Beowulf 클러스터를 이용한 격자 양자 색소 역학 계산

세종대학교 김세용

1. 격자 양자 색소 역학(Lattice Quantum Chromo Dynamics)

자연계에는 중력, 전자기력, 약한 핵력, 그리고 강한 핵력 등 네 가지 기본적인 상호작용이 있다. 격자 양자 색소 역학은 이 네 가지 힘 중의 하나인 강한 상호 작용을 이해하고자 하는 입자 물리 및 핵 물리 분야의 이론이다. 강한 상호 작용이 발현되는 가장 일상적인 현상은 양성자와 중성자들이 결합하여 원자핵을 구성하는 것이다. 여기서 양성자와 중성자는 좀 더 기본적인 입자들인 쿼크(quark)와 글루온(gluon)이 속박되어 형성된 복합 입자로써, 핵 내의 양성자와 중성자 사이의 결합력도 이 쿼크와 글루온 사이의 상호 작용 결과이다. 이 강한 상호 작용은 전자기 현상을 묘사하는 양자 전기 역학(Quantum Electro Dynamics : QED)과 비슷한 게이지 상호 작용으로 밝혀졌으며 이에 관련된 게이지 이론을 양자 색소 역학(Quantum Chromo Dynamics : QCD)이라고 부른다 격자 양자 색소 역학은 양자 색소 역학의 정량적 이해를 목적으로 이 이론을 격자화한 것이다. 다음은 "물리학과 첨단기술"에 소개된 글[1]을 발췌 보충한 것이다.

1.1 물리적인 측면

강한 상호 작용은 asymptotic freedom이라고 하는 성질[2] 때문에, 다른 상호 작용과 달리 짧은 거리에서는 힘이 약해서 섭동적이고 먼 거리에서는 힘이 강해져서 비섭동적인 특이함을 지닌다. 강한 상호 작용은 앞에서 이야기한 것과 같이 전하를 띤 입자와 이들 사이의 게이지 힘으로 이해되는 전자기 상호 작용처럼, 색소 전하(color charge)를 띤 기본적인 자유도인 쿼크와 글루온이 존재하여 이들 사이에 게이지 원칙을 따르는 상호 작용을 하는 것으로 묘사된다. 그러나 이 기본적인 입자인 쿼크와 글루온은 보통의 환경에서 먼 거리를 자유롭게 움직이지 못하기 때문에 속박된 상태인 color singlet의 중간자(meson)나 바리온(baryon) 형태로 나타난다. 중간자와 바리온을 통칭하여 하드론(hadron)으로 부르며, 하드론의 대표적인 예가 양성자와 중성자 즉 핵자(자연의 물질을 구성하고 있는 여러 원자들의 핵을 이루고 있는)이다. 이러한 핵자들은 u(up)-쿼크, d(down)-쿼크, 그리고 글루온으로 형성되어서 자연계의 기본적인 입자가 아니다. 따라서 핵과 핵자에 관한 많은 것을 이해한다고 말하기 위해서는 기본적인 입자와 그들 사이의 상호 작용으로부터 핵과 핵자에 관한 것을 정량적으로 설명할 수 있어야 한다. 그런데 쿼크와 글루온이 양성자나 중성자를 이루는 과정은 비섭동적이고 비선형적이라서, 강한 상호 작용의 성질을 상대론적 양자장론 원칙에 근거하여 정량적으로 연구하려면 기존의 섭동적인 방법과 다른 방법이 필요하다. 이런 점에서 QCD에 격자 게이지 이론(lattice gauge theory)을 도입하게 된다. 격자 이론은 양자장론의 발산을 제어(regulate)하기 위해 사용되는 여러 가지 제이 방법 중의 하나로서 시공간을 연속체가 아닌 격자 구조로 생각하는 것이다. 물론 이러한 가상적인 격자 구조는 응집물리에서 많이 나타나는 결정(crystal)과 같이 실재하는 구조가 아니다. 모든 물리량은 이 가상 격자 구조와 관계가 없어서 격자 구조는 단

지 계산의 중간 과정으로 나타날 뿐인데, 이 격자 이론이 비섭동적인 현상을 포함하면서 양자장론을 묘사하는 거의 유일한 방법으로 알려져 있다. 노벨 물리학상을 수상한 K. G. Wilson이 이러한 격자 방법을 양자 색소 역학에 처음으로 적용했는데 이것이 바로 격자 양자 색소 역학(Lattice Quantum Chromo Dynamics; Lattice QCD)이다[3]. 이 격자 이론의 장점은 강한 상호작용에 관한 다른 모델 이론과 달리 체계적인 근사(systematic approximation)를 할 수 있다는 것이다 근사가 격자 상수에 의해 이루어지기 때문에 이 격자 상수만 연속체 극한을 취하면 원래의 양자 색소 역학의 모습을 되찾게 된다. 이 극한을 취하는 과정은 양자장론의 재규격화이론에 의해서 잘 이해되어 있다. 구체적으로 격자 게이지 이론을 사용하여 어떤 물리량을 계산하기 위해서는 경로 적분 방식(path integral method)으로 공식화된 양자장론을 사용하는 것이 편리하다. 즉 연결된(connected) Green의 함수는 partition function을 source로 미분하면 얻을 수 있다는 성질을 사용하여, Minkowski 시공간 상의 양자장론에서 임의의 물리량이

$$\langle O\rangle=\frac{\int[d\Psi][d\overline{\Psi}][dA_\mu^a]e^{iS_M}O(\Psi,\overline{\Psi},A_\mu^a)}{\int[d\Psi][d\overline{\Psi}][dA_\mu^a]e^{iS_M}}$$

로 표현된다는 사실을 이용한다[4]. 여기서 Planck 상수 $\hbar$를 1로 표시하는 단위를 사용했으며, $\Psi(\vec{x},t)$는 쿼크 장, $\overline{\Psi}(\vec{x},t)$는 반 쿼크 장, $A_\mu^a(\vec{x},t)$는 글루온 장이다. 그리고 $[d\Psi]$, $[d\overline{\Psi}]$, $[dA_\mu^a]$ 는 적분과 관련된 적분 measure이다. S_M는 QCD action인데 QCD 라그랑지안(Lagrangian)과, $S_M=\int dx^3dtL_{QCD}$의 관계가 있다. L_{QCD}는

$$L_{QCD}=\sum_f\overline{\Psi}_f(iD_\mu\gamma_\mu-m_f)\Psi_f-\frac{1}{4}F^a_{\mu\nu}F^{\mu\nu a}$$

$$\equiv\sum_f\overline{\Psi}_fM_f\Psi_f-\frac{1}{4}F^a_{\mu\nu}F^{\mu\nu a}$$ 로 주어진다. 여기서 f는 쿼크의 종류 (u, d, s, c, b, t)이고, m_f는 각 쿼크의 질량이고, γ_μ는 시공간의 Lorentz 구조와 관련된 Dirac matrix이고,

$$D_\mu=\partial_\mu-igA_\mu^aT^a,(a=1.2,\ldots,8)$$

$$F^a_{\mu\nu}=(\partial_\mu A^a_\nu-\partial_\nu A^a_\mu-igf^{abc}A^b_\mu A^c_\nu)T^a,$$

