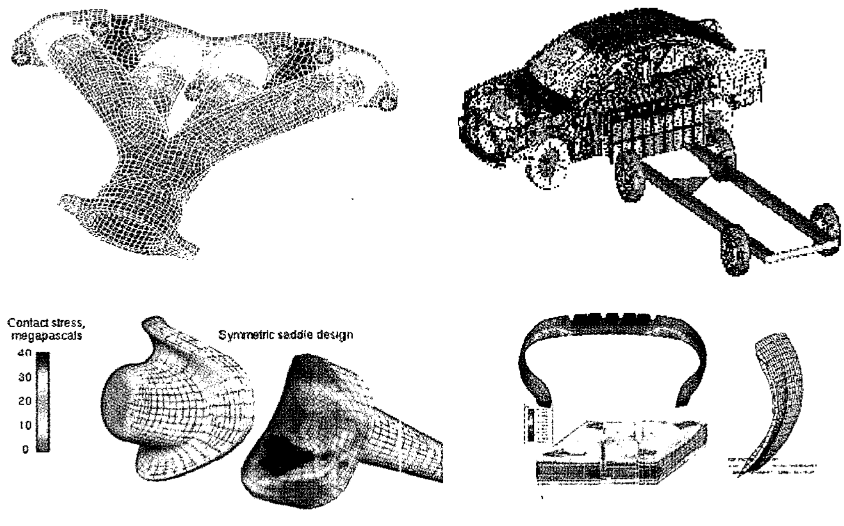
구조해석 기법의 병렬화

• 김정호 ▌KISTI 슈퍼컴퓨팅응용실, 선임연구원 / e-mail:aresto@hpcnet.ne.kr

날로 적용 범위가 넓어져 가고 있는 유한요소법을 이용한 구조해석에 있어서 병렬화의 필요성은 어느 때보다도 커졌다. 이에 따라 이 글에서는 유한요소법의 병렬화 기법들을 소개한다.

컴퓨터를 가장 일찍부터 적극적으로 활용한 분야 중의 하나가 바로 기계, 항공기, 건축물 등 구조물의 역학적인 거동을 계산하는 구조해석 분야이다. 이 분야의 핵심적인 수치기법인 유한요소법은 컴퓨터가 출현하기 이전부터 구조물 설계를 위해 사용되어 오던 매트릭스 구조해석법에 그 근간을 두고 있다. 복잡한 구조물의 거동을 간략히 살피기 위해서 구조물을 트러스, 보 등 간단한 요소로 나타내고 이를 행렬식으로 조립하여 계산하였는데 이와 같이 기본적으로 행렬을 다루도록 되어있는 매트릭스 구조해석법은 컴퓨터로 프로그래밍하기에 매우 적합하게 되어 있었고 따라서 컴퓨터의 발전과 함께 체계화되면서 유한요소법으로 발전하게 된 것이다. 유한요소법은 체계화를 통해 트러스, 보 등과 같은 단순한 구조적 요소뿐만 아니라 연속체와 그 내부에서 일어나는 다양한 물리현상을 표현할 수 있는 보다 일반적인 요소들을 사용할 수 있도록 발전되었다. 이에 따라 그 적용 범위도 크게 확장되어 현재는 단순히 하중에 대한 구조물의 변형, 응력 등을 계산하는 구조해석뿐만 아니라 재료의 소성변형, 충격거동 등 보다 복잡한 역학적 현상을 규명하는 데까지 널리 활용되고 있다. 이와 같이 유한요소법의 적용범위가 확장되고 풀고자 하는 현상이 복잡해짐에 따라 유한요소법을 적용한 문제를 풀기 위한 계산자원의 규모 및 계산량이 급격히 늘어나고 있고 이러한 상황에서 단순히 단일 프로세서



