그리드 컴퓨팅 환경에서 성능지수를 사용한 작업할당 기법

김영균*, 조금원**, 송영덕**, 나정수**, 고순흠***, 오길호*
*금오공과대학교 컴퓨터공학부
**한국과학기술정보연구원 슈퍼컴퓨팅 센터 슈퍼컴퓨팅응용실
***서울대학교 기계항공공학부
e-mail : ygkim@cescp1.kumoh.ac.kr

The Job Assignment Method using the Performance Index in a Grid Computing Environment

Young-Gyun Kim*, Kum Won Cho**, Young-Duk Song**, Jeong-Su Na**,
Soon-Heum Ko*** and Gil-Ho Oh*
*School of Computer Engineering, Kum-oh National Institute of Technology
**Supercomputing Application Technology Department, Supercomputing Center,
Korea Institute of Science and Technology Information
***School of Mechanical and Aerospace Engineering, Seoul National University

요 약

본 논문에서는 그리드 컴퓨팅 환경에서 연산자원의 성능지수를 사용한 작업할당 기법에 관해 연구하였다. 제안한 연산자원의 성능지수를 사용한 작업할당 기법은 작업을 할당하기 전에 작업을 할당하기 위한 연산자원(프로세서)의 성능지수를 구하고, 이를 바탕으로 작업할당을 수행한다. 연산자원의 성능지수를 사용한 작업 할당 기법은 동적으로 변화하는 그리드 컴퓨팅 환경에서 보다 더 효과적인 작업할당 방법이다. 성능지수를 고려한 작업할당 방법이 고려하지 않은 방법에 비해 3 차원 Euler 방정식을 이용한 CFD 연산 작업의 연산시간을 단축함을 K*Grid 환경에서 실험으로 확인하였다.

1. 서론

최근 그리드 컴퓨팅(Grid Computing)에 관한 연구가 활발히 진행되고 있다. 그리드 컴퓨팅은 문제들을 해결하기 위해 고속의 네트워크로 연결된 연산 자원들(Computational resources)을 활용하는 것이다. 최근 초고속의 통신망과 고성능 컴퓨팅 자원들의 일반화로 단일 사이트의 연산자원을 넘어서는 다중 사이트의 연산 자원들을 효율적으로 이용하는 많은 그리드 컴퓨팅 시스템(Grid Computing System)들이 구축, 운영되고 있다. 이러한 대표적인 그리드 컴퓨팅 시스템으로서 국외의 경우 NEES(Network for Earthquake Engineering Simulation) Grid[1], Griphyn(Grid Physics Network)[2]가 있으며, 국내의 대표적인 K*Grid[3]가 있다.

그리드 컴퓨팅은 지리적으로 분산된 다중의 사이트에 구축된 HPC 와 클러스터 컴퓨터가 초고속 인터넷으로 연결되어 대용량의 연산 자원들을 요구하는 많은 응용들에 대해 연산 자원들을 제공하는 것을 목표로 한다[5]. 그리드 컴퓨팅은 단일 사이트의 HPC, 클러스터 시스템이 제공하지 못하는 대용량, 고속의 연산 자원들을 제공하여 지구규모의 기상예측, 블랙홀의 충돌, 초신성의 폭발과 같은 천문학적 문제, 우주선, 항공기 등의 유체역학 문제, 유전자 분석 등의 계산에 대해 널리 사용이 되고 있다[5,6]. 본 논문에서는 그리드 컴퓨팅 환경에서 3 차원 Euler 방정식을 이용하여 전산유체역학(CFD: Computational Fluid Dynamics) 문제를 계산하기 위해 Cactus 프레임웍과 MPICH-G2 를 사용하는

글로버스 기반의 그리드 컴퓨팅 환경에서 성능지수를 고려한 작업 할당 기법에 대해 연구하였다.

