



구조 해석 분야에서의 슈퍼 컴퓨팅 기술 동향

김 승 조*

1. 구조해석과 슈퍼컴퓨터(Structural Analysis and Supercomputers)

현재 유한요소 구조해석법으로 널리 알려진 구조해석 기법은 초기에 매트릭스 구조해석법¹⁾이라는 이름으로 시작되었다. 1940년대에 들어서면서 각 분야에서의 구조물 설계가 크게 복잡해지고, 그 안정성 해석이 중요한 문제로 대두되면서 자연스럽게 매트릭스를 이용한 해석기법이 고안되었고, 마침 디지털 컴퓨터(Digital Computer)가 탄생하여 발전되고 있었기에 자연스럽게 매트릭스 연산에 뛰어난 능력을 발휘하는 컴퓨터를 구조해석 분야에 능동적으로 활용하게 되었다. 특히, 2차대전 후 항공우주산업의 급격한 발전을 통해 대단히 복잡한 구조를 가진 여객기, 전투기 그리고 우주선 등의 설계가 활발히 이루어지면서 항공우주산업체에서는 이들의 조직적인 해석에 가장 능동적으로 고가의 컴퓨터를 활용하게 된 것이다.

슈퍼컴퓨터(Super Computer)가 당대의 가장 고성능의 컴퓨터를 일컫기에, 당연히 당대의 가장 복잡한 문제인 항공기 설계시의 구조 안정성 해석에 당시의 최고성능 컴퓨터들, 즉 슈퍼컴퓨터의 성능

을 빌리게 된 것이다. 특히, 달착륙을 위한 아폴로 우주선의 설계를 위해서 NASA에 의해 개발된 Nastran²⁾ 프로그램은 당시 과학기술분야 해석 프로그램으로서 가장 방대하다고 알려졌으며, 새롭게 탄생한 각종 슈퍼컴퓨터들에는 모두 Nastran이 어떤 프로그램보다도 먼저 포팅되었고, 컴퓨터의 CPU 활용비율에서도 상당한 계산시간이 구조해석에 할애되었다. 저 유명한 CDC Cyber 6600, 7600 그리고 CRAY 슈퍼컴퓨터들로 이어지는 당대의 최고 수준의 신제품 슈퍼컴퓨터들에 주요 구조해석 프로그램들이 장착되었던 것이다. 최근에 들어서도 비선형 구조해석, 충돌 구조해석 프로그램 등이 슈퍼컴퓨터의 성능을 벤치마킹(Bench Marking)하는 주요 아이템이 되기도 하였다.

2. 구조해석의 슈퍼컴퓨팅(Supercomputing on Structural Analysis)

구조해석에 많이 사용되는 소프트웨어로는 앞에서 언급한 Nastran을 상용화한 MSC/NASTRAN²⁾, 건설토목분야에의 응용으로 출발한 ANSYS³⁾, 비교적 늦게 비선형 해석으로 특화된 ABAQUS⁴⁾ 그리고

