다양한 PC 클러스터 시스템 환경에서 CFD 코드의 성능 분석

Performance Analysis of a CFD code in the Several PC Cluster System

조 금 원[1]*, 홍 정 우[2], 이 상 산[3]
Kum Won Cho, Jungwoo Hong and Sangsan Lee

At the end of 1999, the TeraCluster Project in the KISTI Supercomputing Center was initiated to explore the possibility of PC clusters as a scientific computing platform to replace the Cray T3E system in KISTI by 2002. Since actual performance of a computing system varies significantly for different architectures, representative in-house codes from major application fields were executed to evaluate the actual performance of systems with different combination of CPU, network and network topology. As an example of practical CFD(Computational Fluid Dynamics) simulations, the flow past the Onera-M6 wing and the flow past a infinite wing were simulated on a clusters of Linux and several other hardware environments.

1. 서 론

한국과학기술정보연구원(KISTI) 슈퍼컴퓨팅센터에서는 TeraCluster로 명명된 프로젝트를 수행 중에 있다. 본 프로젝트의 목적은 2002년도에 KISTI에서 운영중인 Cray T3E(450MHz EV5 알파 프로세서)를 대체하기 위한 PC 클러스터를 개발하는 것이다. 이의 과정으로 표 1과 같은 다양한 PC 클러스터 시스템들을 구성하여 성능을 분석하였다.

표 1. 시스템 구성

	CPU	OS	네트워크			CPU 수	협력사	비고
			종류	Latency [Pingpong] (μsec)	Bandwidth [peak] (Mbyte/sec)			
DS10	466MHz EV6 알파프로세서	Linux (kernel 2.2.0)	Fast Ethernet	140	8.9	65	Linux1, Compaq Korea	- CPU 1개 Front-End로 사용 - Latency와 Bandwidth는 PMB[1]측정 결과임
UP2000	667MHz EV6 알파프로세서	"	Myrinet	13	139.7	65	Zion, 삼성전자, 삼성물산	
Intel III (SCI)	667MHz Pentium III 인텔프로세서 Tyon Thunder 2500	"	SCI	6	24.93	17	Scali, Dolphin	
Intel III (Myrinet)	667MHz Pentium III 인텔프로세서 Tyon Thunder 2500	"	Myrinet	11	141.7	17	삼성물산	
Cray T3E	450MHz EV5 알파프로세서	UNICOS/ mk	3D Torus	13	157	128		

1) 한국과학기술정보연구원 슈퍼컴퓨팅센터 선임연구원
2) 한국과학기술정보연구원 슈퍼컴퓨팅센터 연구원
3) 한국과학기술정보연구원 슈퍼컴퓨팅센터 선임연구원

구성된 시스템에 대한 성능분석은 크게 두 분야로 나뉘어 수행되어졌다. 첫 번째는 시스템의 기본 특성을 파악하기 위한 전산분야 시험으로 시스템간의 통신 및 MPI 라이브러리 등의 기본 성능이 분석되었다. 이를 위해 PMB(Pallas MPI Benchmarks)[1]와 NPB(NAS Parallel Benchmarks)[2] 시험이 수행되어 Cray T3E에서의 결과와 비교되었다[3]. 두 번째로 주요 슈퍼컴퓨터 사용자 그룹 분야[구조, 기상, 물리, 열유체, 화학]에서 실제 사용되는 코드를 이용하여 성능을 분석하였다[3]. 이 중에서 본 연구는 열유체 분야 코드를 구성된 시스템에 적용하여 특성을 파악하였으며 마찬가지로 Cray T3E에서의 결과와 비교하였다. 성능분석을 위하여 Onera M6 날개와 3차원 유한 날개주위의 천음속 유동장을 해석하였다.

2. 병렬화 기법

클러스터 시스템의 병렬 성능 분석에 대한 주요한 시험은 확장성의 검토이다. 이의 검증을 위해 이루어진 각 부분별 특징은 다음과 같다.

2.1 영역분할

전체 해석영역 격자계를 미리 분할하고 각 프로세서가 해당 영역 격자계를 입력으로 받는 SPMD(Single Program Multiple Data) 기법을 사용하였으며, 이러한 병렬처리를 행하는데 있어서 다음의 두 가지 경우를 시험하였다.