$(\mu,\nu=1,2,3,4)$이다. g는 결합 상수이고, T^a는 $SU(3)$ 게이지 그룹과 관련된 Gell-Mann matrix이다. 따라서 경로 적분 방식으로 정의된 양자장론은 수학적으로 보면 시공간 연속체 상에서 정의된(uncountably 무한 차원에서의) 적분 문제로 생각할 수 있다. 이렇게 정의된 양을 $t\to it$로 표현되는 Euclidean 시공간 상에서 보면 수치적으로 다루기 힘든 진동 위상(oscillating phase) 형태보다, 크기에 따라 감소되는 형태로 경로 적분이 쓰여지게 된다. 즉 위 공식은

$$\langle O\rangle=\frac{\int[d\Psi][d\overline{\Psi}][dA_\mu^a]e^{-S_E}O(\Psi,\overline{\Psi},A_\mu^a)}{\int[d\Psi][d\overline{\Psi}][dA_\mu^a]e^{-S_E}}$$

가 된다(여기서 S_E는 S_M을 Euclidean 시공간에서 다시 표현한 Euclidean action이다). 통상적인 섭동 방법은 action과 양자장을 자유 해(free solution) 근처에서 섭동하여 이 공식을 계산하는 것을 의미한다 만일 문제가 비섭동적인 경우에는 이러한 방법이 사용될 수 없기 때문에 이 경로 적분식을 직접 계산해야 한다. 우리가 관심있는 다양한 물리량은 $O(\Psi,\overline{\Psi},A_\mu)$에 따라서 결정된다. 예를 들어서 시공간의 두 점에 상관하는 상관 함수,

$G(\vec{x}',t';\vec{x},t)=$

$\langle\overline{\Psi}(\vec{x}',t')\Gamma_\mu\Psi(\vec{x}',t')\overline{\Psi}(\vec{x},t)\Gamma^\mu\Psi(\vec{x},t)\rangle$ 또는

$\langle\Gamma\ \overline{\Psi}^a(\vec{x}',t')\overline{\Psi}^b(\vec{x}',t')\overline{\Psi}^c(\vec{x}',t')$

$\times\Gamma_{bdu}\Psi^l(\vec{x},t)\Psi^m(\vec{x},t)\Psi^n(\vec{x},t)\rangle$를 계산하면 중간자나 바리온의 질량과 붕괴 상수에 관계된 양을 계산할 수 있다. 하드론 물리 현상에 중요한 역할을 하는 하드론 행렬 요소(hadronic matrix element)는 시공간의 세 점에 관계하는 세 점 함수(three point function)를 계산하면 구할 수 있다 예를 들어, 세 점 함수를 사용하여 하드론 현상론에 중요한 역할을 하는 bag 상수를 알 수 있다. 또, 영점 함수(zero-point function)를 통하여 QCD의 상전이 현상을 연구할 수 있다

1.2 수치 해석 알고리듬 입장에서 생각해 본 격자 양자 색소 역학

양자장론의 발산을 제어하기 위해 시공간 격자를 도입하면, 위의 공식이 시공간의 셀 수 없는(uncountable) 모든 점의 개수만큼의 차원에서 정의되는 적분에서, 격자 점에서만 정의되는 셀 수 있는(countable) 개수만큼의 차원에서 정의되는 무한 차원의 적분이 된다. 여기서 쿼크 장과 반 쿼크 장은 그림과 같이 격자 점에서 정의되고 게이지 장인 글루온 장은 그 기하학적 의미 때문에 격자 점과 격자 점을 잇는 link로 정의된다.

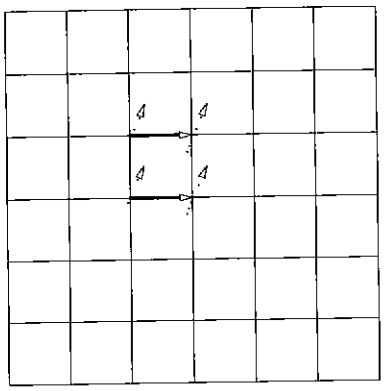

그림 1 격자 상의 쿼크 장(점선)과 글루온 장(실선)

시공간은 무한 공간이라서 물론 격자 점의 개수는 무한개이다. 따라서 적분은 여전히 무한 차원의 적분이다. 만일 격자 점으로 표시된 이 4차원의 무한 공간을 충분히 큰 상자 모양의 4차원 유한 공간으로 근사하면, 유한 개의 격자 점에서 정의되는 적분으로 표시된 문제가 된다. 이 모양은 유한한 자유도를 가지기 때문에 수치 해석이 가능하다. 여기서 주의할 점은 격자의 크기가 충분히 작아서 연속체 극한과 큰 차이가 없어야 되면서 상자의 크기가 충분히 커서 무한 공간과 물리적 차이가 작아야 한다는 점이다. 우리가 관심있는 현상이 하드론 크기 정도에서 생기는 것이므로 여기서 필요한 상자의 크기는 2.5 내지 4 fermi(1 fermi = 10^{-15} m) 정도이다. 연속체 극한과 큰 차이가 없는 격자 상수의 크기는 약 0.05 fermi 정도이거나 이보다 작아야 한다. 따라서 이러한 조건을 만족하는 격자 점의 개수는 1 차원 공간의 경우 약 50개 정도라서 만일 시간 차원의 격자점이 N_t인 경우, 전체적으로 약 $50^3 \times N_t$ 격자 점이 필요하고 적분 차원은 (격자점의 개수 × 격자 점 당 자유도)와 같으므로 $50^3 \times N_t \times$ 격자 점 당 자유도가 된다. 그리고 각 격자 점에서의 내적 자유도는 쿼크의 경우, Lorentz 자유도 × color = 4×3 = 12이고 link인 글루온의 경우, Lorentz 자유도×color = 4×8 = 32이다. 수치 적분 알고리듬은 여러 가지가 있지만 적분의 차원이 이렇게 큰 경우에 직접 적분하는 방법은 거의 불가능하기 때문에 몬테 카를로(Monte Carlo) 적분 방법이 많이 사용된다[5]. 쿼크 장과 반 쿼크 장은 반 가환인(anti-commuting) Grassmann 변수라서 물론 위 공식에 곧바로 몬테 카를로 적분 방법을 사용할 수는 없다. 우선 해석적으로 $[d\overline{\Psi}][d\Psi]$ 적분을 수행해야 하는데, 다행이 쿼크 장과 반 쿼크 장은 항상 $\overline{\Psi}Q\Psi$와 같은 bilinear 형태로 action에 나타나기 때문에 이 해석적 과정은 손쉽게 처리할 수 있어서

$$\langle O\rangle = \frac{\int [dA_\mu^a] e^{-\frac{1}{4}F^{\mu\nu}F^{\mu\nu}} Det(M) O(A_\mu, M^{-1})}{\int [dA_\mu^a] e^{-\frac{1}{4}F^{\mu\nu}F^{\mu\nu}} Det(M)}$$