2.관련연구

2.1 그리드 컴퓨팅

그리드는 몇몇 단체나 가상의 조직(Virtual Organization)의 요구를 서비스하기 위해 유지되는 컴퓨터와 기억장치 자원들의 분산된 집합이다[4,9]. 그리드 컴퓨팅은 문제들을 해결하기 위해 네트워크로 연결된 연산 자원들을 활용하는 것이다. 다중의 사이트에 분포되어 있는 많은 고성능의 연산 자원이 초고속의 통신망으로 연결된 그리드 환경에서는 다양한 사용자들의 요구에 따라 동적으로 변화하는 많은 연산 자원들(예를 들어, 프로세서, 보조기억장치, 기타 그리드에 연결된 장치들)이 존재하며, 이러한 자원들을 효율적으로 사용하는 것은 중요하다.

2.2 글로버스와 MPICH-G2

고성능을 필요로 하는 응용 프로그램들의 출현은 지리적으로 분산된 다양한 자원들을 활용하는 능력을 요구한다. 이들 응용 프로그램들은 네트워크로 연결된 가상의 슈퍼컴퓨터(networked virtual supercomputer) 또는 메타컴퓨터들을 형성하기 위해 슈퍼컴퓨터, 대규모 데이터베이스, 보조기억장치, 첨단의 가시화 장치들과 과학적인 장치들을 고속의 네트워크를 사용하여 통합한다[10]. 계산 그리드(Computational Grids)[8,9]는 대규모의 연산, 분산된 데이터의 분석, 원격 시각화(remote visualization)와 같은 목적들을 위해 지리적으로 분산된 자원들을 결합하고 조정 및 통합하여 사용하는 것을 가능하게 한다[5]. 이러한 그리드 환경의 이질성과 동적인 특성은 응용 프로그램의 개발을 어렵게 한다[7]. 글로버스는 분산된 자원들간의 통신, 자원의 위치, 자원의 스케쥴링, 인증과 데이터 접근 등에 있어서 기본적인 능력과 인터페이스를 제공하는 메타컴퓨팅 기반 툴킷(metacomputing infrastructure toolkits)이다[10]. MPICH-G2 는 MPI(Message Passing Interface)의 그리드 컴퓨팅이 가능하도록 한 구현으로서 사용자가 병렬 컴퓨터상에 사용되는 동일한 명령어를 사용하여 동일하거나 다른 사이트에 위치한 다중 컴퓨터(Multiple Computers)들 간에 MPI 프로그램들을 실행할 수 있도록 한다. MPICH-G2 는 글로버스 툴킷 서비스를 사용해서 인증, 허가, 프로세스의 생성, 모니터링, 제어, 통신, 표준 입출력 재지정, 원격 파일 접근 등의 목적을 위해 글로버스 툴킷 서비스를 사용함으로써 이질성을 숨긴다. 결과적으로 사용자는 다른 사이트에 있는 다중의 컴퓨터들을 통해 병렬 컴퓨터상에서 사용되는 동일한 명령어들을 사용해서 MPI 프로그램들을 실행할 수 있다[5]. 그림 1 에서 다양한 글로버스 툴킷 콤포넌트들이 이질성을 숨기고 관리한다. "Fork", "LSF"와 "LoadLeveler"는 서로 다른 지역 스케쥴러이다.

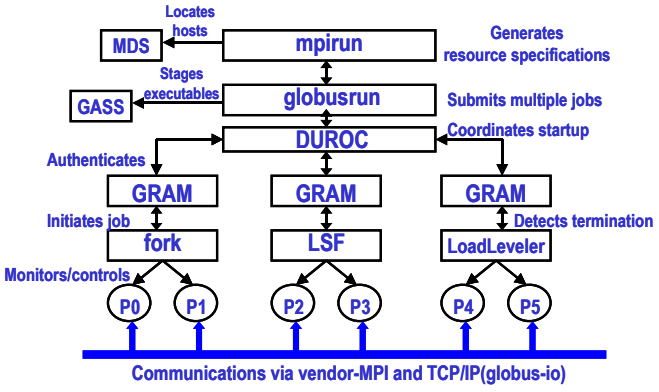

그림 1 MPICH-G2 의 개략도[5]