* 서울대학교 기계항공 공학부, 교수

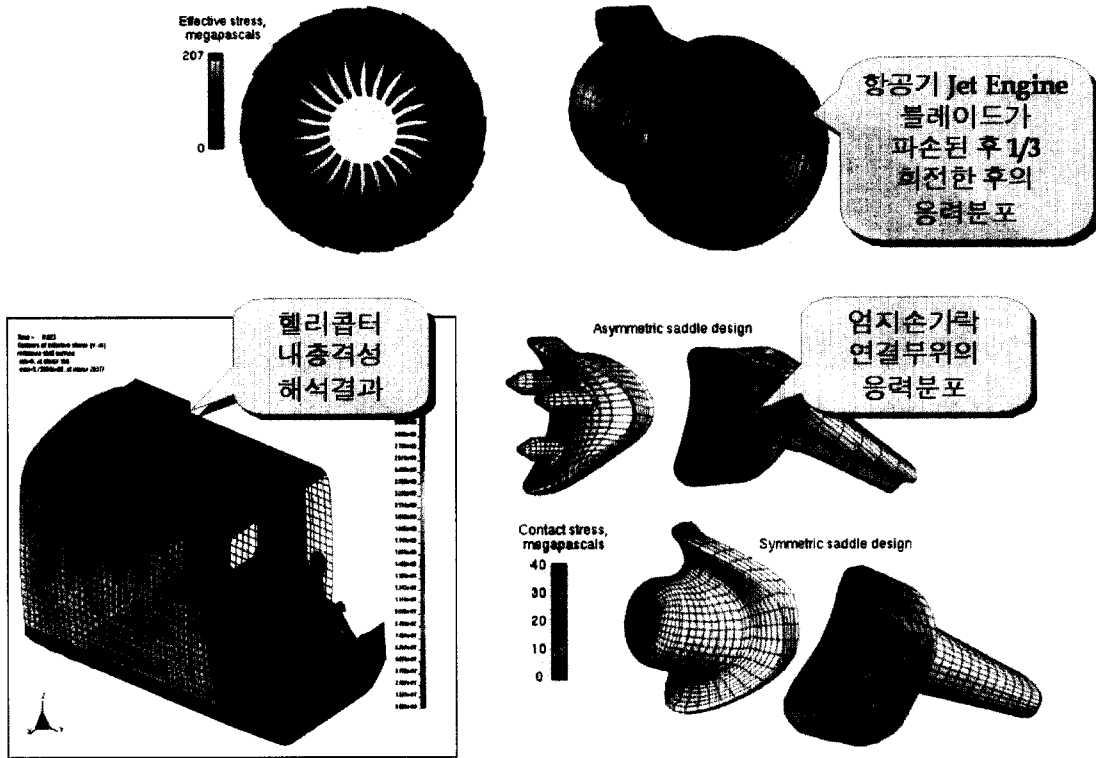


Fig. 1 Finite Element Method의 다양한 활용분야

충돌 시뮬레이션에 많이 사용이 되는 LS/DYNA⁵⁾와 PAM/CRASH⁶⁾ 등을 들 수 있다. 이들은 주로 고성능의 슈퍼컴퓨터에 설치되어 구조해석 분야에서의 슈퍼컴퓨팅에 활용되고 있다. 이들 프로그램들은 기본적으로 중앙 집중식 컴퓨터들(스칼라형, 벡터형)을 토대로 핵심 알고리즘이 개발되었기 때문에 상당 수준의 병렬성능을 요구하는 현재의 MPP (Massively Parallel Processors)⁷⁾ 형태의 슈퍼컴퓨터들의 성능을 완전히 활용하지는 못하고 있다. LS/DYNA와 같은 충돌해석용 프로그램들은 시간적분에서 완전히 외연적(Explicit) 기법을 사용하고 있으므로 비교적 용이하게 병렬화가 이루어져 일부 응용문제에서 좋은 병렬효율을 보이고 있지만, 일반적인 구조물의 충돌 시뮬레이션(Simulation)에서는 아직 기대할만한 효율이 나오지 않고 있다. MSC/NASTRAN이나 ABAQUS가 병렬 기능들을 가지기 시작하고 있지만, 자기들이 제시하는 벤치마킹용 문제 이외의 일반적인 문제에서는 병렬 효

율이 기대한 만큼 나오지 않고 있으며, 8개 정도까지의 병렬계산에서만 어느 정도의 병렬 효율을 얻을 수 있고, 50개 100개 이상의 CPU를 활용하는 비교적 큰 병렬 컴퓨팅에서는 아직 충분한 병렬 효율을 얻을 수 없다. 다시 말하면, CPU를 100개, 1000개 이상으로 밀집시켜 만들어진 슈퍼컴퓨터들에서는 아직까지 100만, 1억씩의 미지수를 유발하는 초대형 고정밀 구조해석 문제들이 쉽게 풀리지 못하고 있다. 특히 대부분이 내재적(Implicit) 기법을 주로 사용하는 유한요소 구조해석 문제들은 강성 매트릭스(때로는 질량 매트릭스도)를 어떤 형태로든 저장하고 있어야 하기 때문에 초대형 문제인 경우, 강성 매트릭스의 저장에 대규모의 메모리가 필요하게 되어서, 효율적인 병렬 알고리즘이 적용되더라도 메모리 부족이라는 난관에 부딪혀서 계산이 불가능한 경우도 많다. 따라서 현재에는 일부의 구조해석문제(병렬처리가 용이한 충돌해석 등)를 제외하고는 대부분이 수