경우 1: 전체는 동일한 격자수를 유지하고 CPU가 증가할 때 해당하는 격자수는 감소하게 하는 방법(Onera M6 날개가 해석 대상이며, C1으로 이름함)
경우 2: 단일 CPU에 해당하는 격자수가 항상 동일하게 유지하도록 전체 격자수를 증감하는 방법(3차원 유한 날개가 해석대상이며, C2로 이름함)

C1은 CPU수가 증가할 때 자료교환이 상대적으로 증가하게 되므로 주어진 시스템에 대해 계산에 따른 통신성능을 비교하게 되며, C2는 실제적으로 격자수가 증가함에 따라 병렬 성능이 확장성을 가지고 이루어지는지를 판별하게 된다.

2.2 프로그램 구성

유동 해석 과정은 입출력 단계, 해석 사전 단계, 해석 단계, 영역간의 자료교환단계, 수렴성 확인 단계, 경계조건처리 단계 그리고 실제 값을 계산하기 위한 후처리 단계로 구성되어져 있다. 각 과정에 대한 병렬화 과정을 설명하면 다음과 같다.

2.2.1 입출력 과정

단일 또는 다중 격자계를 Front-End 노드에 위치시킨 후(각 노드들은 100Mbps의 Fast Ethernet을 사용하여 NFS(Network File System)으로 연결되었음), 해당 노드들이 해당 자료를 입력하는 형태를 취하게 하였다. 이 방법은 입력자료를 분할시킬 필요 없이 쉽게 자료를 보관 및 분산시킬 수 있다. 그리고 노드 수가 적은 경우 또는 네트웍 성능이 좋은 경우에는 그 시간이 적게 걸리는 반면 노드 수가 많아질 경우 현재의 네트워크 환경에서는 성능이 저하될 것임을 예상할 수 있다. 대안의 방법이 각 노드에 해당되는 격자계 입력 파일을 만든 후 이를 서로 독립적으로 입력하는 방식이다. 각 자료의 출력은 해당 노드가 자신에 속한 격자계와 해석결과를 분리해서 출력하도록 하였다. 표 1의 각 시스템에 대한 입출력 과정의 성능 분석 결과는 참고문헌[4]의 결과와 매우 유사하므로 본 논문에서는 이를 나타내지 않았다.

2.2.2 해석 전후 처리 단계 및 수렴성 확인 단계

본 연구에서는 해석 시 기본이 되는 변수들(예: 각 방향 격자 수, 노드당 다중블럭 수, 전체 격자계 수, 각 격자계의 해당 노드 번호 등)을 공통적으로 가지고 있도록 하였다. 양력 계수를 포함한 힘과 모멘트 계수를 계산하기 위해 namelist 입력 자료로 주어진 경계조건에서 해당되는 격자 경계만을 뽑아내어 계산한 후 MPI_ALLREDUCE[5]를 사용하여 계산하였다. 그리고 각 노드에서 계산한 RMS 오차를 MPI_ALLREDUCE로 계산하여 전체 수렴성을 확인하도록 하였다. 표 1의 각 시스템에 대한 전후처리 과정의 성능 분석 결과는 2.2.1의 경우와 같이 참고문헌[4]의 결과와 매우 유사하므로 본 논문에서는 이를 나타내지 않았다.

2.2.3 경계조건 처리

본 연구에서 병렬 경계처리를 위해 해당 경계면에 대해 상대편 경계면을 지정하게 함으로써 병렬 경계 처리가 격자의 형태에 무관하게 이루어지게 하였다. 또한, 임의의 CPU가 단일 및 다중블럭(multi-block) 격자를 포함할 수 있게 하였다. 실제로 2.1절의 경우에 64개의 다중블럭을 구성한 후 주어진 CPU는 각각 다수의 격자계를 포함하도록 하였다.(예로 4개의 CPU를 사용할 경우 1개의 CPU는 각각 16개의 다중블럭 격자를 포함한다.)

2.3 자료교환 방법

해석자(Solver)의 정확도를 유지하도록 자료교환을 수행했으며[6], 자료교환방법으로 SEND/RECV 방법(경우 3: C3로 이름함)과 IRECV/SEND 방법(경우 4: C4로 이름함)을 시험하였다. 이를 표 2에 나타내었다. 각 경우의 알고리즘을 설명하면 다음과 같다. C3는 PingPong[1]시험과 유사 한 형태를 가지며, C4는 PingPing[1]의 형태를 갖는다. C4의 라이브러리 사용 형태를 좀 더 세분하게 보면 표 3과 같다.