로 주어진다. 이 식은 주어진 글루온 link에 대해서 $O(A_\mu^a)$를 계산하고 이 값에 가중치, $W(A_\mu^a)$,

$$W(A_\mu^a) = \frac{e^{-\frac{1}{4}F^{\mu\nu}F^{\mu\nu}} Det(M)}{\int [dA_\mu^a] e^{-\frac{1}{4}F^{\mu\nu}F^{\mu\nu}} Det(M)}$$

를 곱한 것을 구해서 모든 가능한 글루온 link에 대해서 더한 것으로 이해할 수 있다. 다시 말해서 물리량을 계산한다는 것은

$$\langle O\rangle = \int [dA_\mu^a] W(A_\mu^a) O(A_\mu^a)$$

로 이해할 수 있다. 보통의 몬테 카를로 적분의 경우에 A_μ^a을 무작위로 추출하여 위 식을 계산하지만, 양자장론과 같이 경우의 수가 많은 수치 문제에 있어서는 A_μ^a의 표본을 구할 때 $W(A_\mu^a)$의 확률을 가지고 표본 추출을 한다. 그러면 이러한 표본에 대하여

$$\langle O\rangle \sim \sum_{\{A_\mu^a \text{표본}\}} O(A_\mu^a)$$

의 방법으로 O를 계산하면 우리가 원하는 물리량

을 구할 수 있다. $W(A_\mu^a)$의 확률을 가지고 표본을 추출하는 방법은 양자장의 위상 공간에서의 random walk를 사용한다[6]. 이 과정에서 가장 시간이 많이 걸리는 부분은 M^{-1}과 같이 아주 거대한 행렬의 역행렬을 구하는 부분인데, 이 행렬 역시 차원이 너무 크기 때문에 직접 역행렬을 구하는 알고리듬보다 conjugate gradient와 같은 반복 (iterative) 알고리듬이 사용된다[5].

1.3 가벼운 하드론의 질량 계산을 예로 본 수치 계산의 양

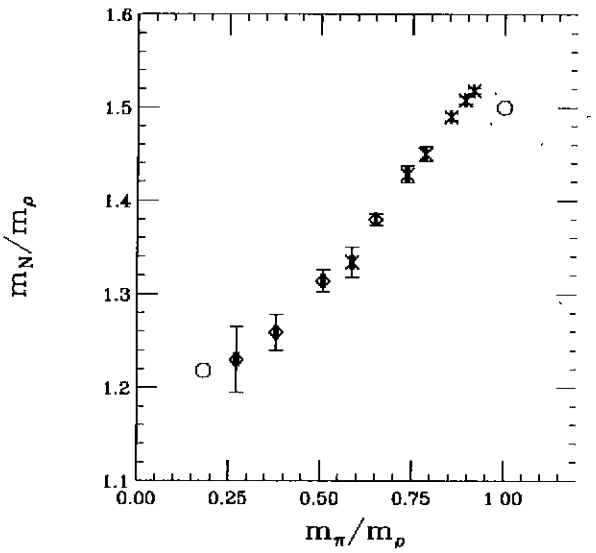

그림 2 가벼운 하드론 질량비

위 그림은 quenched 근사의 격자 양자 색소 역학을 이용하여 가벼운 하드론의 질량을 계산한 결과이다[7]. 가로축은 π 중간자 질량 대 ρ 중간자 질량, 세로 축은 핵자 질량 대 ρ 중간자 질량을 나타낸다. 왼쪽의 동그라미는 이 비의 실험값을 나타낸다. 오른쪽 동그라미는 가벼운 하드론에 관한 현상론에 많이 사용되는 비 상대론적 쿼크 모형의 값이다. 그림에 표시된 점들은 쿼크의 격자 질량을 변화시키면서 계산한 값들이다. 이 경우에 대하여 실제로 필요한 수치 연산 양에 대해서 생각해 보자. 이 결과를 얻기 위한 수치 연산은 크게 두 부분으로 나누어 생각할 수 있다. 하나는 몬테 카를로 적분에 관한 부분이고 또 다른 부분은 하드론의 질량을 얻기 위한 상관 함수의 계산 부분이다. 몬테 카를로 적분은 multi-hit Metropolis 알고리듬과 Over-relaxation 알고리듬이 사용되었는데, 이 알고리듬을 한 번 수행하는데 필요한 실수 연산의 개수는 격자 부피 당 약 40,000번이다. 몬테 카를로 적분 방법은 data sample사이에 auto-correlation이 있기 때문에 통계적으로 독립된 data (글루온 장)를 얻기 위하여 data 사이에 이 알고리듬을 각각 교대로 1,000번씩 사용하였다. 따라서 몬테 카를로 방법에서는 격자 부피 당 모두 4×10^7번의 실수 연산이 수행되었다. 상관 함수를 계산하는 부분은 conjugate gradient 알고리듬이 사용되었는데, 이 알고리듬을 한 번 수행하는데 필요한 실수 연산의 개수는 격자 부피 당 약 600이다. 위 그림의 값을 얻기 위하여, 쿼크 격자 질량을 격자 단위로 0.05, 0.04, 0.03, 0.02, 0.015, 0.01, 0.0075, 0.005, 0.0025, 0.00125로 선택하였다. 각각의 경우 conjugate gradient는 평균적으로 700, 900, 1100, 1600, 2200, 3300, 4400, 6600, 13200, 26400번 수행되었다. 결과적으로 conjugate gradient에서 각 data에 격자 부피 당 약 3.6×10^7 번의 실수 연산이 수행되었다. 이러한 숫자의 실수 연산을 250개의 data sample에 대하여 수행하였기 때문에 전체 실수 연산의 숫자는 약 1.9×10^{10}이다. 위 그림 결과는 $48^3\times64$ 격자 위에서 수행한 결과인데, 이 경우 필요한 연산의 숫자는 $1.9\times10^{10}\times$ 격자점의개수 $=1.3\times10^{17}$이다. 만일 컴퓨터의 연산 속도가 20 GFLOPS이면 이 연산은 약 18,700 시간이 걸린다. 통상적인 PC(200MHz Intel Pentium Pro MMX)의 경우, 그 연산 속도는 약 20 MFLOPS라서 총 연산은 18,700,000 시간이 걸리게 될 것이다. 그리고 이 격자 부피 상에 정의된 글루온 장의 정보 양은 2GBytes이고, 쿼크의 상관 함수 정보 양은 쿼크 격자 질량마다 약 500MBytes라서 250개의 data는 약 1.7TeraBytes의 정보를 의미한다. 이 예에서 볼 수 있듯이 격자 양자 색소 역학 문제는 수치적으로 large memory 문제라기 보다는computation-intensive 문제이다. 따라서 실제 프로그래밍에 있어서 가장 중요한 고려 대상은 코드의 속도이다(필요에 따라서 코드의 많은 부분을 assembly language로 작성하기도 한다). 병렬 프로그래밍 입장에서 격자 양자