사용자는 컴퓨터들을 선택하기 위해 MDS(Monitoring and Discovery Service)를 사용한다. 인증을 거친 후, 사용자는 MPI 연산 작업을 실행하기 위해 mpirun 명령을 사용한다. 이 명령은 작업을 기술하기 위해 RSL(Resource Specification Language)를 사용한다. RSL 스크립트에 기초해서, MPICH-G2 는 글로버스 툴킷이 갖는 DUROC(Dynamically-Updated Request Online Coallocator)라이브러리를 호출하여 사용자가 명시한 다양한 컴퓨터들 상에서 응용 프로그램을 스케쥴하고 시작시킨다. DUROC 라이브러리는 GRAM(Grid Resource Allocation and Management) API 와 프로토콜를 사용하여 각 컴퓨터당 하나씩의 부작업(subcomputation)을 시작하고 계속해서 관리한다. GRAM 은 GASS(Global Access to Secondary Storage)를 사용하여 URL 이 가리키는 원격지로부터 실행할 수도 있도록 한다. 또한 GASS 는 응용프로그램이 시작된 후 표준 출력과 표준 에러(stdout 과 stderr)를 사용자의 터미널로 전송한다.

2.3 Cactus 의 특성 및 구조

응용 연구자들의 부가적 노력을 줄이기 위하여 연구자들이 전산 분야의 지식 없이도 발전된 계산 자원 형태 내에서 자신의 해석자를 활용할 수 있게 하고자 하는 연구들이 진행되어 왔다. 특정 문제 해석을 위해 필요한 모든 계산적 편의를 제공하는 컴퓨터 시스템을 문제 풀이 환경(PSE : Problem Solving Environment)이라 지칭하며, Nimrod/G, Triana 및 Cactus 등이 현재 개발되어 있다. 이 중 Cactus 는 기본적으로 천체물리학 연구자들의 공동 연구를 위한 기반으로서 개발되었으나, 천체물리학 연구 분야 뿐 아니라 전산유체역학 분야(CFD)에서도 활용 가능하다. Cactus 는 Albert Einstein Institute(AEI), Washington University Gravity Group, National Center for Supercomputing Applications(NCSA), Argonne National Laboratory 등의 많은 연구기관의 협동 작업의 결과로

만들어진 소프트웨어 프레임 워크(Software Framework) 이다.

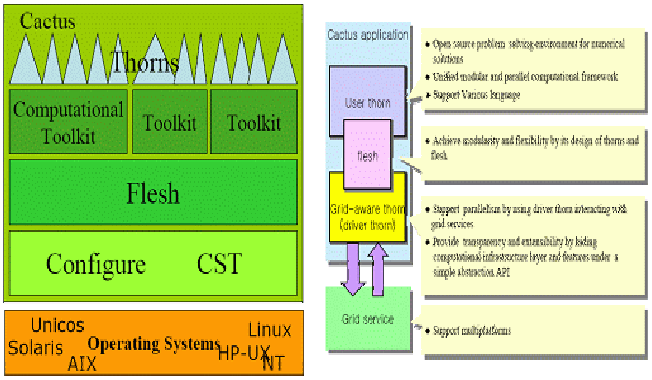

그림 2 Cactus 구조

Cactus 는 과학자와 공학자들을 위해 설계된 공개 문제 해결 환경으로서 모듈구조를 가지고 있으며 서로 다른 구조를 갖는 컴퓨터들에서 병렬 연산이 가능하고 물리학자들과 계산 과학자들의 국제적인 협동 작업으로서 다년간 개발되고 사용되었다[5].