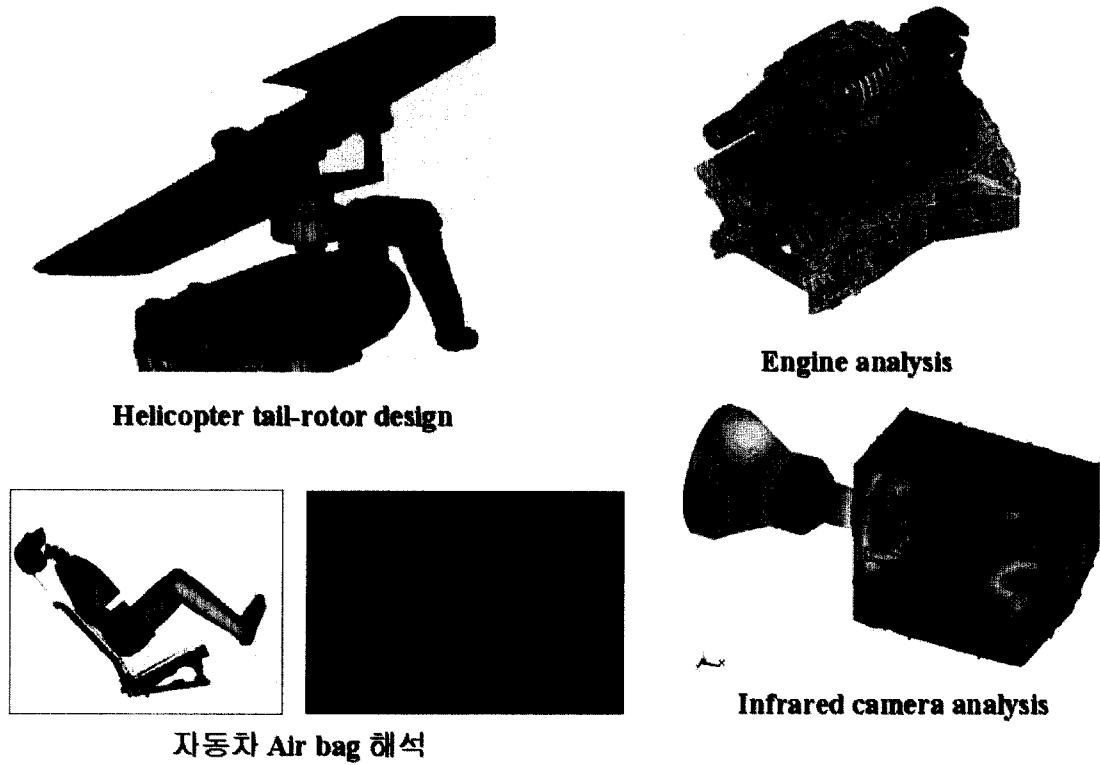


Fig. 2 FEM과 슈퍼컴퓨팅 기술을 이용한 해석사례

퍼컴퓨팅의 스케일에 걸맞은 계산은 이루어지고 있지 않다고 본다. 다시 말하면 슈퍼컴퓨터의 초기 발전을 이끌고 또한 슈퍼컴퓨팅 이용에 있어서 항상 Top list에 올랐던 복잡한 구조물의 유한요소 해석이 슈퍼컴퓨팅 파워가 필요한 응용분야에서 점점 제외되어 가고 있다. 이는 현실적으로 설계 현장에서 설계된 구조물을 아주 자세하게 해석하지 않아도 그간의 설계경험을 활용하면 구조물의 안전을 확인하는데는 충분한 결과를 얻을 수 있고, 또한 앞에서 언급한 더욱 자세한 유한요소 구조해석은 슈퍼컴퓨팅 구현 시의 난관 때문에 구태여 시간과 비용을 들여가면서 더 정밀한 결과를 얻으려 하지 않으려는 경향과 큰 관계가 있다고 본다.

그러나 인간의 욕심은 무한하고, 훨씬 값싼 가격의 고성능 컴퓨터가 지속적으로 개발되고 있고, 이에 따라 이제는 고정밀 해석에 시간과 비용이 크게 들지 않게 되면서, 다시금 슈퍼컴퓨팅을 통한 정밀구조해석의 시도가 일부에서 일어나고 있다.

다시말하면, 초대형 구조해석 문제를 풀기 위해 현재의 하드웨어 환경에 적합한 새로운 틀의 구조해석 병렬 알고리즘 개발이 진행되고 있다. 현재, 국내외적으로 초대형 구조해석기술을 개발하기 위한 연구로 다음 등을 언급할 수 있다. 미국의 경우, ASCI 프로그램⁸⁾을 통해서 개발된 1,000개내지는 10,000개 가까운 CPU를 가진 초병렬 컴퓨터들이 여러 대 있고, 이들 컴퓨터들을 대상으로 병렬구조 해석을 하려는 시도가 많이 이루어지고 있다. 대표적으로 분산 병렬 구조해석 프로그램개발을 시도하는 SALINAS⁹⁾ 계획을 들 수 있다. 일종의 영역분할기법(DDM, Domain Decomposition Method) 알고리즘을 기본으로 하여 개발되고 있으며, 궁극적으로 1000만 이상의 자유도를 가진 구조물의 진동해석을 수행하기 위한 것이다. 보잉(Boeing)사의 팬텀워크 연구 그룹¹⁰⁾에서도 앞으로 3~4년 내에 1000만 미지수의 고유치 해석 문제를 풀기 위해서 노력하고 있다. 유럽에서는 Europort¹¹⁾라는 프로젝트를 통해 응용