색소 역학문제는 프로그램의 수행 도중에 통신의 패턴이 바뀌지 않는 fixed-grid 문제이다.

2. Alpha 클러스터 아키텍쥬어

위에서 살펴 본 바와 같이 격자 양자 색소 역학 계산은 수치 연산의 양이 많은 문제이다. 따라서, 범용 PC와 범용 네트워크 장비를 가지고 값싸지만 고성능인 슈퍼컴퓨터를 직접 제작하여 계산에 사용하는 것은 상당히 매력적인 생각이다. 실제로 많은 이론 물리 그룹들이 수 백 GigaFlops범위의 슈퍼컴퓨터를 만들어 격자 양자 색소 역학 계산에 사용하고 있다[1,8]. 현재 구성된 세종대학교 물리학과의 Alpha 클러스터는 이 그룹들의 노력보다는 훨씬 작은 규모의 프로젝트로, off-the-shelf 하드웨어와 소프트웨어만 가지고 제작되어 작은 노력으로 쉽게 유지 보수할 수 있다는 장점이 있다[9]. 물론 다른 그룹의 예에서 볼 수 있듯이 쉽게 scale-up할 수도 있다[10].

2.1 하드웨어

세종대학교 물리학과 Alpha 클러스터의 하드웨어는 크게 노드와 네트워크 두 부분으로 나누어 생각할 수 있다. 노드 컴퓨터는 특별한 장비의 교체 없이 모니터, 마우스, 키보드를 연결하면 각각이 독립적인 PC로 언제든지 사용될 수 있는 범용 PC이다. 다음은 노드 PC의 사양이다.

600 MHz 21164 Alpha CPU
164 LX main board (2MBytes L3 cache)
256 MBytes SDRAM main memory
SCSI Hard disk (2 또는 4GBytes)
CD-ROM drive
3 1/2" Floppy drive
3COM 3C905 100 Mbps Ethernet card

각 노드 PC는 300watt의 전력을 필요로 한다. Alpha CPU를 장착한 PC를 노드 컴퓨터로 선택한 이유는 클러스터의 구성 목적이 high available system이나 large data transaction이 아니라 high performance computing에 있기 때문이다. 클러스터 전체의 이론 성능과 가격을 생각하면 값 싼 Intel CPU를 장착한 PC를 더 많이 사용하여 클러스터를 구성하는 것이 유리할 것 같으나, 클러스터의 구조상 연산 속도를 높이기 위해서 통신을 적게 하는 것이 유리하기 때문에 단위 노드의 연산 속도가 빠른 것이 훨씬 더 효과적이다. 현재 사용하고 있는 네트워크 장비는 통상적인 LAN 장비에서 많이 볼 수 있는 Bus 방식의 HUB가 아니고, 다른 노드의 영향을 받지 않고 임의의 두 노드 사이의 통신을 지원할 수 있는 100Mbps switched Ethernet 방식이다. 다음은 Intel Express 510T switch의 사양이다.

24 port
8000 MAC per port

이 switch 덕분에 임의의 두 노드 사이의 통신 거리가 같아서, 사용 가능한 24개의 switch 포트의 숫자보다 작은 숫자인 8개의 노드로 구성된 세종대 클러스터에서는 내부 통신 topology에 관하여 특별히 다른 고려가 필요 없다. 단지 노드 0은 외부 세계와의 연결을 위하여 Fast Ethernet card(Intel 100Mbps EtherPro)를 한 개 더 장착하였다. 따라서 클러스터의 통신 구조는 다음 그림과 같다

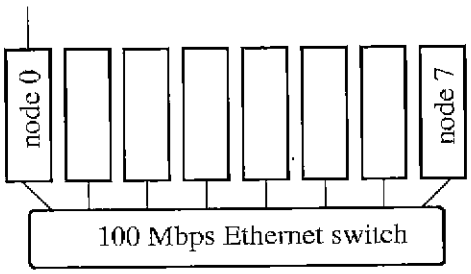

그림 3 클러스터 노드의 논리적 배열

각각의 노드에 GNU/Linux에서 local subnet 주소로 따로 지정되어 있는 주소 192.168.x.x 중에서, 192.168.1.1에서 192.168.1.8까지의 IP 주소를 각각 할당하였다. 노드 0은 이 local subnet 주소뿐 아니라, 클러스터 외부와의 통신을 위하여 DNS에 등록된 IP를 가지고 있다. 이러한 논리적인 구조는 할당 가능한 IP 주소에 대한 고민 없이 클러스터의 크기를 확장할 수 있다는 장점이 있다. 외부에서 볼 때는 단 한 곳의 IP 접속점 만을 제공하기 때문에 여러 사용자가

있을 경우 외부로의 네트워크에 부담이 생길 가능성이 있지만, 클러스터를 소수의 이용자가 독점적으로 사용하는 세종대학교 클러스터의 경우 별다른 문제점이 없다. 그리고, 클러스터의 유지, 보수, 업그레이드를 용이하게 하기 위하여 각 PC의 시리얼 디바이스(모니터, 마우스, 키보드)는 시리얼 디바이스 공유기(Master View Plus CS-138D)를 통하여 한 개의 모니터, 마우스, 키보드에 연결되어 있다.