유한요소법의 다양한 활용 예

의 시스템 규모 및 성능 발전 속도에만 의존하고 있어서는 늘어나는 계산량과 계산자원의 규모에 대한 요구를 더 이상 감당할 수가 없게 되었다. 뿐만 아니라 병렬형 시스템이 점점 더 일반화되어가고 있는 상황에서 이러한 병렬형 시스템을 제대로 활용해서 성능을 내도록 하기 위해서는 병렬화는 필수적이라고 할 수 있다. 프로그램이 병렬화가 되어 있지 않다면 아무리 고성능의 수퍼컴퓨터라 하더라도 무용지물이 되기 때문이다. 따라서 유한요소법을 이용한 구조해석 프로그램의 병렬화의 필요성은 어느 때보다도 커졌다고 할 수 있다. 이에 따라 이의 병렬화를 위한 노력이 있어 왔는데 여기서는 현재까지 개발된 유한요소법의 병렬화 기법들에 대해서 소개하고자 한다.

병렬 프로그래밍 모델

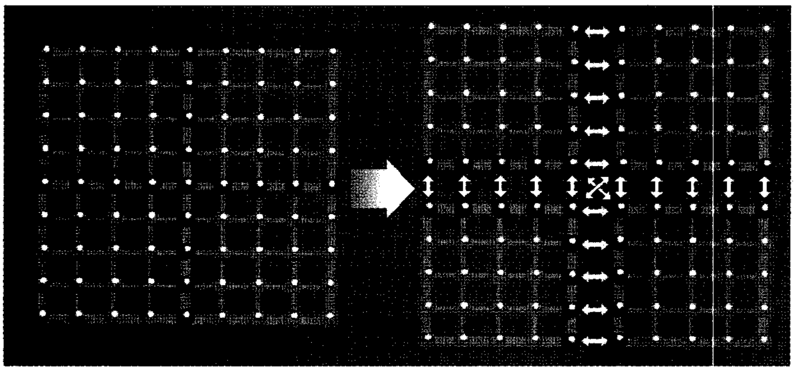
병렬화 기법은 사용하고자 하는 하드웨어의 구조에 따라 공유메모리 병렬 프로그래밍 모델 또는 분산메모리 병렬 프로그래밍 모델을 사용할 수 있다. 공유메모리 병렬 프로그래밍 모델을 사용하여 병렬화를 구현하는 경우는 하나의 메모리를 여러 개의 프로세서가 공유하는 시스템 구조에서 수행되므로 공유되는 데이터는 분리하지 않고 연산만을 독립적으로 동시에 여러 개의 프로세서에서 처리할 수 있도록 분리해주면 된다. 반면 분산메모리 병렬 프로그래밍 모델을 사용하는 경우는 각각의 프로세서가 별개의 메모리 시스템을 가지게 되므로 연산이 병렬로 처리되도록 하기 위해서는 데이터도 완전히 분리가 되어야 하고 각 연산 간의 연관성은 프로세서 간의 통신을 통해 유지해야 한다. 따라서 프로그래밍이 훨씬 복잡하고 많은 노력이 들어가지만 물리적인 메모리 시스템의 크기에 의해 풀 수 있는 문제를 제한받지 않고 프로세서 수를 늘림으로써 그만큼 더 큰 메모리를 확보할 수 있으므로 단순히 계산 성능의 향상뿐만 아니라 순차 프로그램으로 풀 수 없는 보다 큰 규모의 문제를 해결하기 위해서는 분산메모리 병렬 프로그래밍 모델을 쓰는 것이 바람직하다. 뿐만 아니라 공유메모리 병렬 프로그래밍 모델로 병렬화 된 프로그램은 분산메모리 시스템에서 수행하기 곤란하지만 분산메모리 병렬 프로그래밍 모델로 병렬화된 경우에는 공유메모리 시스템에서도 아무 문제없이 수행이 가능하다. 따라서 여기서는 분산메모리 병렬 프로그래밍 모델을 사용하는 것으로 가정하고 병렬화 기법들을 설명하도록 하겠다.