Cactus 는 기본적으로 각종 전산 분야의 기술들이 집약된 Flesh 를 기반으로 하여 각 사용자의 단일 Application solver, 체크포인팅을 위한 도구, 병렬 입출력을 위한 도구, 가시화 도구 등을 사용자의 필요에 따라 첨가하게 된다. 또한 Cactus 는 단일 시스템뿐만 아니라 클러스터나 슈퍼컴퓨터 등 다양한 플랫폼에 구축될 수 있으며 현재 응용 연구자가 사용하는 도구를 쉽게 연결해 사용할 수 있는 환경을 제공한다. 예를 들어 글로버스 툴킷, HDF5 I/O, PETSc 과학 라이브러리, 가시화 도구 등을 연결하여 활용할 수 있다.

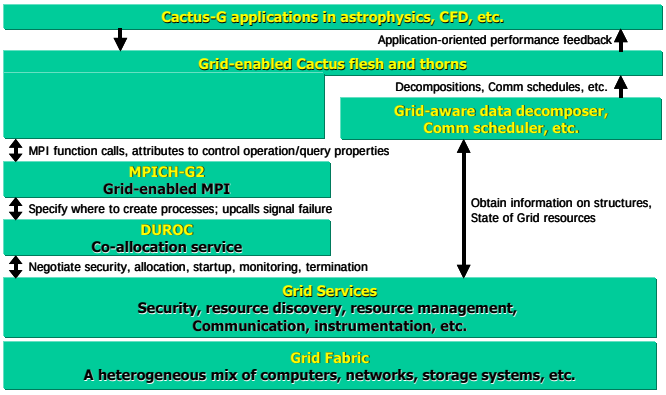

그림 3 MPICH-G2 를 사용하는 그리드 환경에서의 Cactus 구조

3.제안한 작업할당 기법

제안한 작업할당 기법은 작업을 할당하는 사이트의 간략한 성능지수를 나타낸 식 (1)의 P 값을 평가한 후 보다 큰 값을 갖는 사이트에 작업을 할당한다. 작업을 할당하는 사이트의 프로세서의 평균부하 L_c,유휴 프로세서의 개수 $N_{c,cpu}$, 유휴 메모리 $M_{c,freememory}$, 유휴 하드디스크의 양을 $S_{c,freedisk}$, 클러스터 시스템의 가용 프로세서를 연결하는 네트워크의 전송 속도를 $S_{c,connection}$, 가용 프로세서의 비트수를 $B_{c,cpu}$ 라 하면 간략한 성능 지수 P 로서 식 (1)과 같이 나타낸다. α,β,γ 는 가중치로서 P 의 값에 프로세서의 성능, 유휴메모리의 양, 유휴 하드디스크의 양을 어느 정도로 반영할 것인가를 제한하는 가중치이다.(단, 0≤α,β,γ≤1) 일반적으로 연산속도에 미치는 영향이 프로세서의 성능>메모리 용량>하드디스크의 용량으로 가정할 때, α>β>γ의 조건을 만족하는 0≤α,β,γ≤1 의 값을 선택한다.

$$P=\alpha\cdot(1-L_c)\times S_{c,cpu}\times N_{c,cpu}\times B_{c,cpu}\times S_{c,connection} + \beta\cdot M_{c,freememory}+\gamma\cdot S_{c,freedisk} \quad (식\ 1)$$

실험에 사용한 성능지수는 표 2 로부터 성능지수의 기준 값으로 나누어 계산한 표 1 의 값을 사용한다. 성능지수의 기준 값으로서 CPU Clock 속도 $S_{basis,cpu}$=1GHz, 주기억장치의 용량 $M_{basis,freememory}$=512Mbytes, 하드 디스크 용량 $S_{basis,freedisk}$=40Gbytes, 클러스터 내부의 가용 프로세서를 연결하는 네트워크의 전송 속도 $S_{basis,connection}$=1Gbps, 가용 프로세서의 수 $B_{basis,cpu}$=32Bits 를 사용한다.