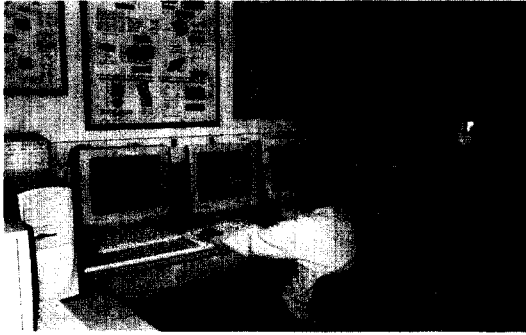


Fig. 3 인터넷 슈퍼컴퓨팅 실험장치
(서울대학교 항공우주공학과 항공우주구조연구실)

프로그램수준에서의 병렬 슈퍼컴퓨팅 소프트웨어 개발이 이루어졌으며, 일본에서 수행하고 있는 병렬 슈퍼컴퓨터를 이용하여 1억개 미지수 수준의 구조해석 문제를 풀기 위한 ADVENTURE¹²⁾라는 프로젝트도 언급될 수 있다. 국내의 경우, 서울대학교 항공우주구조연구실 팀이 국가지정연구실 사업으로 Internet Supercomputing이라는 새로운 개념의 슈퍼컴퓨팅으로 앞으로 4년 이내에 인터넷 상에 연결되어 있는 PC들을 활용하여 1024개 이상의 CPU로 1억 개 미지수 수준의 문제를 풀 수 있는 IpSAP(Internet Parallel Structural Analysis Program) 개발을 추진하고 있다.

3. 구조해석 분야에서 사용되는 병렬 슈퍼컴퓨팅 알고리즘(Supercomputing Algorithms of Structural Analysis)

앞에서 이미 언급한 것처럼, 최근의 슈퍼컴퓨터들은 몇개의 초고성능 CPU로 이루어져 있지 않고, 고성능의 CPU를 수 백개에서부터 수 천개까지를 활용하여 이루어져 있으므로, 효율적인 병렬 알고리즘을 이용하지 않고는 성능을 제대로 발휘하기 힘들다. 따라서, 슈퍼컴퓨팅의 구현에 있어서 슈퍼컴퓨터를 구성하는 하드웨어적인 기술뿐만이 아니라, 효율적인 병렬 알고리즘의 개발 또한 매우 중요하다. 그렇기 때문에 구조해석분야에서도 효율적인 병렬 알고리즘을 개발하기 위한 노력이 국내 외적으로 많이 시도되고 있는 것이다. 먼저 선형정적 해석(Linear static analysis)의 경우를 살펴보자.

유한요소법을 사용한 구조해석의 경우에, 대부분의 시간은 강성행렬과 하중벡터로 구성된 연립방정식을 푸는 과정에서 소비된다. 그래서 효율적인 선형 연립방정식의 해법에 대한 연구가 많이 진행되어 왔다. 이러한 선형정적 해석을 위한 연구는 크게 직접적 해법과 간접적 해법으로 다시 나누어 볼 수 있다. 직접적 해법은 선형연립 방정식의 해를 가우스 소거법 등을 이용하여 구하는 것으로, 수치적으로 안정하다는 장점이 있는 반면에 효율적으로 병렬화를 구현하기가 매우 어렵다는 단점이 있다. 간접적 해법인 경우에 연립방정식을 풀기 위해서, 초기 해를 가정하고 원하는 정확도를 얻을 때까지 반복적으로 계산을 수행하는데, 병렬화가 용이한 행렬연산으로 주로 구성되므로 병렬화 연구가 상당히 많이 수행되었다. 그러나 반복적 해법의 경우에는 수렴성이 보장되지 않는다는 단점을 가지고 있어서 이를 극복하기 위한 연구가 많이 진행되었다. 직접적인 해법의 경우에는 유한요소 구조해석법의 특징을 고려한 Frontal Solver¹³⁾와 Multifrontal solver¹⁴⁾가 개발되었다. 특히, Multifrontal Solver의 경우에는 FDM(Finite Difference Method)나 FVM(Finite Volume Method)에 비해 유한요소법의 가장 큰 문제로 지적되던 거대 메모리 요구 문제를 어느 정도 해결했을 뿐만 아니라, 계산 속도도 상당히 빨라서 상용 구조해석 소프트웨어들이 채택하기도 했다. 최근에는 이러한 Multifrontal Solver에 기반한 Parallel Multifrontal Solver¹⁵⁾가 개발되어 슈퍼컴퓨팅에 활용되고 있다. 반복적 해법의 경우에는 Preconditioning 기법을 통해 안정성을 향상시키고, DDM을 활용한 PCGM(Preconditioning Conjugate Gradient Method)¹⁶⁾를 많이 사용하고 있으며, 최근에는 FETI(Finite Element Tearing and Interconnecting)¹⁷⁾ 해법의 근간으로 활용되고 있다.