병렬화 알고리듬 개발이라는 관점에서 봤을 때 유한요소법을 사용한 구조해석에서 나타나는 주 계산 과정은 크게 세 가지로 분류될 수 있다. 즉, 외연적 시간적분(explicit time integration) 기법, 내재적 시간적분(implicit time integration) 기법, 그리고 고유치 해석(eigen value analysis) 기법으로 분류할 수 있는데 이는 병렬화와 관련된 수치기법적인 관점에서 본 것이므로 실제 물리적인 문제의 분류 방법과는 다를 수 있다. 이 분류는 유한요소해석 과정에서 공통적으로 포함되는 요소강성행렬 계산 등 요소 단위의 계산을 제외한 나머지 부분의 계산 과정을 기준으로 구분한 것으로 선형연립방정식을 푸는 경우(내재적 시간적분 기법)와 풀지 않는 경우(외연적 시간적분 기법), 그리고 고유치 문제를 푸는 경우로 나눈 것이며 각각의 경우에 따라 병렬화 기법은 크게 달라진다.

외연적 시간적분 기법의 병렬화

외연적 시간적분 기법은 구조물의 외부하중 등에 의한 시간에 따른 응답을 계산하는 데 있어서 다음 시간단계(t_{n+1})의 응답을 계산하기 위해 이미 알고 있는 현재 시간단계(t_n) 및 그 이전의 값들만을 필요로 하기 때문에 시간단계 간의 연성(coupling)이 없어져서 연립방정식을 풀지 않고도 해를 구해 나아갈 수가 있다. 그러나 해의 안정성이 보장되는 시간증분(Δt)의 크기에 제한이 있으므로 시간증분을 임의로 크게 할 수 없다. 대신 연립방정식을 풀지 않기 때문에 각 시간단계를 계산하는 데 필요한 계산량은 내재적 시간적분 기법에 비하여 적으며, 따라서 대부분의 계산 시간은 요소 단위로 각 절점에 가해지는 힘을 계산하는 등 요소 단위의 계산에 소요되게 된다. 이에 따라 요소 단위로 영역을 분할함으로써 쉽게 연산을 병렬화할 수가 있다. 즉, 전체 문제의 영역을 프로

세서 갯수만큼의 부영역(subdomain)으로 쪼갠 다음, 각 영역 내의 요소에 대한 계산을 각각의 프로세서에 할당하는 것이다. 이렇게 되면 하나의 프로세서에 할당된 계산량은 순차적으로 계산할 때에 비하여 프로세서의 갯수만큼의 비율로 줄어들게 되므로 그만큼 빠른 시간에 계산을 끝낼 수 있다. 이 경우 각 절점에서의 값들을 구하기 위해서 그 절점을 공유한 주위의 요소들로부터 계산된 값을 누적해야 하므로 영역간의 경계면에 있는 절점들에 해당하는 값들은 각 영역 내부에서의 계산만으로는 구해지지 않는다. 따라서 그 절점을 공유하는 요소들을 포함한 이웃 영역들과의 데이터 교환을 통해 필요한 데이터를 모아야 한다.



병렬 계산을 위한 영역분할 및 경계 절점에서의 통신

이상에서 본 바와 같이 외연적 시간적분 기법을 쓰는 경우, 병렬화를 비교적 간단하게 구현할 수 있고 그 기법에 대해 특별히 기술적으로 문제가 될만한 부분은 없다고 볼 수 있다. 따라서 병렬화 기법이 쉽게 실용화될 수 있으므로 이러한 외연적 시간적분 기법을 쓰는 LS-DYNA나 PAM-CRASH와 같은 상용 유한요소 해석 소프트웨어의 경우 비교적 일찍부터 병렬화가 잘 되어 있는 편이다. 그러나 시간적분 외에 접촉문제 등과 같은 다른 복잡한 물리적 현상들을 동반하는 실제 문제를 푸는 데 있어서는 이러한 부분에서의 병렬성의 한계로 인하여 기대만큼의 병렬 효율을 내지 못하는 경우가 많다.