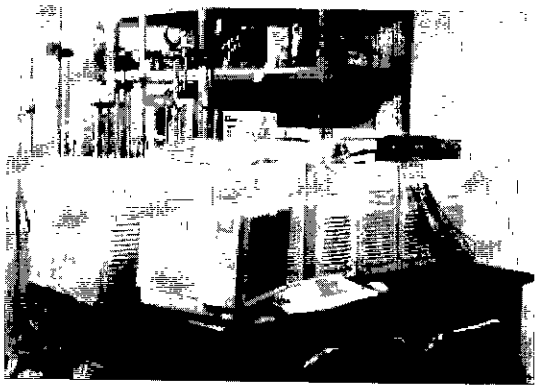

그림 4 세종대 클러스터

2.2 소프트웨어

소프트웨어 측면은 크게 노드 운영체계와 병렬 프로그래밍 환경으로 나누어 생각할 수 있다. 노드 운영 체계는 Alpha processor를 위한 알짜 리눅스[11] 버전 5.2a이고 kernel 버전은 2.2.1이다. 병렬 프로그래밍 환경은 기본적으로 message passing 형태로, PVM(Parallel Virtual Machine)[12] version 3.4, LAM (Local Area Multi-computer)[13] version 6.2, MPICH (Message Passing Interface- Chameleon)[14] version 1.2.2, 그리고 KMPI (Kool Message Passing Interface)[15] version 0.5 등을 설치하고 시험하였다. 실제 production run은 LAM 환경을 이용하고 있다. 이 모든 message passing library는 TCP/IP를 통신의 기본적인 mechanism으로 채택하고 있다. 알짜 리눅스 배포판은 GNU compiler system (egcs version 1.0.3)을 포함하고 있어서, FORTRAN, C, C++ 프로그램을 GNU compiler를 이용하여 compile 할 수 있다. 세종대 클러스터에는 GNU compiler뿐만 아니라, Compaq 사가 제공하는 Compaq FORTRAN/C compiler version 1.0.2를 "Technology Enthusiast" 라이선스 조건[16] 하에서 무료로 사용하고 있다. 그리고 위에서 이야기한 병렬환경은 FORTRAN과 C 언어에서 병렬 라이브러리를 사용할 수 있도록 각 언어의 compiler에 대한 wrapper를 제공한다. egcs compiler에 비해서 Compaq Fortran compiler의 성능이 2배 내지 3배 뛰어나기 때문에 실제 production run에서는 Compaq Fortan compiler를 사용한다. 병렬 프로그램 수행은 이러한 병렬환경이 제공하는 "mpirun"은 rsh을 사용하도록 되어 있기 때문에, 각각의 노드가 rsh을 사용할 수 있어야 한다 (ssh을 사용할 수도 있다). 세종대 클러스터는 NIS service를 사용하여 각각의 노드가 password를 공유한다. 그리고 각각의 노드가 SCSI Hard disk를 가지고 있기 때문에 이를 효율적으로 사용하기 위하여 file system을 크게 네 개의 partition으로 구성하여 사용한다.

1) 운영체계가 있는 / local system partition
2) 각 사용자의 home directory가 있는 NFS-mounted /home directory
3) 사용자가 대규모의 I/O를 할 수 있도록 마련된 /scratch directory
4) 일반적인 backup를 위한 NFS directory

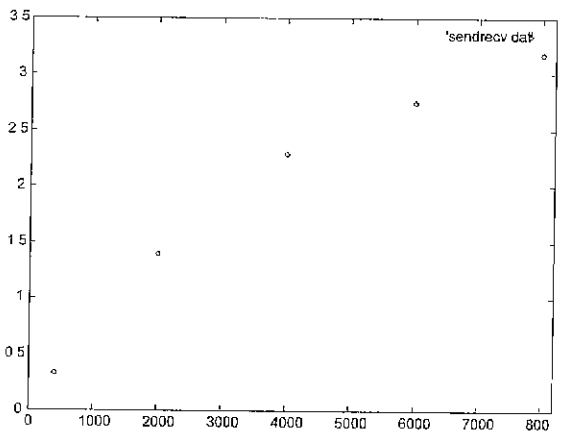

그림 5 네트워크 성능 시험 결과

1)과 3)은 각각의 노드 Hard disk에 위치하며, 2)는 물리적으로는 노드 0의 /home directory이고 다른 노드는 mount point만을 제공한다. 4)는 물리적으로 노드 0, 1, 2, 3에 위치하며, 다른 노드는 역시 mount point만을 제공한다. 사진은 삼성 전자의 도움으로 제작, 운영되고 있는 세종대 물리학과의 8 노드 클러스터이다. 이 클러스터의 이론적 최대 성능은 9.6 (1.2GFLOPS/노드 × 8 노드)GFLOPS이다.

그리고 위 그림은 네트워크 bandwidth를 MPI의 blocked send-receive를 사용한 round robin test 시험 결과이다. 가로축은 message byte의 크기이고 세로 축은 MBytes/sec이다. 네트워크 장비의 이론적인 최고 성능이 100Mbps (약 12MBytes/sec)이므로 작은 message에 대해서 클러스터가 약 25% 정도의 실제 성능을 보임을 알 수 있다. 다음 표는 세종대 물리학과의 클러스터를 구성하는데 소요된 비용을 부품별로 나열한 것이다.

표 1 클러스터 구성 가격표

부 품	가 격	수 량	총 액
PC	N/A	8	N/A
memory	178,461	18	3,212,300
switch	2,530,000	1	2,530,000
LAN Card	82,160	1	82,160
UPS	1,870,000	1	1,870,000
Console	649,000	1	649,000
기타(cable 등)	49,000	N/A	49,000
총 액			8,392,460

* used PC를 삼성 전자가 증여하였기 때문에 정확한 가격을 알 수 없음. PC의 메모리만 업그레이드하였다(메모리 가격 참조)