표 2. 자료교환 방법의 예

경우 3(C3)	경우 4(C4)
do n = 1,nsurf call MPI_BARRIER call MPI SEND call MPI RECV end do	do n = 1,nsurf call MPI IRECV call MPI SEND call MPI_WAIT end do

표 3. IRECV/SEND 자료교환 알고리즘

```
call MPI_IRECV(buf,icount,MPI_REAL,ncpu,tag,comm,ireq,ierr)
call MPI_SEND(buf,icount,MPI_REAL,ncpu,tag,comm,ierr)
call MPI_WAIT(ireq,istatus,ierr)
```

표 3에서 point-to-point 자료교환의 특징을 이용하면 ncpu 값은 각 경우에 동일하게 된다. 위의 과정은 Cray T3E와 클러스터에서 자료교환의 실제적 특성을 파악하는데 중요한 역할을 할 것이라 생각되며, 이것이 일반적으로 PC cluster에서 병렬 처리 코드를 작성할 때 대두되는 병렬 성능 및 확장성을 보장하기 위한 방법을 제시하여 줄 것으로 판단된다.

3. 병렬 성능 분석

3.1 해석 예제

2.1절에서 언급한 C1의 격자계를 그림 1에 나타내었으며, 단일 블록의 격자 수는 17×17×9 개, 총 블록수는 64개로 전체 격자수는 166,464개이다. 해석에 사용된 형상은 O-H 격자 형태의 ONERA M6 날개[7]로 자유흐름 마하수가 0.84 받음각이 3.06도인 경우이다. 2.1절에서 언급한 C2에 대한 단일 CPU의 격자계는 그림 2에 나타나 있다. C2에서 단일 블록의 격자수는 140,481 (129×33×33)개로 1개의 CPU에서 해석되며, 64개 CPU를 사용할 경우의 격자수는 1개의 CPU를 사용한 격자수에 비해 64배 많은 8,990,784개이다. 해석에 사용된 형상은 C1과 같이 O-H 격자형태로 ONERA M6 날개 뿌리(root)의 에어포일 형상을 갖고 스팬(span)방향으로 유한 길이를 갖는 날개이다. C2의 예제는 Cray T3E에서 프로그램 메모리 한계로 해석이 수행되지 못하였으며, 구성된 PC 클러스터에서만 해석이 수행되었다

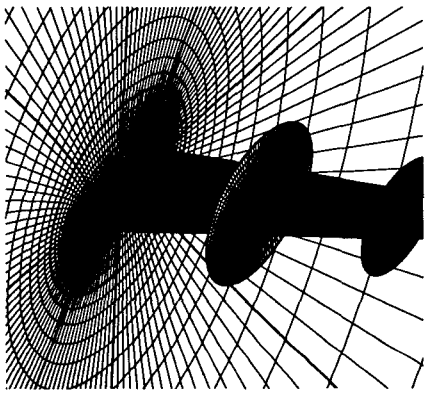

그림 1: 64개의 영역으로 분할된 ONERA M6 날개 형상: C1

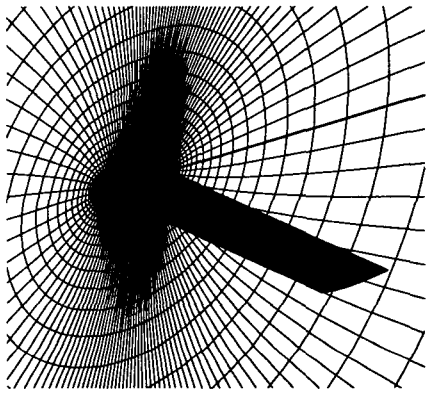

그림 2: 단일 영역의 유한 날개 형상: C2

3.2 해석 결과 검증

해석 결과의 타당성을 검증하기 위하여 클러스터와 Cray T3E에서 64개 CPU를 사용하여 ONERA M6 날개를 해석하였으며, 그림 3에 등압력 선도를 나타내었다.