표 1. Venus, Newcluster 시스템의 기준 값에 대한 성능지수

성능지수	KISTI Venus	KonKuk Univ. Newcluster
S_{cpu}	2.0	4.0(2.0×2)
N_{cpu}	0~63	0~7
$M_{freememory}$	0~1.0	0~2.0
$S_{freedisk}$	0~1.0	0~0.4
$S_{connection}$	0.1	1.0
B_{cpu}	1.0	1.0

4.구현 및 실험

실험을 하기 위해 사용한 3 차원 Euler 방정식은 Fortran 77 로서 Cactus 4.0 beta 12 프레임웍에서 구현 되었다. 실험에 사용한 작업은 3 차원의 Onera-M6 비행기 날개 형상에 대한 압력분포를 계산한다. 이 경우 총 격자점은 그림 5 와 같이 143x33x65 개를 갖는다.

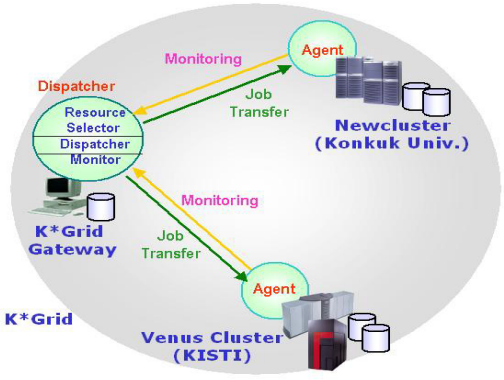

그림 4 Venus Cluster 와 newcluster 에 작업할당

제안한 작업 할당 기법은 그림 4 와 같이 Dispatcher 와 Agent 로 구성되어 있으며 Dispatcher 는 제안한 식 (1)을 사용하여 연산자원의 성능지수 P 값을 계산

한 후 P 의 값이 높은 사이트의 연산자원에 작업을 할당한다. Agent 는 사이트의 부하 등의 정보를 수집하여 Dispatcher 에게 전송하고 Dispatcher 로부터 전송되어 온 작업을 실행시켜 주는 역할을 수행한다.

표 2 K*Grid 중 실험에 사용한 Venus, newcluster 시스템

Organization		KISTI	KonKuk Univ.
Model		Venus	Newcluster
Architecture		Linux Cluster	Linux Cluster
OS		Redhat Linux 7.3	Redhat Linux 7.3
CPU	CPU	Intel Pentium® IV	Intel Pentium® IV
	Clock	2.0GHz	2.0GHz
	#CPU/Node	1	2
	#Node	63	7
	Total	63	14
RAM	#RAM/Node	512MB	1GB
	Total	31.5GB	7GB
Hard Disk	#Hard Disk/Node	40GB	16GB
	Total	40GB+500GB(nfs)	16GB
Network	Login node	venus	newcluster
	Host name	ve001~ve063	node2~node8
	Domain name	gridcenter.or.kr	konkuk.ac.kr
	Interface	Fast Ethernet (100Mbps)	Gigabit Ethernet (1Gbps)

실험은 운영체제로서 Redhat Linux 7.3 을 사용하고 Globus 2.2.4 와 MPICH-G2 1.2.5-1a 를 사용하는 표 2 와 같은 사양을 갖는 클러스터 시스템을 사용하여 K*Grid 에서 수행하였다. 실험에 사용한 작업파일은 실행코드 파일,격자좌표 파일,파라미터 파일,RSL 파일로 구성되어 있다.

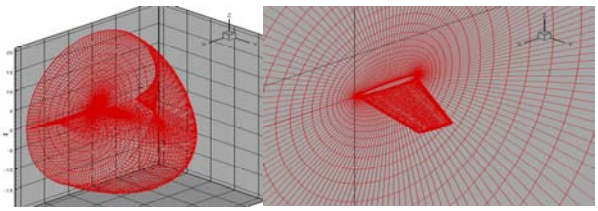

그림 5 실험에 사용한 3차원 Onera-M6 날개 해석 격자계(O-type)

