

전산유체역학 고속 연산을 위한 리눅스 클러스터의 구현 및 성능평가 Building and Benchmarking of Linux Cluster Systems for Computational Fluid Dynamics

이 보 성¹⁾

Bo-sung Lee

요 약

현재 전세계적으로 과학 기술 연산용 리눅스 클러스터에 대한 연구와 개발이 진행되고 있는 추세이다. 본 논문에서는 펜티엄 프로세서와 알파 프로세서 및 고속 이더넷을 이용한 리눅스 클러스터를 구현하고 이를 전산유체역학 분야에 적용하기 위한 선행연구로서 NAS의 병렬 벤치마크 프로그램인 NPB를 사용하여 리눅스 클러스터들의 성능을 비교 분석하였다. NPB 프로그램은 전산유체역학 분야의 해석프로그램에서 널리 이용되는 알고리즘들로 구성된 벤치마크 프로그램으로 이를 이용한 성능 평가의 결과는 향후 리눅스 클러스터에서 실제 전산유체역학 코드들의 성능을 나타내는 중요한 지표가 될 수 있다. NPB 벤치마크 결과 리눅스 클러스터는 고가의 슈퍼컴퓨터에 비해 전산유체역학 분야에서 높은 가격 대 성능비를 보임을 알 수 있으며 이를 통하여 저비용 슈퍼컴퓨팅의 가능성을 제시하고자 한다.

1. 서 론

클러스터링이라는 개념은 IBM이 1960년대에 대형 메인프레임들을 연결하여 비용을 절감하기 위한 시도로서 처음 제안된 이래 1980년대까지는 보편화되지 못하였다. 그러나 1980년대에 이르러 고성능 마이크로 프로세서, 고속 네트워크 및 고성능 분산 컴퓨팅 도구의 개발과 더불어 클러스터 컴퓨팅이라는 개념이 보편화되기 시작하였다. 이와 함께 고가의 슈퍼컴퓨터에서만 수행 가능한 과학계산 및 상업용 어플리케이션에 대한 요구가 증가하면서 저비용 고성능 클러스터 컴퓨터에 대한 필요성이 증대되었으며 지속적인 기술의 발전으로 현재는 시중에서 구할 수 있는 부품을 이용한 클러스터 혹은 네트워크 컴퓨터(PC, 워크스테이션, SMP 등)의 구현이 널리 이루어져서 이를 가능하게 하고 있다[1]. 이 결과 Cray로 대표되는 고가의 슈퍼컴퓨터에서 리눅스(Linux)로 대표되는 무료 소프트웨어와 시중에서 판매되는 저가의 컴퓨터를 네트워크로 연결한 클러스터로 슈퍼컴퓨터의 판도가 변해가고 있다. 이는 전세계 슈퍼컴퓨터의 순위를 매년 발표하는 TOP500 리스트[2]에서도 클러스터가 상위를 차지하고 있다는 점에서도 잘 알 수 있다.

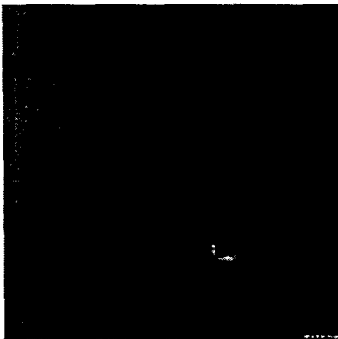


Fig. 1 Avalon Cluster[3]

현재까지 개발되었던 클러스터들은 구축 목적과 방법에 있어서 대단히 다양하다. 단순히 PC나 워크스테이션들을 네트워크로 연결하기만 한 느슨한 형태의 네트워크 클러스터에서부터 SSI(Single System Image)를 구현하고자 하는 연구도 이루어져왔다. 뿐만 아니라 네트워크의 경우도 이더넷에서부터 SAN(System Area Network)을 사용하는 경우까지 다양하며 이에 따라 그 성능 및 비용, 구축방법 등이 천차만별이다. Figure 1에 나타나 있는 Avalon 클러스터는 미국의 Los Alamos 연구소에서 구축한 알파 리눅스 클러스터로 대표적인 계산용 클러스터이다. 따라서 클러스터는 너무나 다양하기 때문에 목적에 적합하게 구축하는 것이 필수적이며 이 경우 성능 및 비용면에서 가장 만족한 성능을 얻게 된다.