내재적 시간적분 기법의 병렬화

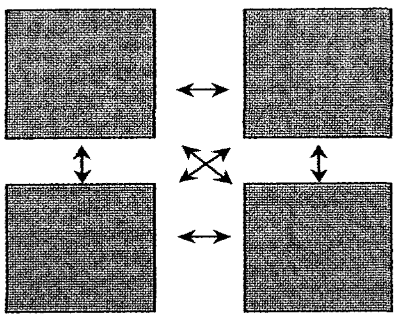
내재적 시간적분 기법은 구조물의 시간에 따른 응답을 계산하는 데 있어서 매 시간단계마다 힘 평형을 고려하게 되므로 다음 시간단계(tn+1)의 응답이 현재(tn) 및 이전 시간단계에서의 값들과 연성(coupling)이 생기게 되어 반드시 연립방정식을 풀어야 한다. 따라서 각 시간단계를 계산하는 데 필요한 계산량은 외연적 시간적분 기법에 비하여 많으나 시간증분(t)의 크기에 관계없이 해의 안정성이 보장된다. 병렬화 기법의 관점에서 봤을 때 정적인 평형상태에서의 응답을 구하는 정적 구조해석 문제의 경우는 연립방정식을 풀어야 하므로 실제로 시간적분이 수행되지는 않지만 내재적 시간적분 기법으로 분류할 수 있다. 내재적 시간적분 기법에서는 결국 연립방정식의 병렬화 기법이 핵심적인 문제가 되는데 이와 관련한 접근 방법은 크게 행렬 기반의 병렬화와 영역 기반의 병렬화 두 가지로 나눌 수 있다.

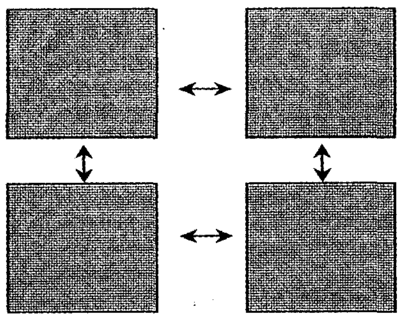
행렬 기반 접근 방법은 요소강성행렬의 조립 등을 통해 이미 연립방정식이 완전히 구성된 다음부터 병렬화를 고려하는 것으로 이는 유한요소법의 병렬화라기보다는 선형연립방정식 해법 자체의 병렬화의 문제이다. 물론 연립방정식 행렬 구성 과정의 병렬화도 고려해야 하지만 이 과정은 외연적 시간적분법의 경우와 마찬가지로 완전히 요소 단위로 진행되므로 그 과정 자체는 병렬화에 있어서 큰 문제가 되지는 않는다. 선형연립방정식 해법의 병렬화에 있어서 연립방정식을 구성하는 행렬을 조밀 행렬(dense matrix)로 풀게 되면 비교적 병렬화를 용이하게 구현할 수 있을 뿐만 아니라 이미 이런 문제를 최적의 성능으로 풀 수 있는 ScaLAPACK 등과 같은 병렬 연립방정식 해법 라이브러리 루틴들이 개발되어 있으므로 이를 활용할 수도 있다. 그러나 실제 유한요소해석에서 나타나는 행렬은 0인 원소를 매우 많이 가지고 있으므로 조밀 행렬로 처리하게 되

는 경우 불필요한 계산들을 너무나 많이 하게 될 뿐만 아니라 기억용량도 지나치게 많이 요구하게 된다. 따라서 실제 유한요소해석 프로그램에서는 이 방법을 쓰기가 매우 곤란하다. 대신 불필요한 기억용량과 계산량을 줄이기 위해서 간단하게는 대각원소로부터 일정한 거리 내에 0이 아닌 원소들이 들어오도록 저장하는 밴드 행렬에서부터 좀더 복잡하게는 스카이라인 행렬을 이용하는 방법 정도를 일반적으로 주로 써오고 있다. 이 경우 조밀 행렬로 처리하는 경우에 비하여 계산 효율(시간 당 연산회수) 및 병렬 효율은 다소 떨어지지만 계산량은 훨씬 적어지므로 결과적으로 전체적인 계산시간은 크게 줄일 수 있다. 최근에는 이보다도 훨씬 더 계산량을 줄일 수 있는 다중 프런트 기법이 내재적 시간적분 기법을 쓰는 몇몇 상용 유한요소해석 프로그램에 적용되고 있으며, 보다 일반적인 문제에 적용할 수 있는 형태의 라이브러리로도 개발되어 있다. PSPASES (http://www.cs.umn.edu/~mjoshi/pspases)라는 이름의 이 라이브러리는 유한요소해석 문제에도 성공적으로 적용할 수 있는 것으로 알려져 있으며, 병렬효율에 있어서도 조밀 행렬로 된 연립방정식을 푸는 경우에 필적할 정도인 것으로 알려져 있다. 이와 같이 결국 유한요소법의 병렬화를 위한 행렬 기반 접근 방법은 유한요소해석 과정에서 나타나는 희소 행렬(sparse matrix)을 효율적으로 풀기 위한 선형연립방정식 해법과 이의 병렬화의 문제가 된다. 이러한 접근 방법은 ABAQUS, NASTRAN 등의 주요 상용 유한요소해석 프로그램에서 채택하고 있다.

