

클러스터 구조 특성에 따른 영역분할 알고리즘

The Cluster Characterization on the Domain Decomposition Algorithms

박 대 효* · 탁 문 호** · 이 경 재***

Taehyo Park · Moonho Tak · Kyungjae Lee

요 약

유한요소법은 편미분방정식(Partial Differential Equation)의 수치적 근사 해를 구하기 위한 가장 일반적이고 효율적인 방법으로 다양한 공학 분야에서 널리 사용되어지고 있다. 유한요소법의 해석은 연속적인 범위를 가지는 문제를 여러 개의 요소로 나누어 다항식의 형상함수를 만들게 되며 결과적으로 근사 해를 구하게 된다. 이때 해석의 정확성을 높이기 위하여 형상함수의 차수를 높이고 요소의 개수를 늘리게 되면, 이에 따른 수치 계산량의 급격한 증가로 인해 수치해석의 효율성은 떨어지게 된다. 이를 보완하기 위해 유한요소법에 영역분할기법을 적용하여 병렬해석을 수행하면 해의 정확성과 효율성을 동시에 높인다. 병렬해석을 수행하는데 있어서 클러스터의 구조적 특성은 해석의 효율성에 영향을 미치게 된다. 따라서 본 논문에서는 일반적인 모델에 대하여 병렬해석의 수행을 통하여 클러스터의 구조적 특성이 병렬해석의 효율성에 미치는 영향에 대해 확인한다.

keywords: 유한요소법, 영역분할기법, FETI, 클러스터

1. 서 론

유한요소법은 편미분방정식(PDE)의 수치적 근사 해를 구하기 위한 가장 효율적인 방법 중 하나로 발전해왔다. 유한요소법의 초기에는 주로 복잡한 구조물의 응력해석을 위해 사용되었으나, 지속적인 발전을 거듭하여 전반적인 공학 분야뿐만 아니라 여러 과학 및 생체학 분야의 평형, 고유치 및 전달문제의 해석분야에도 널리 사용되고 있다. 유한요소법은 연속적인 범위를 가지는 문제에서 요소는 공통되는 절점이 존재하게 되고, 요소에서의 물리적 변수들은 절점들에서의 값을 이용하여 근사화 하게 된다. 그리고 절점들 사이의 관계를 형상 함수로 정의 하여 형상함수들을 편미분방정식으로 표현하게 되면 이들을 근사화하는 연립선형방정식을 얻게 된다. 여기서 구해지는 근사 해의 정확성을 높이기 위하여 요소의 개수를 늘리면 공통되는 절점이 많아지고, 이에 따라 근사 해를 구하는 편미분방정식과 연립선형방정식의 개수가 늘어나게 되므로 해를 구하는데 있어서 계산량은 급격히 증가하게 된다. 이러한 특징으로 인하여 유한요소법과 컴퓨터의 발전은 비례하여 성장 하였지만 최근 해석하는 대상이 대형화 및 복잡해짐에 따라 수치해석의 효율성을 높이기 위한 많은 연구가 이루어져 오고 있다. 최근 단일 컴퓨터의 성능을 극대화 하는 대신 여러 대의 컴퓨터가 고속 네트워크로 연결된 클러스터 시스템을 이용해 연산의 성능을 극대화하는 방향으로 개발이 이루어짐에 따라 병렬해석과 영역분할기법에 대한 많은 연구가 이루어져 왔다.

* 정회원 · 한양대학교 건설환경공학과 교수 cepark@hayang.ac.kr

** 한양대학교 건설환경공학과 박사후연구원 pivotman@hayang.ac.kr

*** 한양대학교 건설환경공학과 박사과정 htriplehh@hayang.ac.kr

C. Farhat(1991)에 의해 처음 제안되어진 Finite Element Tearing and Interconnecting(FETI) 기법은 병렬 해석을 수행하는데 있어서 매우 효과적인 영역분할기법이다. 주어진 공간상의 도메인은 서로 겹치지 않는 여러 개의 하위도메인으로 나누어지고, 나누어진 각각의 하위도메인은 직접법 또는 간접법에 의하여 해석되어진다. 기존의 영역분할기법에서는 영역의 경계에서 하위도메인끼리의 연속성을 직접적으로 구현하였지만 FETI기법에서는 라그랑지 승수(Lagrange Multiplier)를 이용하여 간접적으로 나타내고 있다.

FETI를 이용한 병렬해석은 클러스터를 이용하여 각각의 하위도메인에 CPU를 할당하게 되고 여러 개의 CPU에서 수행되어진 후 각각 CPU의 결과 값들을 조합하게 된다. 클러스터를 이용하여 병렬해석을 진행하면 수행하는 CPU의 개수가 증가함에 따라 효율성은 증가되지만 클러스터의 구조 특성에 따라 병렬해석의 효율성은 달라질 수 있다. 따라서 본 논문에서는 유한요소법과 FETI기법을 조합한 병렬해석을 통하여 클러스터의 구조에 따른 병렬해석의 효율성을 알아보고 클러스터 구조에 가장 적합한 시스템을 제안한다.