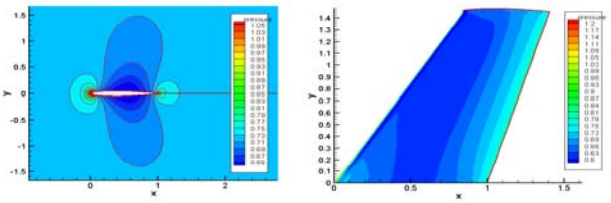

그림 6 압력분포 연산결과(날개 뿌리 단면 및 표면 압력)

실험에 사용한 식 (1)에서 P 의 가중치 값은 α=0.7,β=02,γ=0.1 을 사용하였다. 연산결과는 그림 6 과 같으며 작업 파일의 크기는 15.965Mbytes 의 크기를 갖는다. 작업의 연산회수는 Venus 와 newcluster 각각 8 개의 CPU 를 사용하여 10,000 회 연산을 수행 하였다. Venus 의 성능지수 P_{venus}=13.4113 를 갖으며, 715 분의 연산시간을 갖는다. newcluster 의 $P_{newcluster}$= 22.19145 를 갖으며, 603 분의 연산시간을 갖는다. 따라서, 성능지수 P 의 값이 보다 큰 newcluster 에 작업을 할당하는 것이 연산시간을 단축할 수 있음을 알 수 있다.

5.결론

본 논문에서는 성능지수를 고려하여 성능지수가 보다 우수한 프로세서를 갖는 사이트에 작업을 할당함으로써 작업의 전체 연산시간을 단축함을 보였다. 그리드 컴퓨팅과 같이 연산자원이 동적으로 변화하는 경우 제안한 방법이 보다 더 효과적이다. 차후, 대규모의 사이트를 포함한 연구를 수행할 필요가 있다.

참고문헌

[1] NEESgrid, http://it.nees.org

[2] The Grid Physics Network, http://www.griphyn.org

[3] K*Grid, http://www.gridcenter.or.kr

[4] I. Foster, C. Kesselman, and S. Tuecke, "The Anatomy of the Grid: Enabling Scalable Virtual Organizations", International Journal of High Performance Computing Applications, 15(3). 200-222.

[5] 5] Nicholas T. Karonis, Brian Toonen, Ian Foster, "MPICH-G2: A Grid-Enabled Implementation of the Message Passing Interface", In Proceedings of ASCM/IEEE SC'98 Conference, ACM press, 1998.

[6] Garbrielle Allen, Werner Benger, Thomas Dramlitsch, Tom Goodale, Hans-Christian Hege, Gerd Lanfermann, Andre Merzky, Thomas Radke, Edward Seidel, John Shalf, "Cactus Tools for Grid Applications", Cluster Computing, Vol.4(3), Pages 179-188, 2001.

[7] Garbrielle Allen, Thomas Dramlitsch, Ian Foster, Nicholas T. Karonis, Matei Ripeanu, Edward Seidel, Brian Toonen, "Supporting Efficient Execution in Heterogeneous Distributed Computing Environments with Cactus and Globus", SC2001 November 2001, Denver.

[8] I. Foster. "The Grid: A new infrastructure for 21st century science". Physics Today, 54(2), 2002.

[9] I. Foster and C. Kesselman. editors. The Grid: Blueprint for a New Computing Infrastructure. Morgan Kaufmann Publishers, 1999.

[10] I. Foster and Carl Kesselman, "Globus: A Metacomputing Infrastructure Toolkit", International Journal of Supercomputing Applications, 11(2), 1997.

[11] P. Roe, C. Szyperski, "Transplanting in Gardens: Efficient Heterogeneous Task Migration for Fully Inverted Software Architectures", Proceedings of the Fourth Australasian Computer Architecture Conference, Auckland, New Zealand, January 18-21, 1999.

[12] Garbrielle Allen, David Angulo, Ian Foster, Gerd Lanfermann, Chuang Liu, Thomas Radke, Ed Seidel, John Shalf, "The Cactus Worm: Experiments with Dynamic Resource Discovery and Allocation in a Grid Envrionment", The International Journal of High-Performance Computing Applications and Supercomputing 15(4), Winter, 2001