내재적 시간적분 기법에 있어서 영역 기반 접근 방법은 연립방정식이 구성되기 이전 단계에서 병렬화를 구현하는 것으로 수학적 관점에서 보았을 때 편미분방정식 수준에서부터 병렬화를 고려하는 것으로 생각할 수 있으며, 이를 영역분할기법(domain decomposition method)이라고 한다. 영역분할기법에 의한 병렬계산 과정을 구조적 관점에서 정리하면 다음과 같다. 먼저 구조물을 프로세서의 갯수만큼의 영역으로 나눈다. 그 다음 나눈 영역 간의 접합면(interface)의 변위값(수학적으로는 편미분방정식의 미지수에 해당)이 일치하도록 접합면의 초기값을 가정하고 이를 각 영역의 경계조건(boundary condition)으로 하여 각 영역을 각각의 프로세서 내에서 독립적으로 푼다. 그 결과 나온 접합면에서의 힘을 각 영역 간의 통신을 통해 합력을 구하고 이 힘이 평형(수학적으로는 미지수의 구배를 일치시키는 효과)을 이룰 때까지 접합면에서의 변위값을 적절히 변경시켜 가며, 이 계산을 반복한다. 이 과정에서 구해진 접합면의 힘의 합력으로부터 새로운 변위를 계산해내기 위해 일반적으로 공액구배법(conjugate gradient method)을 많이 사용한다. 공액구배법은 앞서 설명한 행렬 기반 접근 방법에서 연립방정식을 푸는 방법으로 사용할 수도 있는데 이 경우에 비하여 영역분할기법을 적용한 경우는 문제 전체에 대한 연립방정식에 공액구배법을 적용한 경우에 대해 선조절(preconditioning) 과정의 역할을 하게 되어 수렴 속도를 향상시킬 수 있다.

영역 기반 접근 방법 중 비교적 근래에 개발된 것으로 FETI (finite element tearing and interconnecting) 기법이 있는데 이 방법은 간단히 말해서 접합면의 변위를 일치시키고 힘의 평형을 이룰 때까지 반복 계산하는 일


영역분할기법의 통신 패턴


FETI의 통신 패턴

반적인 영역분할기법과는 반대로 힘의 평형을 먼저 이룬 상태에서 접합면의 변위가 일치할 때까지 반복 계산을 수행하는 방법이다. 구조해석 문제에 이 방법을 쓰는 경우 일반적인 영역분할기법에 비하여 문제의 수학적인 조건이 좋아져서 수렴속도가 더 빠른 것으로 알려져 있다. 또한 통신 패턴도 영역분할기법에 비하여 우수해서 병렬효율도 뛰어나다.

영역 기반 접근 방법은 이와 같이 반복적 계산을 필요로 하기 때문에 문제에 따라 계산량의 예측이 힘들고 같은 문제의 경우도 영역의 개수에 따라 계산량이 달라진다는 문제점이 있다. 그러나 병렬 효율이 행렬 기반 접근 방법에 비하면 우수한 편이므로 MARC나 ANSYS 같은 일부 구조해석 프로그램에서 이 방법(영역분할기법)을 사용하고 있다.