구조물의 진동특성 파악을 위한 고유치 해석의 경우에도 자유도가 커지면 그 계산량도 엄청나게 증가한다. 이를 해결하기 위해 병렬화 연구가 많이 진행되고 있다. 해법의 특징에 따라 분류하면 크게 직접 해법(Direct method)과 근사 해법(Approximate method)으로 나뉘며, 직접 해법으로는 Jacobi method, Householder method 등이 있으며, 근사

해법으로는 Lanczos method¹⁸⁾, Sub-space iteration method¹⁹⁾ 등이 있다. 특히 구조물이 3차원 구조로 복잡한 형태를 지니고 있으면, 기억용량과 계산 시간은 더욱 커질 뿐만 아니라, 필요로 하는 고유모우드의 숫자가 증가하거나 보다 정확한 해를 필요로 할 경우에도 모델의 크기와 해석 시간은 크게 증가한다. 또한 최적설계 같이 반복적인 해석이 필요한 경우에는 해석 시간이 설계 작업의 효율에 영향을 미친다. 따라서, 이전부터 고유치 문제의 빠른 해결을 위해서 자유도 축소법, 부분 해석법 등 여러 기법이 연구되어 왔는데, 최근에는 보다 정확한 해의 계산과 정적 해석법에서 구현되는 병렬처리 알고리즘 등과의 효과적인 연계를 고려하여 Lanczos method와 Blocking 기법을 이용한 Block Lanczos 알고리즘²⁰⁾이 초대형 구조물의 고유치 해석에 널리 활용되고 있다.

구조물의 외부하중 등에 의한 시간응답을 구하는 시간영역 해석은 구조해석 분야 중에서 병렬화에 대한 연구와 실제 응용이 가장 많이 이루어진 분야라고 볼 수 있다. 시간 영역 해석을 위한 알고리즘도 역시 크게 내재적(Implicit) 기법과 외연적(Explicit) 기법으로 나눌 수 있는데, 외연적 기법은 다음 시간 스텝(Δt_{n+1})에서의 변위를 계산하는데, 현재 시간스텝(Δt_n)에서의 값들만을 필요로 하기 때문에 연산에 연립방정식을 푸는 과정이 필요 없으며, 대부분의 계산이 행렬연산으로 이루어지게 되므로 손쉽게 병렬화를 구현할 수 있다. 그러나 해의 안정성이 조건적으로 보장되므로, Δt (시간 증분의 크기)를 크게 할 수 없다는 단점을 안고 있다(CFL Number condition)²¹⁾. 따라서, 이 외연법은 외부하중에 의한 초기의 시간응답이나 충격에 의한 구조물의 손상 등의 연구에 많이 활용되고 있으며, 항공기뿐만 아니라 자동차 등의 설계에도 널리 활용되고 있다. 내재적 기법의 경우에는 다음 시간스텝에서의 변위를 계산하는데, 현재 시간스텝(Δt_n)에서의 값들뿐만 아니라 다음 시간스텝(Δt_{n+1})에서의 변위도 함께 필요하게 되므로 연립방정식을 풀어야 한다. 그러나 해의 안정성이 무조건적으로 보장되므로 소성가공과 같은 비교적 긴 시간동안의 구조물의 응답특성을 파악하는데 많이 활용될 수 있다. 그런데 구조물의 동적 해석에 외연적 방법을

사용하게 되면, 해의 안정성을 보장하기 위해서 CFL condition이라는 것을 만족시켜야 하는데 고정밀 해석을 위해 유한요소 격자를 더욱 정밀하게 만들수록 시간스텝 간격(Δt)을 작게 해야 하므로, 한번의 시간방향 전진을 위해서 필요한 계산량도 커질뿐만 아니라 원하는 시간에서의 거동을 살펴보기 위해서 풀어야 하는 시간스텝의 수도 같이 늘어나서 상당한 어려움이 있다. 그래서 구조물의 충격 혹은 동적 접촉문제를 적절히 해석할 수 있도록 급작스러운 거동을 불연속으로 가정하여 개발된 불연속 시간적분법(DTIM, Discontinuous Time Integration Method)²²⁾이 개발되었는데, 기존의 Newmark류의 알고리즘보다 수치잡음이 훨씬 작을뿐만 아니라, 시간스텝 간격(Δt)도 크게 할 수 있어서 초대형 고정밀 해석에 매우 적합하다. 내재적 방법을 사용할 경우에도, 한번의 시간증분시의 계산에서는 선형정적 해석과 같은 메모리문제와 효율적인 병렬화 구현의 어려움이 있어서 외연적 기법에 비해서 상대적으로 널리 활용되어지지 못하고 있는데, 최근에는 고차의 정확도를 갖는 동시에 무조건적인 안정성을 가지며, 한번의 시간증분시 여러 시간대의 거동을 한꺼번에 구할 수 있어서 초대형 고정밀 해석에 적합한 사영시간적분법(PTIM, Projective Time Integration Method)^{23),24)} 개발되었다.