본 연구에서는 이러한 관점에서 전산유체역학에 적합한 고속 연산용 리눅스 클러스터의 적용사례를 언급하고 그 성능을 평가함으로써 최대의 가격 대 성능비를 갖는 전산유체역학용 리눅스 클러스터의 구축 방법 및 요구 사항에 대해서 논하고자 한다. 2장에서는 본 논문에서 성능을 평가한 리

눅스 클러스터에 대해서 간략히 논하고 3장에서는 성능 평가에 사용된 대표적인 CFD 벤치마크인 NPB(NAS Parallel Benchmark) 프로그램[4]의 간략한 특징에 대해서 언급할 것이다. 4장에서는 다양한 플랫폼에서의 NPB 성능 평가 결과와 본 논문에서 구축한 리눅스 클러스터의 NPB 성능을 비교 분석할 것이다. 마지막으로 5장에서는 이상의 결과를 바탕으로 전산유체역학용 고성능 계산 클러스터가 가져야할 특징 및 향후 연구 방향에 대해서 논하고자 한다.

2. 리눅스 클러스터

1994년 NASA의 Goddard Space Flight Center(GSFC)에서는 그 당시의 고성능 워크스테이션 1대 비용(\$50,000)

1) 정회원, 리눅스원(주)

이하로 10 GByte의 저장장치와 1 GFlops의 최대 성능을 갖는 시스템을 보유하고자 하였다. 그러나 그 당시에 단일 프로세서 워크스테이션으로 위와 같은 성능을 내기 위해서는 비용이 10배 이상 드는 것이 현실이었다. 따라서 GSFC의 연구진들은 시중에서 구할 수 있는 일반적인 PC를 사용하여 원하는 목적의 컴퓨터를 직접 제작하기로 결정하였다. 이렇게 탄생된 것이 486 DX-4 100Mhz 프로세서를 연결한 최초의 Beowulf[5] 시스템이다. 그 이후 지속적인 프로세서 성능의 증가와 더불어 성능이 향상된 Beowulf 클러스터가 계속 구현되었다. Beowulf 시스템의 목적은 최고의 가격 대 성능비를 나타내는 계산용 클러스터를 구현하는 것이며 이에 따라서 Beowulf는 저비용 슈퍼컴퓨팅을 뜻하는 고유명사로 여겨지게 되었다. 현재까지 구현된 소규모 Beowulf는 수십 종에 이르며 이는 대부분 인텔의 x86 계열의 프로세서 혹은 컴팩의 알파 프로세서를 채택하고 있다. 그리고 네트워크로는 고속 이더넷 혹은 미리넷을 주로 사용하고 있으며 리눅스를 기본 운영체제로 채택하여 비용을 절감하고 있다.

본 연구에서 구현하고 성능을 평가한 클러스터 시스템은 NASA의 Beowulf 시스템과 마찬가지로 저비용 고속 연산을 목적으로 하고 있다. 특히 그 중에서도 전산유체역학 분야의 해석프로그램의 효율적 수행을 목적으로 하고 있다. 이를 위하여 기본적으로 운영체제는 리눅스를 채택하였으며 각 프로세서간의 연결에는 100Mbps 고속 이더넷을 네트워크 스위치로 연결하였다. 각 시스템에는 2개의 NIC(Network Interface Card)를 장착하였으며 첫 번째 NIC는 일반적인 통신(telnet, r-commands, NFS)을 담당하고 두 번째 NIC는 MPI전용으로 사용되도록 구성하여 병렬 해석 프로그램의 수행시간이 시스템 통신에 영향을 받지 않도록 구성하였다.

