST), 문화관광 콘텐츠 분야(CT)와 마찬가지로 그리드에 대한 관심과 폭 넓은 지원을 하고있는 중이며, 현재 국내 기업, 연구기관, 대학교에서도 많은 연구와 투자를 하고있는 실정이다.

1.1 그리드의(GRID) 분류와 정의

지역적으로 분산되어있는 고성능의 컴퓨터자원의 공유를 통해서, 대용량의 데이터를 계산 처리할 수 있는 분산 컴퓨팅 기술 이다.

그리드의 분류는 다음과 같다. [3]

- 데이터그리드 : 지역적으로 떨어져있는 대용량의 데이터와 데이터베이스를 하나의 자원으로 통합 및 분석을 할 수 있다.
- 계산그리드 : 분산되어있는 고성능 컴퓨터의 계산자원을 활용해서 작업의 시간을 줄일 수 있다.
- 액세스그리드 : 지역적으로 분산되어있는 그리드 시스템의 어플리케이션을 분산처리 하는 그리드 이다.

그리드 분산(Distributed)이란 동일한 네트워크에 있는 고성능의 컴퓨터를 활용해서, 지역적으로 떨어져있는 컴퓨터의 CPU자원을 이용해, 고성능의 컴퓨팅 시스템을 구축하는 것이다. 그리드 분산에서 사용하는 고성능 컴퓨터를 일반적으로 클러스터라고 한다.[4]

클러스터(cluster)는 슈퍼컴퓨터의 대형화와 고비용에 비해 저렴하면서, 슈퍼컴퓨터의 작업을 분산처리 함으로 시간을 절약할 수 있는 효율을 가지고 있다.

끝으로 글로벌스 툴킷(그림1)은 그리드 서비스를 제공하기 위한 미들웨어로 계산 그리드를 구축하는데 필요한 툴킷이다.

Application					
DRM	High-level Services and Tools			Grid Status	
Cactus	MPI	globusrun	PUNCH	Nimrod/G	Contor_G
Core Services					
GASS	Matacomputing	Glovus	Security	GRAM	
GridFTP	Directory Service	Interface	Replica Catalog	I/O	
Condor	MPI	TCP		UDP	
LSF	PBS	NOE	Linux	AIX	Solaris

그림 1. 글로벌스 툴킷

본 논문에서는 그리드 기반 분산시스템에 관한 연구와 실험을 소개하고자 한다. 2절에서 그리드 시스템 자원 및 실험 방법에 관한 구체적인 내용을 소개하고, 3절에서는 실험결과 및 분석을 통해, 그리드 분산의 장.단점을 알아보며, 끝으로 4절에서는 실험의 결론과 그리드 분산의 필요성 및 향후 연구와 목표에 대해 소개를 한다.

2. 그리드 시스템 자원 및 실험 방법

2.1 그리드 시스템 자원 [5]

그리드 분산시스템을 구성하기위해서 모든 클러스터에 글로벌스 미들웨어를 사용했으며, 분산 라이브러리는 MPICH-G2를, 작업관리자로는 OpenPBS를 사용했다. 그리고 각 클러스터의 상태를 모니터링 하기 위해, Tomcat Server를 구축해서 확인하였다.

그리드 분산을 위한 모든 시스템자원을 다음과 같이 가정을 하였다.

- Red Hat linux7.3
- kernel version: 2.4.18-3
- globus version: 2.2.9
- gcc version: 2.96
- 클러스터는 동일한 모듈을 사용했다.
- 데이터의 계산으로 인한 오버헤드는 없음.

표 1. 그리드 시스템 자원의 정보

Cluster	Speed (MHz)	Node count	Memory (MB)
Server Cluster	1800	3	256
Cluster Net18	1800	1	256
Cluster Net11	866	1	128
Cluster Net23	1800	1	256
Tomcat Server	866	1	128

2.2 그리드 분산 실험 방법

그리드 분산시스템의 실험을 위해서 (그림2)와 같은 시스템을 구성하였다.