비선형 해석을 위한 알고리즘의 경우에는, 해석하고자 하는 문제와 고려하고자 하는 비선형 효과에 따라 굉장히 다양한 알고리즘이 개발되고 연구되어 왔는데, 대부분의 비선형 해석이 선형해석을 반복적으로 수행하여 이루어지게 된다. 그렇기 때문에 효율적인 선형정적 분산병렬처리 알고리즘을 활용하면 비선형해석도 쉽게 병렬화되어 슈퍼컴퓨팅에 도전할 수 있게 된다.

4. 슈퍼컴퓨팅의 미래

- 인간 상상력과 도전정신의 한계만이 슈퍼컴퓨팅의 유일한 한계

미래의 슈퍼컴퓨팅의 방향과 흐름에 대해서 살펴보기 위해서는 가장 먼저 살펴봐야 할 것은 어떤

형태로 슈퍼컴퓨팅이 이루어 질 것인가 하는 방법론에 관한 것이다. 우선, 앞에서도 여러 번 언급을 했지만 이제 더 이상 초고성능 CPU를 기본으로 하는 슈퍼컴퓨터는 개발되지 않을 것이다. MPP와 SMP(Symmetric Multi-Processors)⁷⁾ 형태의 슈퍼컴퓨터도 그 경쟁력을 당분간은 유지할 것이지만, 새로운 형태의 슈퍼컴퓨팅 구현을 위한 연구는 끊임없이 계속되어 변화하는 컴퓨터를 둘러싼 환경에서 적합한 슈퍼컴퓨팅 구현법은 계속 개발될 것이다. 여기서 새로운 슈퍼컴퓨팅 구현을 위해서 반드시 고려해야 할 중요한 변화를 살펴보고자 하자. 지난 10년 동안 마이크로 프로세서의 성능향상은 가히 경이적이라고 볼 수 있을 정도로 눈부시게 발달하였다. 이러한 발달은 개인용 컴퓨터(Personal Computer)의 보급과 대중화에 기반이 되었다. 특히, 최근에 PC에 장착되고 있는 CPU들은 그 성능이 현재 최고수준의 슈퍼컴퓨터를 구성하는 CPU의 성능을 능가하고 있다. 일례로, Pentium III 1GHz 프로세서의 경우에, SSE(Streaming SIMD Extensions) 기능을 이용할 경우에 부동소수점 연산에서 이론 최대 성능이 4Gflops(floating point operations per second)까지 나올 수 있는 반면(실제로 수치실험을 통해 확인한 결과 행렬간의 곱셈연산인 경우에 2Gflops 정도까지는 무난하게 달성할 수 있는 것으로 보인다), Cray T3E1200 슈퍼컴퓨터를 구성하는 Alpha 600Mhz CPU의 이론 최대성능이 1.2Gflops이며, IBM SP 슈퍼컴퓨터를 구성하는 Power3 375 Mhz CPU의 경우에는 이론 최대성능이 1.5 Gflops 정도라는 것을 생각하면 PC파워가 엄청나다는 것을 알 수 있다. 따라서 PC는 더 이상 단순한 사무용이나 개인업무용으로만 사용되지 않고, 슈퍼컴퓨팅이나 대용량 데이터 베이스 처리업무에도 능히 사용될 수 있는 것이다. 특히, 최근에는 뛰어난 안정성과 성능을 가진 리눅스라는 공개 운영체제가 개발되고 활용되면서 이러한 PC들을 활용한 Clustering 기술이 각광을 받기도 했다. 개인용 컴퓨터들의 CPU 성능의 눈부신 발전보다 오히려 더 개인용 컴퓨터의 활용분야를 다양하게 만든 것은 인터넷으로 대변되는 네트워크의 확산일 것이다. 우리나라의 경우에도 기업체의 컴퓨터나 각종 연구소의 컴퓨터들은 거의 대부분 인터넷에 연결되어 있으며,

일반 가정의 컴퓨터들도 초고속 인터넷에 연결되어 우리의 생활까지도 상당히 바꾸어 놓고 있는 상황이다. 이러한 급속한 네트워크의 확산과 인터넷의 보급은 더욱 빠른 인터넷을 구현하기 위한 끊임없는 개발과 연구를 자극하고 있으므로, 멀지 않은 미래에는 지금의 네트워크 속도와는 비교도 되지 않을 정도의 빠른 속도로 인터넷을 사용할 수 있을지도 모른다.