병렬처리를 위한 라이브러리는 MPI(Message Passing Interface)와 PVM(Parallel Virtual Machine)[6]을 설치하였다. MPI의 경우 다양한 플랫폼을 위해 여러 곳에서 개발되었는데 그 중에서 무료로 사용할 수 있으며 널리 이용되고 있는 MPICH[7]를 설치하였다. 따라서 본 연구에서 구현한 리눅스 클러스터는 개념상 RS6000 클러스터인 IBM SP2와 개념적으로 거의 동일하다고 볼 수 있다. 뿐만 아니라 MPI 프로그램이 수행된다는 관점에서는 어떠한 플랫폼과도 동일한 구조를 갖는다고 볼 수 있다. 그러나 IBM SP2나 CrayT3E와의 가장 큰 차이점은 리눅스 클러스터는 이더넷 기반의 통신을 바탕으로 하였기 때문에 프로세서간의 통신 성능에서는 기존의 슈퍼컴퓨터와는 성능 면에서 많은 차이를 보일 수밖에 없다. 그러나 통신성능은 미리넷 혹은 기가비트 이더넷 등의 채용으로 차이를 줄일 수 있으며 현재 이러한 고속 통신망을 사용한 리눅스 클러스터가 널리 개발되었으며 이를 위한 니바이스 드라이버들도 속속 발표되고 있어 슈퍼컴퓨터와의 성능 차이는 현저히 줄어들 것으로 생각된다.

한편 리눅스 클러스터의 구현에 있어서 중요하게 고려해야 할 점으로서는 사용자 편의성을 들 수 있다. 일반적으로 전산유체역학과 같은 computational science 분야의 연구자들은 computer science 분야의 연구자들과는 달리 컴퓨터 및 네트워크 자체에 대한 지식 및 이해도가 낮다는 점도 고려해야 한다. 즉, 리눅스 클러스터를 구현하는 것 자체에 관심이 있다기보다는 현재 연구중인 과제의 도구로서 리눅스 클러스터를 사용하고자 하는 욕구가 훨씬 강하게 된다. 따라서 사용자의 입장에서는 해석 프로그램의 병렬화를 제외하고서는 기존의 단일 프로세서 워크스테이션을 사용하는 것과 마찬가지로 프로그램을 컴파일하고 수행하기를 원한다. 따라서 고속 연산 이외에 사용자 편의성 증대를 위한 파일 입출력, 사용자 관리, 시스템 유지 보수 등에 대한 고려가 필수적이며 이를 통하여 전체적인 성과를 향상시킬 수 있다.

3. NAS Parallel Benchmarks(NPB)

NASA Ames 센터에 위치한 NAS(Numerical Aerodynamic Simulation)는 NASA의 전산유체역학 및 관련 연구 분야의 고성능 계산을 주로 담당하고 있는 기관으로 고성능 병렬 컴퓨터들의 성능을 비교 평가하기 위한 도구로서 NPB(NAS Parallel Benchmarks) 1.0을 1991년 발표하였다. NPB 프로그램은 실제 전산유체역학 코드에서 출발한 벤치마크 프로그램으로서 슈퍼컴퓨터의 성능을 나타내기에 적합한 표준으로 여겨졌다. NPB 1.0은 공기역학 문제로 부터 개발된 8개의 벤치마크 문제로 구성되어 있는데 이는 다시 5개의 커널과 3개의 CFD 응용문제로 나누어진다. 그러나 NPB 1.0은 하드웨어 공급업체들이 자신들의 하드웨어에 최적화 되어있을 뿐만 아니라 MPI 혹은 HPF(High Performance Fortran)과 같은 표준적인 라이브러리 혹은 컴파일러를 사용하지 않음으로 인하여 소스 코드 수준의 호환성을 제공하지 못하는 문제점이 발견되어 정확한 하드웨어간의 성능 비교가 거의 불가능하였다. 이러한 문제점을 해결하기 위하여 MPI를 사용하여 단일한 소스 코드로 구현된 NPB 2.0이 1995년 발표되었다[8].