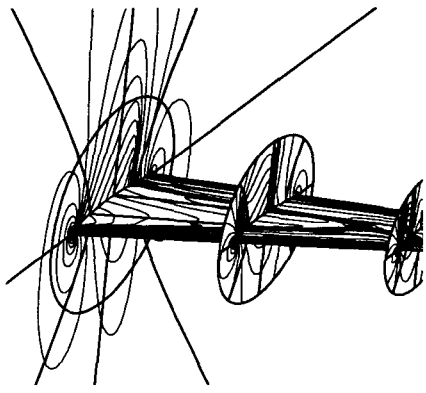

그림 3: ONERA M6 날개상의 등압력 선도(C1): $M_\infty = 0.84, \alpha = 3.06^o$

4. 세부 항목에 따른 성능 분석

4.1 해석 수행 시간 비교

그림 4, 5에서 해석에 소요된 시간을 나타내었다. 각 경우의 해석 결과는 해석자와 자료교환에 소요된 시간만을 포함한 결과이다.

4.1.1 Onera M6 날개위의 천음속 유동장 해석(C1)

그림 4는 C1에 대한 해석 수행시간을 비교한 결과이다. 그림들에서 각 CPU 성능과 네트워크의 특성을 파악할 수 있다. 그림 4.1의 466MHz의 알파 CPU와 Fast Ethernet을 사용하였을 경우, C4의 자료 교환 방법을 사용하지 않으면 병렬프로그램이 바른 성능을 보이지 못하는 결과를 나타낸다. 반면, 모든 경우에서 Cray T3E는 탁월한 성능을 나타내었다. 이런 Cray T3E의 성능에 비교되는 결과가 SCI 또는 Myrinet의 네트워크 카드를 사용한 경우이다. 각각에 대해 C3의 방법을 사용하였을 경우, Cray T3E의 경우와 유사한 기울기를 가지며 성능이 저하됨을 볼 수 있으며, CPU 성능이 우수하고 적절한 자료교환 방법에 대해서는 매우 좋은 결과를 보이고 있으며, 64개 CPU 이상에서도 성능이 좋게 나타날 것임을 예측할 수 있다. SCI와 Myrinet의 특성을 비교한 결과는 그림 4.3과 4.4에서 볼 수 있다. C3의 방법을 사용할 경우, SCI보다 Myrinet을 사용한 경우가 보다 좋은 성능을 보임을 알 수 있다.

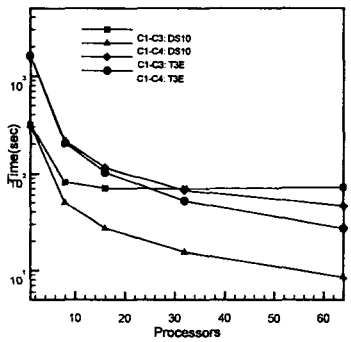

그림 4.1: DS10

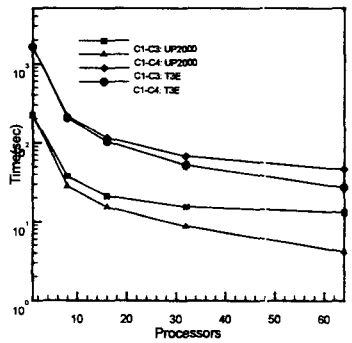

그림 4.2: UP2000

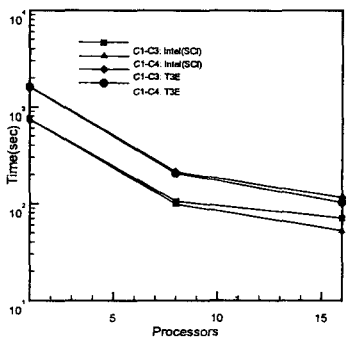

그림 4.3: Intel III(SCI)

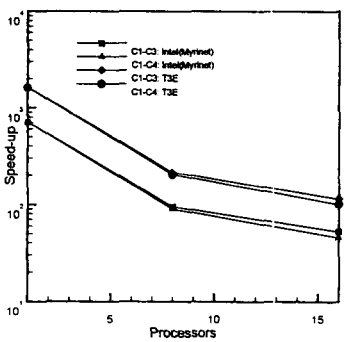

그림 4.4: Intel III(Myrinet)

그림 4: 해석 수행 시간 비교(C1)