고유치 해석 기법의 병렬화

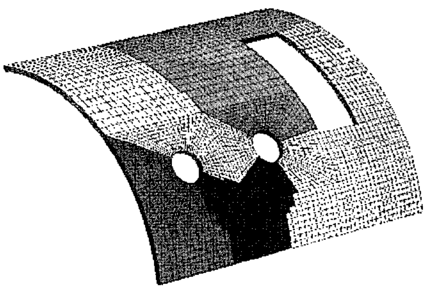
고유치 해석은 일반적으로 구조물의 진동 특성을 파악하거나 좌굴 거동을 예측하기 위해 수행된다. 구조물의 고유치 해석은 유한요소법을 통해 구조물의 특성식(characteristic equation)이 행렬식으로 표현되므로 행렬에 대한 고유치 문제(eigen value problem)로 귀착된다. 행렬의 고유치 문제를 푸는 방법은 매우 다양하지만 크게 행렬 변환 등을 통해 행렬의 모든 고유치 및 고유 벡터를 정확하게 계산해 내는 직접 해법과 부공간 반복법(subspace iteration method) 또는 란초스 법(Lanczos method)과 같이 근사화를 통해 일부 주요 고유치 및 고유 벡터만을 계산해 낼 수 있는 근사 기법으로 나눌 수 있다. 이들 중 병렬화의 대상이 되는 대형 구조해석 문제의 고유치 해석에는 근사 기법이 적합하다. 특히 란초스 법은 ABAQUS, NASTRAN과 같은 상용 소프트웨어에서 주로 사용을 하고 있는데 이 방법의 경우 대부분의 계산 시간을 선형연립방정식을 푸는 데 사용하고 그 외의 계산은 간단한 행렬-벡터 연산이므로 선형연립방정식의 계산 부분만 적절히 병렬화가 되면 전체 과정을 쉽게 병렬화 할 수가 있다. 결국 이 경우 고유치 해석 기법의 병렬화는 내재적 기법에서 행렬 기반 접근 방법의 병렬화 결과가 그대로 적용될 수 있다.

란초스 법 등 행렬의 고유치 해석을 위한 기존 순차 알고리듬의 병렬화를 고유치 문제 병렬화의 행렬 기반 접근 방법으로 생각한다면 고유치 문제 병렬화의 영역 기반 접근 방법으로 크론 법(Kron's method)을 들 수 있으나 실용화되지는 못하고 있다.

병렬화와 관련된 제반 문제점 및 전망

이상에서 유한요소법을 이용한 구조해석에서 주 계산 부분의 병렬화 기법에 대하여 살펴 보았는데 문제에 따라 이러한 부분 외에도 많은 시간을 차지하는 부분이 생길 수 있다. 그 대표적인 경우가 접촉 문제이다. 외연적 시간적분 기법을 쓰는 경우 충돌현상을 다루는 경우가 많은데 시간적분 자체가 그리 많은 계산량을 요구하는 것이 아니므로 이때 접촉 문제의 해결에 걸리는 시간을 무시할 수가 없다. 접촉 문제의 경우 대체적으로 많은 통신량을 요구하기 때문에 병렬효율을 많이 저하시키게 된다.

구조해석 문제의 병렬화에 있어서 병렬효율에 직접적으로 영향을 미치는 중요한 문제의 하나로서 계산할 영역의 분할의 문제가 있다. 외연적 시간적분 기법이나 내연적 시간적분 기법의 영역 기반 접근 방법의 경우처럼 문제를 풀 유한요소 영역을 프로세서 갯수만큼의 영역으로 분할하는 경우, 이 분할이 어떻게 이루어지느냐에 따라 병렬효율은 매우 큰 영향을 받을 수 있다. 일반적으로 각 영역에 같은 수의 요소가 배분되도록 하되, 각 영역 간의 경계면의 절점이 최소화되도록 분할을 해주게 된다. 이를 위해 다양한 알고리듬 및 소프트웨어가 개발되어 있는데 이들 중 Metis라는 소프트웨어가 가장 널리 사용되고 있다. 그러나 영역분할기법이나 FETI 기법을 쓰는 경우 각 영역 내의 요소 수가 같더라도 영역별 계산량은 같지 않은 경우가 많고 또한 경계면의 특성에 따라 수렴속도가 달라질 수도 있기 때문에 병렬 효율을 위해서는 보다 많은 측면을 고려해서 요소를 배분해 주어야 한다. 외연적 시간적분 기법을 쓰는 경우에도 경계면의 절점을 줄이는 것뿐만 아니라 이웃하는 영역의 갯수도 성능에



Metis를 이용한 계산 영역 분할의 예

큰 영향을 끼칠 수 있다.