NPB 2.0에는 NPB 1.0의 8개의 프로그램 중에서 5개의 커널이 포함되었다. 이 중에서 FT는 3차원 FFT 기반의 spectral 코드이며 MG는 3차원 스칼라 Poisson 방정식의 해를 구하기 위해 다격자(multigrid) 기법을 사용하고 있다. 그리고 LU는 3차원 Navier-Stokes 방정식을 unfactored implicit finite-difference 기법을 사용하여 이산화할 때 나타나는 block lower triangular-block upper triangular 연립방정식의 해를 SSOR(Symmetric Successive Over Relaxation) 기법을 사용하는 구하는 프로그램이다. SP와 BT는 Navier-Stokes 방정식을 approximately factored implicit finite-difference 기법을 사용하여 이산화할 때 나타나는 연립방정식의 해를 구하는 프로그램이다. BT는 5×5 의 block-tridiagonal 연립방정식을 푸는 프로그램이며 SP는 완전 대각화 시에 나타나는 scalar pentadiagonal 연립방정식을 풀게 된다. 각각의 자세한 특징은 참고문헌[8]을 참조하기 바란다. 이상에서 보듯이 5개의 벤치마크

커널은 실제 CFD 해석 프로그램에서 사용되고 있는 가장 핵심적인 부분을 바탕으로 작성되었기 때문에 이들을 이용한 성능 비교 결과가 실제 CFD 해석 프로그램을 사용한 결과를 대표하기에 충분하다고 생각된다. NPB 2.0은 MPI와 Fortran 77을 사용하여 개발된 소스 프로그램으로서 표준적인 메시지 패싱 라이브러리와 과학 기술용 프로그래밍 언어로 작성되었기 때문에 네트워크로 연결된 워크스테이션에서부터 슈퍼컴퓨터에 이르기까지 성능을 객관적으로 비교하기에 적합하다.

4. 성능 평가

본 연구에서는 3장에서 언급한 NPB를 사용하여 3종류의 리눅스 클러스터의 성능을 분석하고 이를 기존의 NPB 결과[9]와 비교하였다. 본 연구에서는 최신 NPB 버전인 2.3에서 A와 B 클래스 문제들을 사용하여 각 클러스터의 성능을 비교하였다. 3장에서 언급한 5가지의 커널 이외에 CG(Conjugate Gradient) 프로그램도 성능 비교에 포함되었다.

	SNU M-Cluster	KORDIC SMP Cluster	Linux One 21264 Cluster
노드수/CPU 수	16/16	8/16	8/8
CPU/Node	P-III 450MHz	Dual P-III 450 Mhz	Alpha 21264 466 Mhz
Memory/Node	256 MB	256 MB	128 MB
HDD/Node	8.4 GB IDE	8.4 GB IDE	9.1 GB SCSI
NIC	Dual Intel 100 Mbps	Dual Intel 100 Mbps	Dual Digital 100 Mbps
Switch	Intel 410T Switch	Intel 410T Switch	Intel 410T Switch
MPI	MPICH 1.1.2	MPICH 1.1.2	MPICH 1.1.2
컴파일 옵션	g77 -O3	g77 -O3	fort -O4 -fast
Linux Version	Alzza Linux 6.0	Alzza Linux 6.0	Alzza Linux 6.0
Kernel Version	2.2.12-1	2.2.12-1smp	2.2.10-4

Tab. 1 Specification of Cluster Systems

Table 1에 성능 비교에 사용된 클러스터의 사양 및 MPI의 종류, 그리고 NPB 프로그램 컴파일에 사용된 컴파일러와 컴파일 옵션이 나타나있다. SNU M-Cluster는 서울대학교 항공우주공학과 공력시뮬레이션 연구실에 설치된 리눅스 클러스터로서 단일 프로세서 16노드로 구성되어 있다. KORDIC SMP 클러스터는 KORDIC 슈퍼컴퓨팅 사업단에 설치된 듀얼 프로세서 8노드 SMP 클러스터이며 Linux One 21264 클러스터는 리눅스원(주)에 설치된 알파 21264 8노드 클러스터 시스템이다.