미국을 비롯한 슈퍼컴퓨팅 선진국들에서는 이러한 변화를 고려하여 앞에서 소개한 Clustering 기법과 PC들을 이용한 Beowulf Cluster와 같은 소규모 연구실 단위의 슈퍼컴퓨팅 기법과 여기 저기 흩어져 있는 초대형 병렬 슈퍼컴퓨터들을 서로 엮어서 하나의 컴퓨터로 활용하는 GRID Computing²⁵⁾과 같은 새로운 형태의 슈퍼컴퓨팅 파워 구현에 관한 연구를 추진하고 있다. 우리나라의 경우에는 인터넷에 연결된 수많은 고성능 PC들의 컴퓨팅 파워를 이용한 새로운 개념의 Internet Supercomputing¹⁵⁾ 기법이 개발되어 활발하게 연구되고 있다. Internet Supercomputing은 우리나라 같이 슈퍼컴퓨터와 같은 인프라가 상대적으로 열악한 환경에서도 충분히 초대형 슈퍼컴퓨팅을 구현할 수 있다는 장점이 있다. 고성능 CPU를 장착하고 네트워크로 서로 연결된 PC들이 서로간에 데이터를 빠르게 주고받을 수 있는 상황에서는, 이들을 조직적으로 통합하고 제어할 수 있다면 인터넷 상의 가상의 슈퍼컴퓨터를 구성할 수 있는 것이다. 그리고 각각의 PC들은 이미 개인 사무용이나, 컴퓨터 그래픽 등 원래 고유의 목적으로 이미 활용되고 있는 컴퓨터이므로, Clustered system과 같이 별도의 추가비용을 필요로 하지 않는 매우 경제적인 방법이다. 또한 인터넷에 연결되어 있는 수많은 PC들을 이용하여 가상의 슈퍼컴퓨터 시스템을 구성하는 것이므로, 이론적으로는 무한대의 CPU를 이용할 수 있다. 그리고 Fig. 4에 나타나 있는 것과 같이, Internet Supercomputing은 개념적으로는 개인용 PC뿐만이 아니라, 기존의 슈퍼컴퓨터는 물론이고 Clustered system도 계산 자원으로 활용할 수 있으므로 GRID Computing보다 더 상위의 개념이라고 볼 수 있다. 이미 이러한 Internet Supercomputing을 이용하여 기존의 슈퍼컴퓨터에서도 풀기가 어려운 초대형 구조해석 문

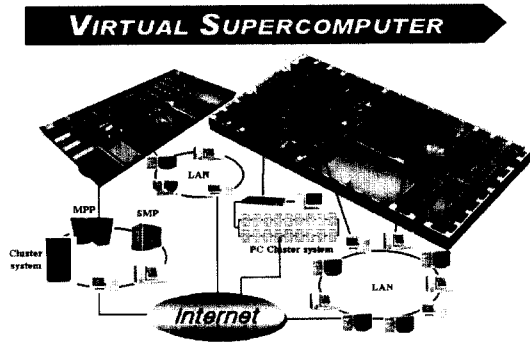


Fig. 4 Internet Supercomputing의 개념도

제를 성공적으로 해석하여, 그 활용가능성 및 타당성에 대한 검토가 수행되었다. 한편, 인터넷 슈퍼컴퓨팅의 경우에는 주로 퇴근시간이나 휴일 등을 이용하여, PC들이 사용되지 않는 시간대를 이용하여 수행되므로, 비선형해석과 같이 오랜 시간을 필요로 하는 슈퍼컴퓨팅에는 널리 활용될 수 없다는 단점을 극복한 Piggyback Supercomputing도 최근에 시도되고 있다. Piggyback Supercomputing의 경우에는 SePC(Server-embedded PC)를 기반으로 한 것으로, 개인사용자가 자신의 PC를 이용한 여러 가지 업무를 수행함과 동시에 병렬컴퓨팅 계산도 수행할 수 있기 때문에, 업무시간이나 퇴근시간에 상관없이 계산을 수행할 수 있다는 점이 최대의 장점이다.

지금까지 살펴 본 바와 같이, 이제 가까운 장래에는 더 이상 하드웨어적인 제한보다는 오히려 인간의 상상력과 도전정신이 얼마나 큰 슈퍼컴퓨팅을 수행할 것인가 하는 문제의 유일한 제약조건이 될 것이다. 슈퍼컴퓨팅 파워는 Internet Supercomputing과 같은 새로운 방법으로 접근하게 되면, 인간이 원하는 만큼 거의 무한대로 확보될 수 있을 것이며, 인간의 욕심만큼 모든 물리적 현상을 다 고려한 해석이 가능할 것이다. 이렇게 초정밀 초대형 해석이 저렴한 비용과 적절한 시간 내에서 수행할 수 있게 되면, 구조설계해석에서도 지금까지와 같이 전체 구조물의 일부나 부품단위별로 수행되던 구조해석이 전체 구조물에 대한 초고정밀 해석으로 대체되어 갈 것이며, 뛰어난 병렬 알고리즘을 통해 예전보다 오히려 더 빠른 시간에 초대형 고정밀 구조해석을 완료함으로써 설계에 보다 더 널리