구조해석의 병렬화와 관련하여 또 하나 고려해야 할 것이 결과의 정확도 문제이다. 일반적으로 병렬화해서 계산한 결과와 순차적인 방법으로 계산한 결과는 어느 정도 차이가 있을 수 있는데 경우에 따라 그 차이가 문제가 될 수도 있다. 영역분할기법이나 FETI를 쓰는 경우 수렴의 정확도를 어느 정도로 지정하느냐에 따라 당연히 결과의 차이가 어느 정도 날 수 있다. 그러나 그 보다 더 문제가 될 수 있는 경우는 외연적 시간적분 기법을 쓰는 경우로 이 경우는 매 시간단계마다 평형을 고려하지 않기 때문에 단계가 진행되어 갈수록 병렬화한 경우와 그렇지 않은 경우의 계산 결과의 차이가 누적되어 커질 가능성이 있다.

이외에도 분산메모리 모델로 구조해석 문제를 병렬화한 경우 사용상의 불편함 등 여러가지 문제점이 많이 있지만 그럼에도 불구하고 병렬화가 제대로 구현된다면 그보다도 훨씬 더 많은 가능성을 가져다 줄 수 있다. 지금까지는 한정된 계산 자원 내에서 실제 구조물을 유한요소 모델로 만들고 결과를 해석하는 과정에서 구조물의 역학적 특성에 대한 깊은 이해와 많은 경험을 가진 전문가의 역할이 매우 중요했던 반면 그 결과를 실제에 적용할 수 있는 범위는 상당히 제한되어 있었다. 그러나 병렬화를 통해 훨씬 더 큰 규모의 계산을 어렵지 않게 할 수 있게 된다면 유한요소 모델링 과정에서 특별한 역학적 가정을 넣지 않고 구조물을 있는 그대로 모델링할 수 있는 여지가 많아지게 될 것이고 계산 결과도 훨씬 직관적이고 실제에 가깝게 얻을 수 있을 것이다. 이에 따라 해석 결과의 적용 범위 또한 훨씬 더 넓어지고 중요해 질 것이다.

기계용어 해설

고무베어링(Rubber Bearing)

면진의 방법으로 가장 널리 사용되는 면진장치이다. 고무와 철판을 적층함으로써 고무의 유연한 특성에 의해 지진의 영향으로부터 구조물을 지반과 분리시키는 역할을 하며, 철판에 의해 상부구조물의 하중을 지탱하는 역할을 한다.

면진(Seismic Isolation)

가진시 교량, 건축물 등이 지진의 영향으로부터 구조물의 피해를 줄이는 방법의 하나로서 지반과 상부구조물을 분리시키는 방법이다. 가장 보편적으로 사용되는 도구로서 고무와 철판의 적층으로 이루어진 면진받침을 사용한다.

분리기법(Partitioning Technique)

연성장 해석시, 각 장의 해석이 순차적으로 이루어질 경우, 즉, 하나의 해석장에서 얻어진 결과가 다른 해석장에서 하중과 같은 입력으로 사용될 경우, 각 장을 분리하여 해석할 수 있다. 이 때 사용할 수 있는 기법이 분리기법이다.

복합재료 주축(Carbon Composite Shaft)

공작기계용 주축이나 자동차용 드라이브 샤프트 등의 고속회전 부품으로 사용되는 탄소섬유복합재료로 제작된 주축을 의미한다. 비강성이 낮은 기존의 강철주축에 비하여 비강성이 높은 탄소섬유 복합재료 주축의 장착으로 복합재료 주축이 장착된 주축부의 고유진동수가 상승하며, 탄소섬유 복합재료의 우수한 진동감쇠 성능으로 주축부의 구동안정성이 향상된다.