Benchmark Code	Class A	Class B	Class C
MG (Multigrid)	256 ³	256 ³	512 ³
CG (Conjugate gradient)	14000	75000	150000
FT (3-D FFT PDE)	256 ² ×128	256 ² ×512	512 ³
LU (LU Solver)	64 ³	102 ³	162 ³
SP (Pentadiagonal Solver)	64 ³	102 ³	162 ³
BT (Block tridiagonal solver)	64 ³	102 ³	162 ³

Tab. 2 NAS Parallel Benchmarks Problem Sizes

Table 2는 NPB의 문제 크기를 나타내는 표이다. 본 연구에서는 시스템의 기억용량 제한내에서 Class A 와 Class B에 대한 성능 평가를 수행하였다. Class A와 Class B의 결과가 거의 동일한 값을 나타내는 경우가 많아서 이 경우에는 지면 관계상 Class A에 대한 결과만을 보일 것이다.

4.1 MG (Multigrid)

Figure 2-1과 2-2는 MG 코드의 성능 평가 결과이다. 그림에서 볼 수 있듯이 21264 클러스터의 경우 4개의 프로세서에서는 CrayT3E-900에 근접하는 결과를 보이고 있다. MG의 경우 1, 4, 9 와 같이 제곱개의 프로세서를 요구하기 때문에 21264 클러스터의 경우 4개 프로세서 이외에는 결과를 얻을 수 없었다. M-Cluster나 KORDIC SMP Cluster의 경우는 CrayT3E-1200의 약 35% 정도의 성능이 있음을 알 수 있었다. MFlop/s/proc 그래프는 클러스터의 병렬화 효율을 나타내게 되는데 프로세서 수의 증가에도 불구하고 MFlop/s/proc의 그래프가 직선에 가까울수록 네트워크에 의한 속도 감소가 작다는 것을 의미한다. 한가지 특이한 사항은 M-Cluster와 KORDIC SMP 클러스터의 16 프로세서에서의 성능인데 8개의 프로세서까지는 두 클러스터가 거의 동일한 성능을 보인 반면 16 프로세서에서

는 SMP 클러스터의 성능이 단일 프로세서 클러스터에 비해 성능이 나쁘게 나타난다. 그 이유는 SMP 머신에서 노드간 bandwidth/proc는 노드 내의 프로세서 수의 증가에 비례하여 감소하기 때문이다. 뿐만 아니라 리눅스 커널의 SMP 지원이 강화되고 있음에도 불구하고 여전히 성능 향상의 여지가 남아있기 때문이기도 하다.