4.1.2 3차원 유한 날개 위의 천음속 유동장 해석(C2)

그림 5는 C2에 대한 해석 수행시간을 비교한 결과이다. 그림들에서 각 CPU 성능과 네트워크의 특성을 파악할 수 있다. 이 경우, CPU수가 늘어남에 비례하여 격자수가 증가하게 된다. 따라서 1회 자료교환의 양은 매회 비슷한 양을 유지하는 반면, 자료교환의 횟수는 비례적으로 증가하게 된다. 따라서 현재의 문제는 네트워크의 Bandwidth보다 Latency가 중요하게 된다. 그림 5.1의 466MHz의 알파 CPU와 Fast Ethernet을 사용하였을 경우, C3의 자료교환방법을 사용하면 CPU수가 증가하면서 발생하는 자료교환 수가 비례하여 증가하므로 네트워크 성능의 한계를 분명히 보여주게 된다. 반면, C4의 방법을 사용할 경우 우수한 성능을 보임을 볼 수 있다. 위의 결과들로부터 Fast Ethernet을 사용할 경우, 여러 가지 문제들에서 Latency를 줄이는 방향의 프로그램을 하는 것이 매우 필요하게 된다. 그림 5.2와 5.4의 Myrinet 네트워크를 사용할 경우, C3의 방법을 사용한 경우 그림 5.1과 같이 좋지 못한 성능을 보이나 다소 그 경향성이 완만한 결과를 보이고 있다. 또한 32개 CPU까지는 자료교환에 소요되는 시간은 무시할 만큼 적다. 그림 4.3과 4.4와 같이 그림 5.3과 5.4에서 Myrinet을 사용한 결과가 SCI를 사용한 결과에 비해 Latency가 우수하며 보다 효율적임을 볼 수 있다.

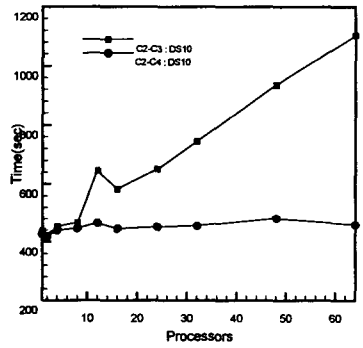

그림 5.1: DS10

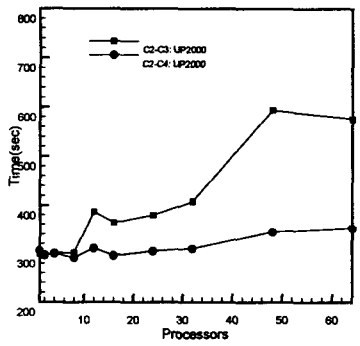

그림 5.2: UP2000

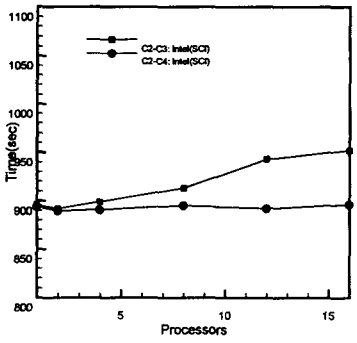

그림 5.3: Intel III(SCI)

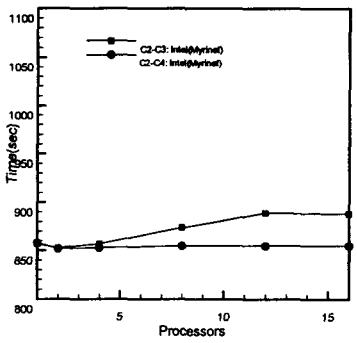

그림 5.4: Intel III(Myrinet)

그림 5: 해석 수행 시간 비교(C2)

4.2 병렬 성능 비교

4.2.1 Onera M6 날개위의 천음속 유동장 해석(C1)

그림 6은 C1의 해석 수행시간에 대한 병렬 성능을 나타낸 결과이다. 그림 4.1에서 예상할 수 있듯이 Fast Ethernet을 사용한 결과의 그림 6.1은 다양한 결과를 보여주고 있다. 그림 6.1에서 C3의 방법을 사용한 경우 speed-up은 8개 이상의 CPU에서 전혀 이루어지지 않고 있으며, CPU수가 증가하면서 상대적으로 계산시간보다 자료교환 시간이 증가하게 되어 Cray T3E와의 성능차이가 더욱 크게 나타남을 볼 수 있다. 반면, SCI나 Myrinet을 사용할 경우에 Cray T3E와 유사한 성능곡선을 나타내고 있다. SCI와 Myrinet의 성능을 나타낸 그림 6.3과 6.4에서 Myrinet을 사용할 경우 보다 우수한 speed-up을 나타내며 C3의 방법을 사용한 경우에도 성능저하가 미미하게 나타나며 16개의 CPU까지 좋은 결과를 보임을 알 수 있다.