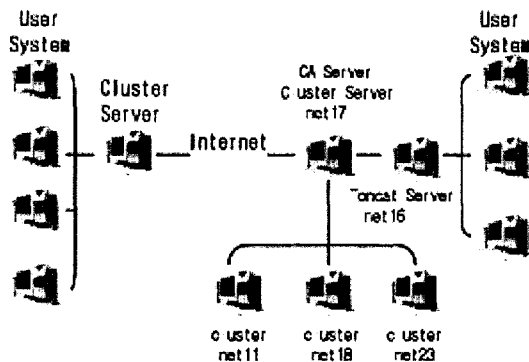


그림 2. 그리드 분산시스템 구성도

실험을 하기 전 확인사항으로 클러스터 자원을 쓸 준비가 되어 있는지, 자원을 사용할 때마다 Proxy가 존재하는지를 먼저 확인을 해야 한다.

서버 클러스터의 역할은 작업의 관리 및 자체 인증 서버로 실험용 인증서를 발급한다.

실험을 위해서 대용량의 데이터 파일이나, 계산 프로그램이 준비 되지 않은 관계로, 자체 제작한 파이(π)프로그램으로 실험을 했다.

우선, 서버 클러스터에서 소스를 작업하고, 각 클러스터에 파일을 옮기기 위해서 Gsincftp를 이용해서, 각 클러스터에 파일을 업로드 한다. 그리고 각 클러스터에서 소스들을 컴파일 하고, 서버 클러스터에서 mpirun 명령을 이용해 RSL파일을 만들고, 실행을 한다.

파이(π)프로그램의 원식에서, 분모는 홀수의 순, 홀수 번째 분수는 - 기호, 짝수번째 분수는 + 기호에서부터, 두 기호가 반복되고, 각 클러스터에서 계산이 많이 이루어질수록 파이의 자리수와 값은 정확한 값을 얻을 수 있다. 또한 클러스터는 자신이 시작할 부분의 분자가 무엇이고, 부호가 무엇인지를 결정할 수 있다.

서버 클러스터에서 실행한 파이프로그램은, 각 클러스터 자원을 활용해서 작업을 수행하며, Tomcat Server에서 네트워크 트래픽 상태(표2), 메모리 상태(표3), CPU 상태(그림3)를 확인할 수 있다.

네트워크 트래픽은 클러스터들의 작업 정보를 패킷의 이동으로 확인할 수 있으며, 순간 패킷의 수치를 아래 표에 Input과 Output으로 분리해서 작성하였다.

표 2. 네트워크 트래픽

Cluster	Input	Output
Server Cluster	4521	5952
Cluster Net18	66	1090
Cluster Net11	4426	5306
Cluster Net23	66	1090

클러스터의 CPU자원을 활용하여 파이(π)프로그램을 수행 하는 동안 시스템의 메모리 상황을 모니터링한 결과를 표3에서 확인할 수 있다. 메모리의 사용을 확인하면 총메모리(Av)에 비해 사용량이 적음을 알수 있다. 위 결과 치는 평균값을 산출한 것이며, Input 데이터의 양과 시스템 사양에 따라서 메모리의 양은 변할 수 있다.

표 3. 클러스터의 메모리상태

Hostname	Av	Free	Used	Buffers
Cluster Net11	126572	118788	7784	16308
Server Cluster	255336	196332	59004	20248
Cluster Net18	255336	165216	90120	25480
Cluster Net23	255336	200984	54352	22412

그림3. CPU 상태에서 Idle(100)상태는 CPU 자원을 사용하지 않고 있는 상태를 말한다. CPU 자원을 사용하게 되면 Use의 그래프가 움직이면서 자원의 수치를 나타내준다. 물론 system 자원도 동작을 수행하면 그만큼의 값을 수치로 나타낸다

서버 클러스터(net17)가 작업을 수행했을 때, 클러스터(net11)와 클러스터(net18)은 시스템 자원을 약 1/2로 사용한다. 클러스터(net23)은 작업수행뿐만 아니라 다른 클러스터들의 작업을 조합하는 작업도 하기 때문에 100% 사용하는 것이다. 작업을 수행하는 동안에 Input 데이터가 들어온다면, 시스템 자원을 100% 사용하고 있지않는, 클러스터로 작업이 분산되어서 실행이 된다.