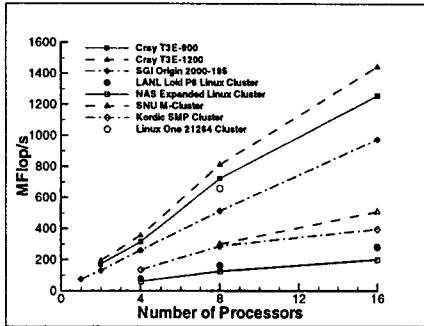


Fig. 2-1 MFlop/s of MG Class A

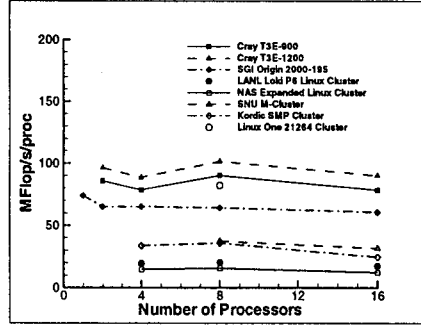


Fig. 2-2 MFlop/s/proc of MG Class A

4.2 CG (Conjugate Gradient)

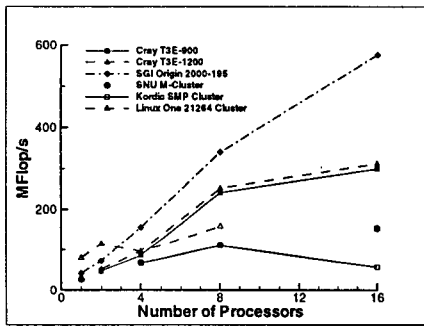


Fig. 3-1 MFlop/s of CG Class A

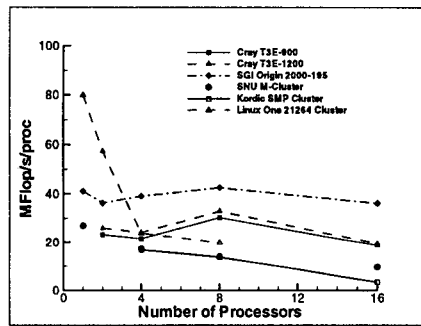


Fig. 3-2 MFlop/s/proc of CG Class A

CG는 다른 5개의 프로그램에 비해 노드간 통신 성능에 크게 영향을 받는다. 이는 매 반복계산마다 MPI_ALL_REDUCE 함수를 호출하는데 이로 인하여 통신부하가 상당히 크게 된다. 따라서 전용 통신 장치를 갖는 슈퍼컴퓨터에서도 프로세서 수의 증가에 따라서 성능이 저하되는 특징을 보이는데 리눅스 클러스터에서는 이러한 현상이 더욱 두드러진다. 따라서 CG를 사용하는 CFD 해석 프로그램의 경우 네트워크 장비에 대한 고려가 필요하다.

4.3 FT (3-D FFT PDE)

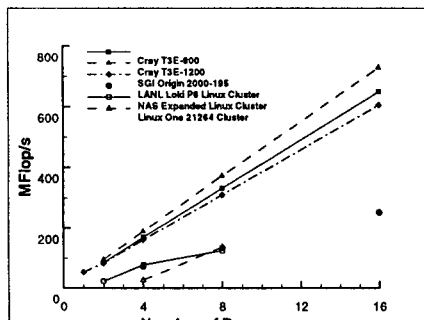


Fig. 4-1 MFlop/s of FT Class A

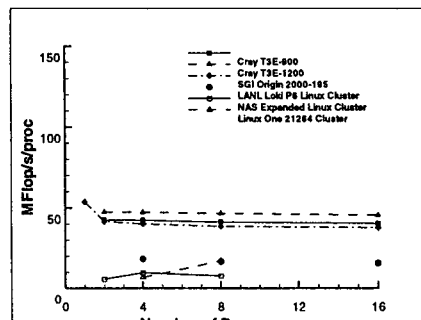


Fig. 4-2 MFlop/s/proc of FT Class A

FT의 경우 SNU M-Cluster와 KORDIC SMP 클러스터에서는 결과를 얻지 못하였다. 그 이유는 두 클러스터에 탑재되어있는 포트란 컴파일러인 g77이 지원하지 않는 형태의 포트란 코드였기 때문이며 이는 g77의 한계이다. 이는



상용 컴파일러를 사용함으로써 해결될 수 있다. 21264 클러스터의 경우 메모리의 양이 A Class를 수행하기에도 부족하여 메모리 swapping이 발생하였고 이로 인하여 정확한 성능치를 얻지 못하였다.

4.4 LU (LU Solver)

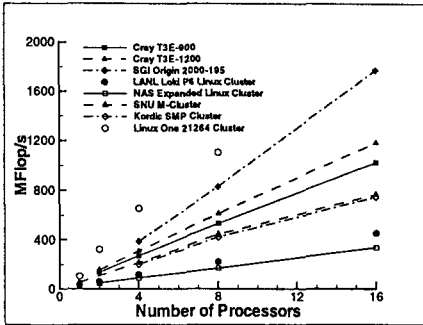


Fig. 5-1 MFlop/s of LU Class A

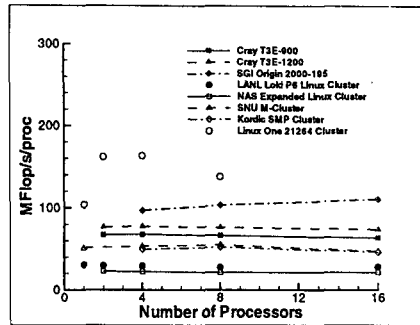


Fig. 5-2 MFlop/s/proc of LU Class A

LU는 다른 프로그램과는 대단히 작은 크기의 메시지를 대량으로 주고 받는다는 특징이 있다. 따라서 하드웨어 성능보다는 오히려 MPI 자체의 작은 크기의 메시지 통신 성능에 민감하게 반응하게 된다. 본 연구 결과에서 볼 수 있듯이 LU의 경우 21264 클러스터가 가장 좋은 성능을 보이고 있으며 SNU M-Cluster와 KORDIC SMP 클러스터도 다른 벤치마크 프로그램에 비해 월등한 성능을 보이고 있다.