초대형 고정밀 해석결과가 활용될 것이다. 그리고 모든 비선형 효과와 물리적 현상이 고려되어 실제 구조물의 거동을 완벽하게 시뮬레이션할 수 있는 기술, 즉 CST(Cyber Structures Technology)의 구현과 이의 실제 응용이 가능하게 될 것이다. CST 기술은 우주공간에서의 구조물의 반응이나 분자레벨에서의 현상규명과 같은 실험환경 구성이 힘든 분야에서 지금까지 볼 수 없었던 다양한 현상들을 규명해 내는 가상실험(Virtual Experiment)도 쉽게 가능케 할 것이다. 그리고 이러한 슈퍼컴퓨팅 기술의 보유 유무가 바로 그 국가의 경쟁력과 직결되면서 새로운 선진국이 되기 위한 필수요구조건이 될 것이며, 이러한 슈퍼컴퓨팅 기술을 보유하고 또한 리드해 나가기 위해서는 그 원천이 되는 각 분야의 효율적인 병렬 알고리즘 개발이 최우선적으로 선행되어야 하는 것은 아무리 강조해도 지나침이 없을 것이다.

참 고 문 헌

1. Martin, H.C, *Introduction to Matrix Methods of Structural Analysis*, McGraw-Hill, 1966
2. MSC Software team, <http://www.mechsolutions.com/products/nastran/>
4. Hibbitt, Karlsson & Sorensen, Inc., <http://www.hks.com>
3. Ansys, Inc., <http://www.ansys.com>
5. LSTC Corp., <http://www.ls-dyna.com>
6. ESI Group, <http://www.esi.fr>
7. Michael J. Quinn, *Parallel Computing Theory and Practice*, 2nd Ed., McGRAW-HILL, 1994
8. www.llnl.com/ascii
9. <http://www.cfd.sandia.gov/rdstruct99.html>
10. www.boeing.com/news/feature/fam98/pworks.html
11. <http://hpcc.lerc.nasa.gov/ahpc>
12. <http://adventure.q.t.u-tokyo.ac.jp/>
13. B. M. Irons, A frontal solution program for finite element analysis, *International Journal for Numerical Methods in Engineering*, 2, 5-32, 1970
14. J. H. Kim and S. J. Kim, "A Multifrontal Solver

- Combined with Graph Partitioners”, *AIAA Journal*, Vol. 38, No. 8, August 1999, pp.964~970.
15. Seung Jo Kim, Chang Sung Lee and Jeong Ho Kim, Large-scale Structural Analysis by Parallel Multifrontal Solver Through Internet PCs, 41st AIAA/ASME/ASCE/AHS/ASC Structures, Structural Dynamics, and Materials Conference and Exhibit, The Westin Peachtree Plaza Atlanta, Atlanta, GA, 3-6 April 2000
 16. Johnson, O. G., Michelli, C. A., and Paul, G., “Polynomial preconditioners for conjugate gradient calculations”, *SIAM J. Numer. Anal.*, 20, 362-376, 1983
 17. C. Farhat and F. X. Roux, “A method of finite element tearing and interconnecting and its parallel solution algorithm”, *Int. J. Numer. Methods Eng.*, 32, 1205-1227, 1991
 18. Simon, H. D., “The Lanczos Algorithm with Partial Reorthogonalization”, *Mathematics of Computation*, Vol. 42, pp.115~142, 1984
 19. K. J. bathe, E. L. Wilson “Large Eigenvalue Problems in Dynamic Analysis”, *A.S.C.E., J. of Engineering Mechanics Division*, Vol. 98, 1972, pp.1471~1485
 20. Grimes, R. G., J. G. Lewis and H. D. Simon, “A Shifted Block Lanczos Algorithm for Solving Sparse Symmetric Generalized Eigenproblems”, *SIAM Journal on Matrix Analysis and Applications*, Vol. 15, pp.228~272, 1994
 21. Huges, T. J. R., *The Finite Element Method - Linear Static and Dynamic Finite Element Analysis*, Prentice Hall, Englewood Cliffs, NJ, 1987
 22. J. Y. Cho and S. J. Kim, “A Discontinuous Time Integration Method for Dynamic Contact/Impact Problems”, *AIAA Journal*, Vol. 37, No. 7, pp.874~880, July 1999
 23. Kim, S. J., Cho, J. Y. and Kim, W. D., “From the Trapezoidal Rule to Higher Order Accurate and Unconditionally Stable Time Integration Method for Structural Dynamics”, *Computer Methods in Applied Mechanics and Engineering*, Vol. 149, Nos1-4, pp.73~88, 1997
 24. Kim, S. J. and Cho, J. Y., “Penalized Weighted Residual Method for the Initial Value Problems,” *AIAA Journal*, Vol. 35, No. 1, pp.172~177, 1997
 25. Poster, I., and C. Kesselman, eds., *The Grid: Blueprint for a New Computing Infrastructure*, Morgan Kaufman, Pub. August, 1998 