3. B_c 입자 성질의 격자 계산

현재 세종대 클러스터에서 진행되고 있는 프로젝트는 최근에 실험에 의해서 존재가 확인된 [17] B_c 중간자의 성질을 격자 계산을 통하여 연구하는 것이다. 이 중간자는 상대적으로 무거운 쿼크인 c(charm) 쿼크와 b(beauty) 쿼크가 결합되어 생긴 중간자($b\bar{c}$ 또는 $c\bar{b}$)인데, 생성되면 약한 핵력에 의해서만 붕괴가 가능해서 이 중간자의 붕괴 과정에 약한 핵력과 강한 핵력이 복잡하게 작용한다. 따라서, 이러한 측면을 연구하는데 있어서 B_c 중간자는 특별한 역할을 한다 [18]. 이 입자의 질량과 붕괴 상수 등을 연구하고자 하는 본 연구에서는 이전의 격자 연구[19]와 달리 가벼운 쿼크(u, d)와 가벼운 반 쿼크($\bar{u}$, $\bar{d}$)의 짝이 진공에서 Heisenberg 불확정성 원리를 따라서 생성 소멸되는 효과가 B_c 중간자의 비섭동적인 성질에 어떤 영향을 주는 지에 대하여 조사하고 있다. B_c 중간자 질량은 기본적인 물리량이고 붕괴 상수는 B_c 중간자의 붕괴 과정의 비섭동적인 현상을 정량적으로 나타내는 기본적인 물리량이다. 이 계산에 사용되는 격자의 부피는 $8^3\times32$이고 격자의 크기는 약 0.4fm이다(뒤에서 설명하는 유효 이론을 사용하기 때문에 격자의 크기가 0.05fm보다 커야 된다). 그리고 공간 방향으로 periodic 경계 조건이 부여하고 시간 방향으로 글루온 장에 대해서는 periodic 경계 조건을, 쿼크 장에 대해서는 anti-periodic 경계 조건을 부여하였다. 1.3 절의 예와 달리 진공 편극(vacuum polarization)의 효과를 고려하기 위해서, 몬테 카를로 적분을 위한 표본 추출 과정에 Hybrid Molecular Dynamics(HMD)라는 알고리듬을 사용한다[20]. HMD는 글루온 장의 표본 추출을 위상 공간에서의 운동으로 생각하여 이 과정을 5-차원 공간(4차원 시공간 + 가상의 시간 차원)의 Hamiltonian dynamics의 일종으로 생각하는 것이다. 이렇게 구한 글루온 장의 표본에 대하여 무거운 쿼크인 b 쿼크와 c 쿼크의 두 점 상관 함수(two-point correlation function 또는 쿼크 propagator)를 구한다. 그런데 무거운 쿼크의 두 점 상관 함수를 구하는데 있어서 어려운 점이 있다. 그것은 무거운 쿼크의 Compton 파장이, 적당한 시간 안에 몬테 카를로 simulation으로 구할 수 있는 글루온 장 표본의 격자 크기보다 작다는 것이다(이 문제를 피하기 위해서 격자 크기를 반으로 줄이면 수치 연산의 양은 2^4 즉 16배가 늘어난다). 이러한 점을 양자장론에서 해결하는 방법은 유효이론(effective

theory)을 사용하는 것이다. 물론 유효 이론과 원래 이론과의 차이에 대한 이해는 양자장론의 방법에 의해서 잘 정립되어 있기 때문에 유효 이론은 모델 이론이 아니다. 따라서 격자 계산은 여전히 체계적인 근사가 가능하고 양자장론의 원칙 하에서 행해지는 계산이다. 무거운 쿼크의 경우에 사용되는 유효 이론은 비상대론적 양자 색소 역학 (non-relativistic Quantum Chromo Dynamics)이다[21]. 이 유효 이론을 사용해서 구한 쿼크의 두 점 함수를 가지고 B_c 중간자의 두 점 함수를 구한다.

$$G_{B_c}(\vec{x}', t', \vec{x}, t) = \langle 0|\bar{b}(\vec{x}', t')c(\vec{x}', t')\bar{c}(\vec{x}, t)b(\vec{x}, t)|0\rangle$$
$$= G_c(\vec{x}', t'; \vec{x}, t)G_b(\vec{x}', t', \vec{x}, t)^{\dagger}$$

(여기서 †는 Hermitian conjugate이다). 이 두 점 함수의 zero-momentum projection을 취하고 두 점 사이의 시간 간격을 크게 하는 극한, $\lim_{|t'-t|\to\infty} G_{B_c}(t', t)$을 취하면 $\lim_{|t'-t|\to\infty} G_{B_c}(t', t) \sim f_{B_c}^2 e^{-E_{B_c}(t'-t)}$를 사용하여 우리가 원하는 B_c 중간자의 질량 ($m_c + m_b + E_{B_c}$)과 붕괴 상수(f_{B_c})를 구할 수 있다.

4. 프로그래밍 및 코드의 성능

위 문제를 컴퓨터로 계산하기 위한 구체적인 프로그래밍은 두 부분으로 나누어진다. 경로 적분을 위한 글루온 장의 표본 추출 부분과 이러한 글루온 장 표본을 이용하여 우리가 관심 있는 물리량을 비상대론적 유효 이론으로 계산하는 부분이다. 글루온 장 추출 부분은 계산 량이 가장 많기 때문에 클러스터의 8개 노드를 모두 병렬로 사용해야 한다. 따라서 MPI를 사용한 병렬 코드로 작성되어 있다. 물리량 계산 부분은 비 상대론적 유효이론을 사용하기 때문에 boundary value 문제가 아닌 initial value 문제이다. 그러므로 이 부분은 EP(Embarrassingly Parallel) 문제 형태로 생각할 수 있기 때문에, 이 부분의 코드는 Von Neumann식 시리얼 코드를 I/O 부분만 병렬화한 형태로 작성된다. 앞에서 이야기한 대로 격자 양자 색소 역학 문제는 fixed-grid 형태의 문제이기 때문에, 격자 점을 각각의 노드에 균등하게 배분하면 병렬 프로그램에 있어서 어려운 문제인 domain-decomposition과 load-balancing은 상대적으로 쉽게 해결할 수 있다.

4.1 몬테 카를로 방법에 의한 글루온 장 표본 추출

앞에서 이야기 한 대로 글루온 장 표본 추출은 5차원 공간($\vec{x}, t, \tau$)에서의 Hamiltonian dynamics를 사용하여 실행한다. 이 운동 방정식은 4차원에서의 점 입자의 운동 방정식인 $\frac{dq}{dt} = \frac{\partial H(p,q)}{\partial p}$, $\frac{dp}{dt} = -\frac{\partial H(p,q)}{\partial q}$과 비슷한 형태로 다음과 같다.

$$\frac{dU_\mu(\vec{x}, t, \tau)}{d\tau} = iH_\mu(\vec{x}, t, \tau)U_\mu(\vec{x}, t, \tau),$$
$$i\frac{dH_\mu(\vec{x}, t, \tau)}{d\tau} = U_\mu(\vec{x}, t, \tau)[\frac{2}{g^2}V_\mu(\vec{x}, t, \tau)$$
$$-\frac{1}{4}N_f\frac{\partial}{\partial U_\mu(\vec{x}, t, \tau)}tr\ln(M^\dagger M)]$$