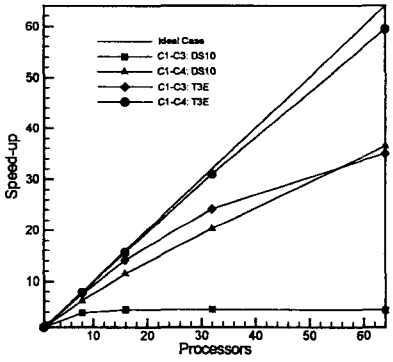

그림 6.1: DS10

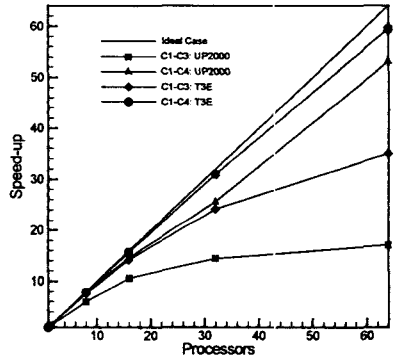

그림 6.2: UP2000

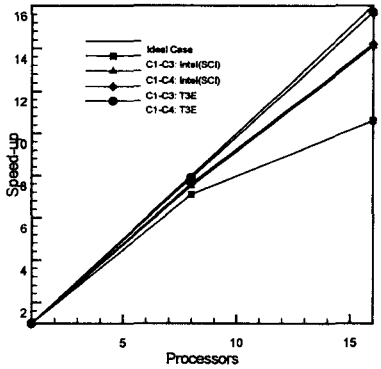

그림 6.3: Intel III(SCI)

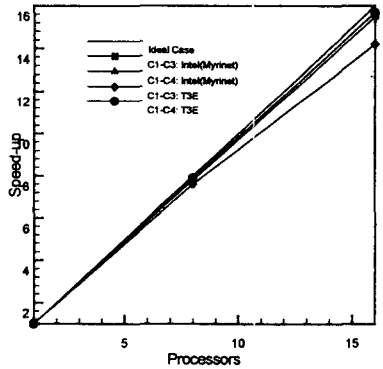

그림 6.4: Intel III(Myrinet)

그림 6: 병렬 성능 비교(C1)

4.2.1 3차원 유한 날개 위의 천음속 유동장 해석(C2)

그림 7은 C2의 해석 수행시간에 대한 병렬 성능을 나타낸 결과이다. 각 그림들에서 C4의 방법을 사용한 경우 매우 좋은 성능곡선을 나타내고 있다. 그림 7.1과 7.2를 비교해보면 그림 7.2의 경

우(알파 667MHz, Myrinet 사용)가 그림 7.1(알파 466MHz, Fast Ethernet)의 경우보다 C3의 방법을 사용할 경우 보다 좋은 성능을 나타냄을 볼 수 있다. 그림 7.3(Intel 667MHz SCI)과 그림 7.4(Intel 667MHz Myrinet)을 비교해 보면 앞의 결과들과 동일하게 Myrinet이 SCI 보다 우수한 성능을 보이고 있다.