따라서 LU의 경우 하드웨어의 성능 비교 도구보다는 MPI의 작은 크기의 메시지 통신 성능을 비교하기에 더욱 좋은 도구라고 할 수 있을 것이다.

4.5 SP (Pentadiagonal Solver) 와 BT (Block Tridiagonal Solver)

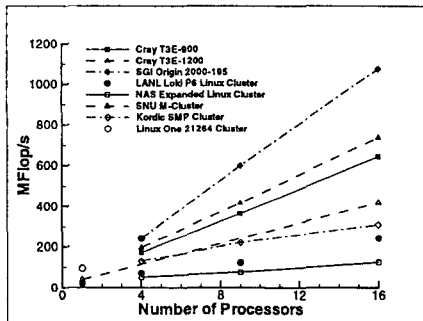


Fig. 6-1 MFlop/s of SP Class A

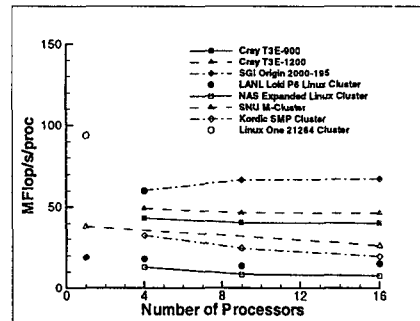


Fig. 6-2 MFlop/s/proc of SP Class A

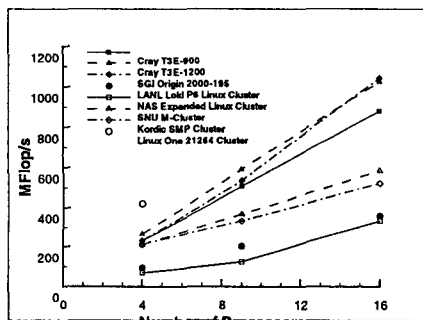


Fig. 7-1 MFlop/s of BT Class A

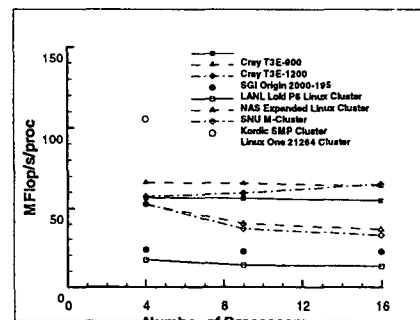


Fig. 7-2 MFlop/s/proc of BT Class A

SP와 BT는 x, y, z 방향으로 순차적으로 연립방정식을 풀어간다는 점에서 유사한 구조를 갖는다. NPB에서는 각 방향의 연립방정식을 풀 때 multi-partition 기법을 사용하는데 이는 큰 크기의 메시지를 소량으로 주고 받는다는 점에서 LU와는 대조적이라고 할 수 있다. 따라서 SP와 BT의 경우 프로세서 수의 증가에 따라서 리눅스 클러스터에서의 병렬화 효율이 감소하는 경향이 있다.

Figure 6-1과 6-2는 SP의 결과를 나타내고 있는데 그림에서 볼 수 있듯이 SNU M-Cluster와 KORDIC SMP 클러스터뿐만 아니라 NAS와 LANL의 리눅스 클러스터에서도 이와 같은 성능 감소가 관찰된다.

Figure 7-1과 7-2는 BT 결과를 나타내는 것으로 SP와 유사한 경향을 보이고 있다. 두 경우 모두 4개의 프로세서에서는 SNU M-Cluster 및 KORDIC SMP 클러스터의 성능이 다른 슈퍼컴에 비해서 그다지 떨어지지 않는 편이나 프로세서의 수가 증가하면서 성능차이가 현저함을 알 수 있다.