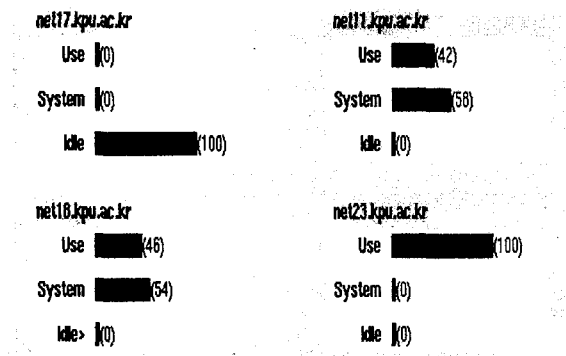


그림 3. 클러스터의 CPU 상태

3. 실험 결과 및 분석

본 논문에서는 클러스터 개수에 따른 파이 프로그램의 계산 속도를 나타낸 그래프를 (그림4)에서 확인할 수 있다. 클러스터의 수에 비례해서 계산 처리속도의 차이는 선형적으로 시간이 줄어들어야 한다. (그림4)을 보면 동일한 작업을 하나의 클러스터에서부터 3대의 클러스터 까지 실험을 했을 때 클러스터 3대에서 작업하는 속도가 한대의 클러스터에서 보다 시간이 줄어드는 것을 알 수 있다.

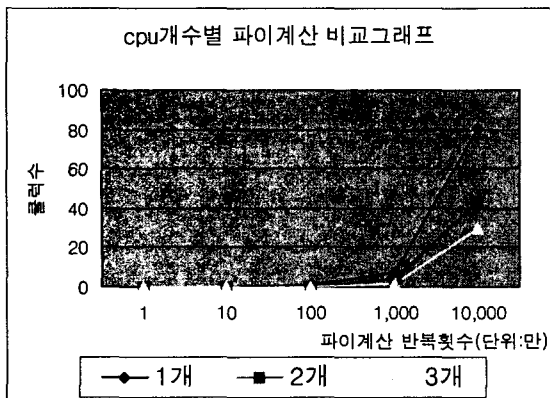


그림 4. CPU 개수별 파이계산 비교 그래프

위 그림에서 보는 바와 같이 작업의 횟수가 100만번 이하에서는 클러스터의 개수와는 상관없이 모두 비슷한 클럭 수의 속도로 계산이 이루어지고, 100만번 이상일 경우에는 클러스터의 개수가 1개일 때보다 2개일 때 클럭수가 약1/2로 줄어든다. 클러스터수가 증가하는 만큼 작업의 속도는 약1/2로 줄어드는 것을 볼 수 있다.

위의 결과로 소량의 데이터 파일이나 DB를 사용할 때는 그리드 분산처리보다는, 하나의 클러스터에서 작업을 하는 것이 빠른 것을 알 수 있고, 대용량의 데이터에서는 그리드 분산 시스템을 활용하면, 데이터의 처리속도와 시간을 절약할 수 있는 장점이 있다.

4. 결론

본 실험에서 MPICH-G2 분산 라이브러리를 사용해서 작업을 수행하는 3대의 분산 클러스터들은, 시스템 자원을 약 1/2로 나누어서 작업을 하는 것을 (그림3)을 통해 확인을 했으며, 실험을 통해서 알 수 있듯이 분산된 그리드 자원의 공유를 통해서 계산된 결과는, 분산되지 않은 자원에서 실험한 결과보다 시간적으로는 단축이 되고, 비용과 경제성도 뛰어나서 앞으로 그리드 분산시스템을 선호하게 될 것이다.[6], 향후 미들웨어 부분과 보안의 대한 많은 연구가 있어야 할 것이다. 목표로는 지역적으로 분산된 그리드 자원을 활용해서 기상 예측분야의 대용량 데이터를 실험하고자 한다.

참고문헌

- [1] 김복주, 유럽 데이터 그리드 테스트 베드의 소개 v1.0 2002.7
- [2] W. Gropp, E. Lusk, and A. Skjellum, Using MPI: Portable Parallel Programming with the Message Passing Interface. MIT Press, 1994.
- [3] 이춘희, 그리드 컴퓨팅(Grid Computing) 2003.1
- [4] <http://www.terms.co.kr/>
- [5] 오영은, 김진석, 그리드 환경에서 MPI 작업을 위한 스케줄링 시스템의 설계 및 구현, 서울시립대학교 컴퓨터학부 2003. 04
- [6] 강경수, 그리드 활용 사례분석, IBM의 그리드 침투 현황 보고서: 마이크로 소프트 그리드 응용연구