여기서 $U_\mu(\vec{x}, t, \tau)$는 4차원 점 입자 운동 방정식에서의 좌표와 같은 역할을 하는 글루온 장이며, $H_\mu(\vec{x}, t, \tau)$는 4차원 점 입자 운동 방정식에서의 conjugate momentum과 같은 역할을 하는 글루온 장의 conjugate momentum 장이다. $V_\mu(\vec{x}, t, \tau)$은 글루온 장 link, $U_\mu(\vec{x}, t, \tau)$와 관련된 plaquette환경변수($\sum_{\mu\neq\nu} U_\nu(\vec{x}+\mu, t, \tau)\ U_\mu(\vec{x}+\nu, t, \tau)^\dagger\ U_\nu(\vec{x}, t, \tau)^\dagger$)이고, M은 1.1절에서 정의된 $U_\mu(\vec{x}, t, \tau)$를 포함한 행렬이고 tr은 color와 spin에 대한 trace를 의미한다. 따라서, 기본적인 알고리듬은 1) 글루온 장을 hard disk에서 읽어 들인다. 2) Gaussian 분포를 따르는 초기 운동량, $H_\mu(\vec{x}, t, \tau)$을 선택한다 3) 이 운동량을 이용하여 $U_\mu(\vec{x}, t, \tau)$을 계산한다. 4) 이 $U_\mu(\vec{x}, t, \tau)$을 이용하여 $H_\mu(\vec{x}, t, \tau)$을 계산한다. 5) 3)과 4) 과정을 되풀이한다. 6) 정해진 τ 간격이 되면 hard disk에 계산된 글루온 장을 저장하거나 2) 과정으로 되돌아간다. 이 알고리듬 과정에서 통신이 필요한 부분은 I/O(읽고 쓰는 과정을 local /scratch partition을 사용하는 경우)와 격자 점의 글루온 장과 운동량 계산과정에서 이웃하고 있는 격자 점에 정의된 글루온 장이나 쿼크 장의 정보가 필요한 경우이다. 클러스터가 message passing에

의한 병렬 처리 환경을 제공하고, Fast Ethernet은 이전의 10 base-T Ethernet에 비해서 상당한 통신 성능의 증가를 가져와서 이러한 병렬 처리 환경의 효율을 높인 것이 사실이다. 하지만 통신에 소요되는 시간이 순수한 system overhead이라는 것은 움직일 수 없는 것이기 때문에, 클러스터의 병렬 성능을 높이기 위해서 되도록 통신의 양을 줄이도록 프로그램을 작성해야 한다. 그래서 시간 축 방향으로 격자 점을 각 노드에 분배하였다. 즉 시공간 격자 점 좌표, $8^3\times(N_t=1,2,3,4)$는 노드 0에 분배하고, $8^3\times(N_t=5,6,7,8)$은 노드 1에 분배하는 식으로 domain decomposition을 행하였다. 이러한 배분은 예를 들어, 격자 점 $N_t=1$을 계산할 때 필요한 $N_t=32$의 격자 점의 정보를(경계 조건 때문에) 노드 0과 노드 7 사이의 통신으로 받게 한다. 이러한 선택이 가장 작은 surface-to-volume ratio를 주게된다. 만일 논리적인 일관성을 유지하기 위해서 격자 점을 모든 방향으로 균등히 분배하면 이 선택보다 더 큰 surface-to-volume ratio를 주게되고 코드의 성능을 감소시키게 된다. I/O는 각각의 노드 hard disk에 마련된 /scratch 영역을 사용한다. 물리량을 계산하기 위해서 $G_b(\vec{x},t), G_c(\vec{x},t)$을 계산하는 과정은 $(D_t-\frac{\vec{D}^2}{2m_q})G(\vec{x},t)=\delta(\vec{x})\delta(t)$를 푸는 것인데, 보통의 시리얼 코드의 작성과 크게 다르지 않아서 주의할 점은 단지 I/O 부분 만이다. 즉 각각의 노드가 제 각각 다른 글루온 장 표본을 읽어 들이고 계산 결과를 각각 다른 file에 쓰도록 하기만 하면 된다.

4.2 클러스터의 성능

클러스터의 성능을 측정하는 방법은 여러 가지가 있지만 그 중 어느 한 가지 방법의 수치가 클러스터의 성능을 모두 보여 준다고 생각할 수 없다. 따라서 많이 사용되는 NPB test suit나 scalability test 등과 같은 benchmark 숫자보다 클러스터의 performance computing 사용자 입장에서 가장 중요한 의미를 갖는 비교를 하는 것이 좋다고 생각된다. B_c 격자 문제를 계산하면서 클러스터를 사용하는 이유는 연산 속도의 증가이다. 예를 들어서 한 대의 컴퓨터를 사용했을 때 문제 해결에 걸리는 시간이 4000시간이라면, 8대의 컴퓨터를 병렬 연결하여 클러스터로 작업하였을 때 500시간 정도에서 해결할 수 있는 것이 중요하다. 비상대론적 격자 양자 색소 역학 계산은 EP 문제이므로 speed-up이 큰 어려움 없이 이상적인 speed-up을 따른다. 자세한 분석이 필요한 부분은 몬테 카를로 적분을 위한 표본 추출 과정이다. 한 대의 컴퓨터로 $8^3\times32$ 격자에 대해서 HMD 과정을 200번 수행하였을 때 걸린 시간이 약 5705초이다(프로젝트 전체로는 이러한 과정을 230000번 수행하였다). 8 노드 클러스터에 병렬로 수행하였을 때 걸린 시간이 약 900초이다. 비선형 문제의 몬테 카를로 과정 특성상 연산 조건이 조금만 바뀌어도 위상 공간에서의 움직임이 달라지기 때문에 5705초/900초 = 6.3이 speed-up이 아니다. 실제 총 FLOP 숫자를 세는 것이 필요하다. HMD 알고리듬의 주된 과정은 micro라고 불리는 과정인데, 이 과정은 다음과 같은 FLOP 숫자의 연산을 하는 과정을 부른다. 다음 표는 각 과정에 필요한 FLOP 숫자를 헤아려본 결과이다(여기서 N은 congrad 과정이 반복된 횟수이다). 따라서 200회 반복된 micro 과정의 경우에 총 FLOP 숫자는 $(24455+603\times N)\times200\times8^3\times32$이다. $8^3\times32$ 격자 부피를 한 대의 컴퓨터에 수행했을 때 N은 평균 272.4회이다. 따라서 총 FLOP 숫자는 6.18×10^{11}이다. FLOPS는 6.18×10^{11}/5705초 = 108 MFLOPS이다. $8^3\times32$ 격자 부피를 8 노드 클러스터에 수행했을 때 N은 평균 264.4회이다. 따라서 총 FLOP 숫자는 6.02×10^{11}이다. FLOPS는 6.03×10^{11}/900초 = 670 MFLOPS이다. 결과적으로 B_c 격자 계산의 몬테 카를로 표본 추출 과정의 코드는 8 노드 클러스터 상에서 speed-up이 6.2 정도이다. 따라서 통신에 의한 system overhead는 22% 보다 작다고 생각할 수 있다.