5. 결 론

본 논문에서는 전산유체역학 분야의 고속 연산을 위한 리눅스 클러스터의 구현 및 성능에 대한 평가를 수행하였다. 리눅스 클러스터의 성능 평가를 위하여 NPB 프로그램을 이용하였다. NPB 프로그램의 수행결과 모든 프로그램에서 리눅스 클러스터가 기존의 슈퍼컴퓨터에 비해 현저히 높은 가격 대 성능비를 보임을 알 수 있었다. 그러나 NPB의 각 프로그램의 특징에 따라서 리눅스 클러스터에 더욱 적합한 형태의 문제가 있는 반면 어떤 경우에는 상대적으로 낮은 성능을 나타낸 경우도 있었다. 대체적으로는 리눅스 클러스터가 기존의 슈퍼컴에 비하여 현저히 높은 가격 대 성능비를 보이지만 CG(Conjugate Gradient)의 경우는 프로세서의 수가 증가함에 따라서 성능 감소가 현저하며 따라서 이러한 형태의 해석프로그램을 주로 수행하는 경우에는 리눅스 클러스터의 구축시 네트워크에 대한 투자가 고려되어야 할 것으로 생각된다. BT와 SP의 경우는 CG 만큼의 성능 감소는 아니지만 프로세서 수가 증가함에 따라서 전체적인 프로그램 성능이 감소함을 알 수 있었다. 그러나 두 경우 모두 네트워크 장비의 가격 대 성능비가 현저히 향상됨에 따라서 문제점이 상당히 개선될 수 있을 것이다. FT의 경우 리눅스 클러스터의 경우 공개 컴파일러를 사용하였을 경우 문제가 발생하였다. 이와 같은 경우에는 컴파일러에 대한 고려가 선행되어야 할 것이다.

한편 단일 프로세서로 구현된 클러스터와 SMP 클러스터의 성능을 비교한 결과 단일 프로세서 클러스터의 성능이 SMP 클러스터에 비해 다소 낮다는 것을 알 수 있었으며 이를 통하여 SMP 클러스터의 경우 bandwidth를 노드의 프로세서 수에 비례하여 늘여야한다는 결론에 도달하였다. 이에 대해서는 리눅스 커널 및 디바이스 드라이버에 대한 수정과 개발이 필요할 것으로 생각된다.

리눅스 클러스터는 현재 모든 과학 기술 분야에서 현재 널리 적용되고 있으며 전산유체역학 분야도 예외가 아니다. 그러나 막연히 리눅스 클러스터를 구축한다고 해서 연구의 성과가 바로 나타나지는 않으며 실제 전산유체역학 분야에서 원하는 성과를 얻기 위해서는 리눅스 클러스터에서 수행하고자 하는 해석프로그램의 알고리즘 특성과 구현하고자 하는 클러스터의 하드웨어의 특성이 적합한지에 대한 많은 선행연구가 수행되어야 하며 이를 바탕으로 전산유체역학 분야의 고성능 해석이 가능할 것으로 생각된다.

감사의 글

본 논문의 성능 평가를 위한 시스템 사용에 대해서 서울대학교 항공우주공학과 공력시뮬레이션 연구실과 KORDIC 슈퍼컴퓨팅 사업단에 감사드립니다.

참고문헌

- [1] Rajkumar Buyya, *High Performance Cluster Computing*, Prentice Hall, 1999
- [2] Top 500 Supercomputer Sites, <http://www.top500.org/top500.list.html>
- [3] Avalon Cluster, <http://cnls.lanl.gov/avalon/>
- [4] NAS Parallel Benchmarks, <http://www.nas.nasa.gov/Software/NPB/>
- [5] Beowulf Project at CESDIS, <http://www.beowulf.org>
- [6] Parallel Virtual Machine, <http://www.epm.ornl.gov/pvm/>
- [7] The Message Passing Interface(MPI) Standard, <http://www-unix.mcs.anl.gov/mpi/>
- [8] David Bailey *et al.*, The NAS Parallel Benchmarks 2.0, Report NAS-95-020, December, 1995