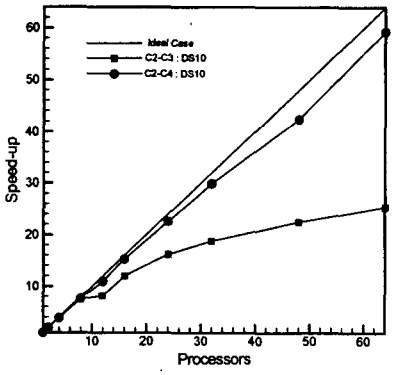

그림 7.1: DS10

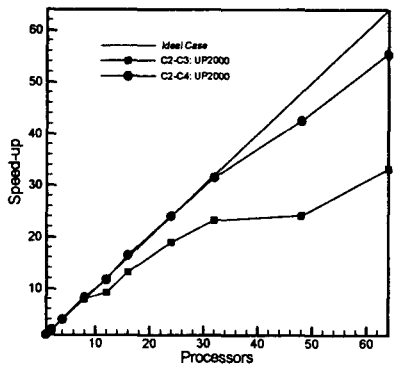

그림 7.2: UP2000

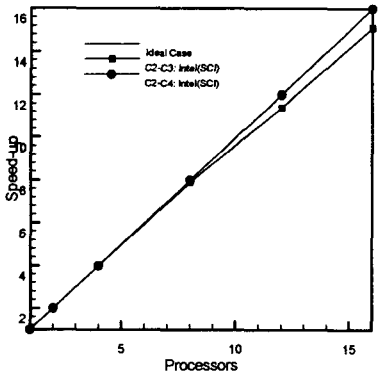

그림 7.3: Intel III(SCI)

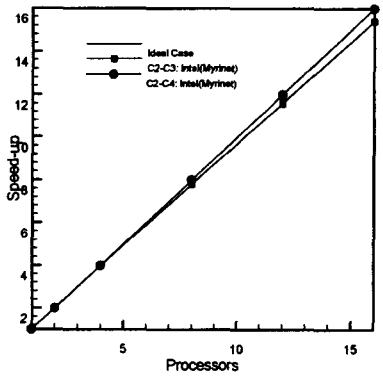

그림 7.4: Intel III(Myrinet)

그림 7: 병렬 성능 비교(C2)

5. 결 론

본 연구는 한국과학기술정보연구원(KISTI) 슈퍼컴퓨팅센터에서 수행한 TeraCluster 프로젝트의 일환으로 수행되었다. 본 연구에서는 DS10, UP2000, Intel III(SCI), Intel III(Myrinet)을 사용하여 구축된 시스템에 대해 CFD 코드를 적용하여 성능을 분석하였다. 본 논문에서는 이들을 종합하여 비교하고자 하였다. 정렬격자계를 사용한 본 CFD 방법은 위의 4개 시스템에 대해 매우 우수한 성능을 보였다. 특히, 큰 문제에서도 Fast Ethernet을 사용하여도 적절한 프로그래밍 기법만을 사용한다면 매우 우수한 성능을 보일 수 있다. 이는 실험실 단위에서 클러스터를 구축할 경우 시스템 확장에 대한 가이드라인을 제공할 것이라 생각된다. 그러나, 향후 Tera Flops급 시스템을 구성하고 Cray T3E를 대체하기 위해서는 보다 우수한 네트워크 장비가 필수적이다. 이에 대해서 본 논문의 분석결과에 따르면 현재 Myrinet이 Cray T3E의 3D Torus 구조에 근접하거나 우수한 성능을 보이고 있으며 대안이 될 것이라 판단된다.

6. 후기

본 연구는 컴팩코리아(주), 리눅스원(주), 삼성전자(주), 삼성물산(주), 자이온(주), Dophin(주) 업체의 도움을 통하여 이루어질 수 있었으며, 이에 감사의 뜻을 전합니다.

7. 참고문헌

[1] Pallas MPI Benchmarks - PMB, Part MPI-1.

[2] http://www.nas.nasa.gov/Research/Software/swdescription.html

[3] 이상산외 7인, "Cray T3E 대안 및 응용분야 최적 테라클러스터 개발(1차년도)", 한국과학기술정보연구원 (KISTI) 슈퍼컴퓨팅센터 내부보고서, TR00-0410-001

[4] 조금원, 이상산, "TeraCluster에서 CFD 코드의 병렬성능 분석," 전산유체공학회 추계발표회, pp.92-100, 서울대학교 2000

[5] M. Snir, S.Otto, S.Huss-Lederman, D.Walker and J. Dongarra, "MPI: The Complete Reference", The MIT Press, 1996.

[6] 조금원, 권장혁, 이승수, "비정상 Euler 방정식을 이용한 Chimera 기법의 병렬처리에 관한 연구", 한국전산유체공학회지, 제4권 제 3호, 1999년 12월, pp. 52-62.

[7] http://amber.aae.uiuc.edu/~m-selig/ads/coord_database.html