670MFLOPS 정도의 연산 속도는 Cray C90 1 CPU의 실제 성능보다 더 나은 속도이다. 하지만 PC의 가격이 삼 백 만원 정도 된다고 추정할 때 나머지 부품의 가격을 포함한 클러스터의 가격이 C90 1 CPU에 비해서 훨씬 싸다. 예상한 대로 가격 대 성능 비 면에서 클러스터가 우수하

다. 그리고 한 대의 컴퓨터를 사용했다면 2000시간 이상이 걸릴 일을 클러스터를 사용하여 300시간 정도에서 해결할 수 있다.

표 2 격자 부피 당 subroutine 별 실수 연산의 개수

subroutine name	fp. operation count/latuce volume
noise	387
gauge_update	10752
congrad	603*N
capsid_	288
sumuuu	9864
update	3164
총 계	24455 + 603*N

5. 결 론

전적으로 off-the-shelf-component인 하드웨어와 소프트웨어만을 가지고, 특정한 분야의 수치 해석 문제를 해결하기 위해서 그 분야에 맞게 사용자가 직접 클러스터를 구성하여 사용하는 것이 큰 어려움 없이 가능해졌다. 이러한 분야의 한 예로, Grand Challenge 문제의 하나인 격자 양자 색소 역학 분야의 경험을 소개했다. 여러 대의 PC를 병렬로 사용하여 슈퍼컴퓨터 수준의 성능을 발휘할 수 있는 Beowulf 클러스터는 가격 대 성능 비 면에서 슈퍼컴퓨터가 필요한 학과나 소단위 연구그룹들이 슈퍼컴퓨터를 사는 것보다 싼 가격으로 제작, 운영할 수 있다는 장점이 있다. 그러나 클러스터가 슈퍼컴퓨터 수준의 연산 속도를 내기 위해서 병렬 프로그래밍이 꼭 필요한데, message passing 형태의 병렬 프로그래밍은 경험과 노력이 필요하다는 점이 이러한 클러스터 아키텍츄어가 사용자들에게 널리 받아들여지는데 어려운 점으로 작용한다. 격자 양자 색소 역학은 이런 점에서 클러스터 아키텍츄어에도 적응할 수 있는 수치 해석 문제이다. fixed grid 형태의 변수 배열은 병렬 프로그래밍에 있어서 중요한 문제인 load-balancing과 domain decomposition을 상대적으로 쉽게 해결할 수 있게 한다. 그리고 고정된 데이터 형태는 통신 형태를 단순화시킨다. 몬테 카를로 형태의 알고리듬은 같은 프로그램을 변수만 바꾸어 여러 번 반복해서 사용한다는 점에서 병렬 프로그래밍에 소요되는 노력과 수고보다 프로그램의 연산 속도를 높여서 전체 과제의 수행 시간을 단축시켜서 얻는 효과가 더 크다. 다른 분야의 수치 해석 문제도 격자 양자 색소 역학의 경우와 비슷한 조건을 만족하면 PC 클러스터 아키텍츄어가 크게 도움이 될 수 있을 것이다.

참고문헌

[1] 김세용, High Performance Computing과 격자 양자 색소 역학, "물리학과 첨단 기술", Vol 8, No. 10, 19 (1999).

[2] D. Gross and F. Wilczek, Phys. Rev. Lett. 30, 1343 (1973); H. D Politzer, Phys. Rev Lett. 30, 1346 (1973).

[3] K. G. Wilson, Phys. Rev. D10, 2445(1974)

[4] R. P. Feynman and A. R. Hibbs, Quantum Mechanics and Path Integral, McGraw-Hill Book Co., New York.

[5] W. Press et. al, Numerical Recipe, Cambridge University Press, Cambridge.

[6] M. Creutz, Quarks, Gluons, and Lattices, Cambridge University Press, Cambridge.

[7] S. Kim and S. Ohta, to be published in Phys. Rev. D.

[8] J C. Sexton, Lattice 95, Nucl. Phys B (Proc. Suppl.) 47, 236 (1996); N H. Christ, Lattice 99, to be published in Nucl Phys. B (Proc. Suppl.).

[9] Beowulf Project, http://www.beowulf.org/.

[10] Avalon project, http://cnls.lanl.gov/avalon.

[11] 알짜 리눅스, http://linuxone.co.kr/alzza.html.

[12] PVM - Parallel Virtual Machine, http://www.epm.ornl.gov/pvm

[13] LAM/MPI Parallel Computing, http://www.mpi.nd.edu/lam.

[14] MPICH - A Portable MPI Implementation,

http://www.mcs.anl.gov/mpi/mpich/index.html.
[15] KMPI (Kool MPI) : http://hana.etri.re.kr/~mgkim/kmpi.
[16] Compaq Fortran Products, http://www.compaq.com/fortran.
[17] F. Abe et al., Phys. Rev Lett. 81, 2432(1998); F. Abe et al, Phys. Rev. D58, 112004(1998).
[18] M. Beneke, G. Buchalla, Phys. Rev. D53, 4991 (1996); I. I. Bigi, Phys. Lett. B371, 105(1996).
[19] S. Kim, hep-lat/9511010, C. T. H. Davies et al., Nucl. Phys. B (Proc. Suppl.) 47, 421(1995); C. T. H. Davies et al., Phys. Lett. b382, 131(1996); A. Ali Khan, T. Battacharya, Nucl. Phys. B (Proc. Suppl.) 53, 368(1997); H. P. Shanahan et al., Phys. Lett. B453, 239(1990.
[20] Gottlieb et. al., Phys. Rev. D35, 2531(1987).
[21] G. Bodwin, E. Braaten, G. P. Lepage, Phys. Rev D51, 1125(1995), Erratum ibid. D55, 5853(1997).

김 세 용

1985 서울대학교 물리학과 학사
1987 Columbia 대학교 석사
1991 Columbia 대학교 박사
1991 ~ 1995 미국 Argonne 국립연구소 연구원
1995 ~ 1997 서울대학교 이론물리연구소 연구원
1994 ~ 1996 일본 이화학연구소 빙문연구원
1997 ~ 1998.2 한국과학기술원 부설 고등과학원 연구원
1998 ~ 현재 세종대학교 물리학과 부교수
관심분야 : 입자 물리 이론(격자 게이지이론), 강한 핵력, 전산물리, High Performance Computing (Vector and parallel)
E-mail:skim@dasan.sejong.ac.kr

● 제2회 한국 소프트웨어공학 학술대회 ●

• 일 자 : 2000년 2월 24 ~ 25일
• 장 소 : 피닉스 파크 컨벤션 센터
• 주 최 : 소프트웨어공학연구회
• 문 의 처 : 한국전자통신연구원 신규상 팀장
Tel. 042-860-6566