2. 클러스터의 구조

본 수치실험에 사용되어진 클러스터는 33개 노드의 리눅스 서버로써 1개의 마스터 노드와 32개의 계산용 노드로 구성되어 있다. 각각의 노드에는 8개의 Core로 이루어져있고 총 264 Core로 구성되어있다. 또한 각각의 노드는 Message Passing Interface(MPI)를 이용한 병렬해석을 위하여 대용량의 데이터가 전송 가능하도록 QDR급 Infiniband로 연결되어 있다.

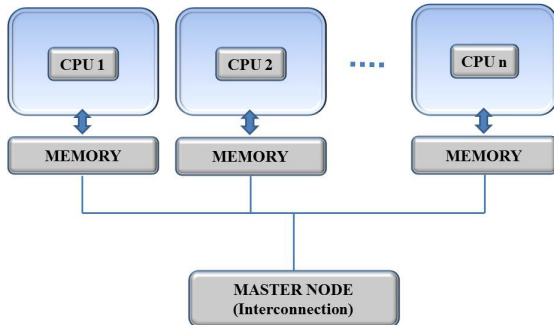


그림 1. Distributed Shared Memory Multiprocessing(DMP)

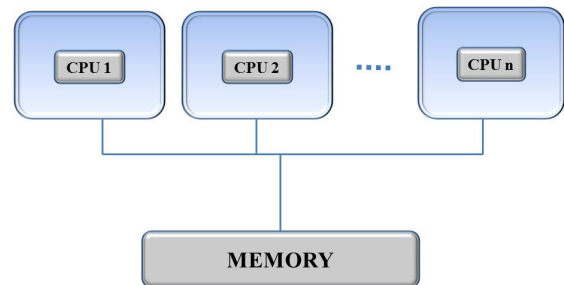


그림 2. Shared Memory Multiprocessing(SMP)

MPI 라이브러리는 점대점 통신이나 집합통신을 이용하여 메시지를 전송하는 규약이다. 점대점 통신은 하나의 CPU에서 다른 CPU로의 메시지를 전송하는 방법으로써 하나의 CPU에서 다른 CPU로의 메시지를 전송하고 다른 작업의 유무에 따라 블록킹통신(Blocking)과 논블록킹통신(Nonblocking)으로 나눌 수 있다. 블록킹통신의 경우 메시지를 전송한 후 다음 전송 메시지가 이상 없이 도착되었다는 신호를 받기 전까지는 그 CPU가 다른 작업을 할 수 없는 알고리즘이고 논블록킹통신은 전송 메시지의 도착유무가 없이도 다른 작업이 가능한 알고리즘이다. 집합통신의 경우 그림 1과 같은 여러 개의 노드들로 구성되어 있는 Distributed Shared Memory Multiprocessing(DMP)구조 또는 그림 2와 같이 여러 개의 CPU가 하나의 메모리를 공유해서 쓰는 Shared Memory Multiprocessing(SMP) 구조와 같은 시스템에 적합한 알고리즘이다.

3. SMP 및 DMP 구조적 효율성

본 수치 실험은 클러스터 구조에 따른 병렬해석의 효율성을 비교하기 위해 SMP 구조와 DMP 구조에서 각각 병렬해석이 수행된다. 먼저 1개 노드에서의 CPU를 분배한 병렬해석(SMP)과 여러 개 노드의 CPU를 각각 분배한 병렬해석(DMP)을 진행한다. 수치해석에는 그림 3과같이 가로 50cm, 세로 50cm의 2-D모델이 사용되며 요소의 개수는 각각 300개, 500개, 1000개, 2500개로 나누어진다. 또한 분할된 하위도메인의 개수는 1개부터 8개까지의 하위도메인으로 나누고, 하위도메인 당 하나의 CPU를 분배하여 수치해석을 진행한다. 이론적으로 SMP 구조는 Open Multi-Processing(OpenMP) 라이브러리, DMP 구조는 MPI 라이브러리의 사용이 이상적이다. 본 수치 실험에서는 두 가지 클러스터 구조에서 사용가능하며 병렬해석에 가장 널리 쓰이는 MPI 라이브러리가 사용된다. MPI통신을 위하여 MPICH2(2010)에서 제공하는 라이브러리와 GCC 컴파일러(2010)가 사용된다.

효율성의 비교에 앞서 MPI가 적용된 병렬해석의 수치 정확성을 확인하기 위하여 상용프로그램인 ABAQUS와의 변위가 비교된다. 지정된 노드에서 FETI 기법이 적용되었을 때의 변위와 ABAQUS를 이용하여 산출된 변위는 표 1과같이 변위의 차는 평균적으로 0.2%내외로 거의 차이가 없음을 볼 수 있다.

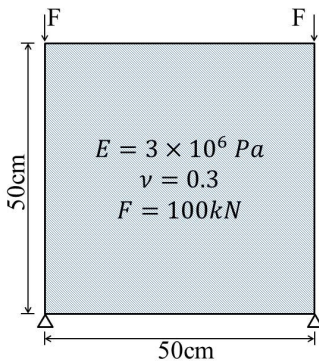


그림 3. 수치해석 모델

| node no. | ABAQUS6.10 (100 element) | Domain Decomposition Program (300 elements) |
|----------|--------------------------|---|
| 6 | 6.43719e-07 | 6.71234e-07 |
| 17 | 4.93137e-06 | 5.2377e-06 |
| 28 | 1.28026e-05 | 1.00026e-05 |
| 39 | 1.64101e-05 | 1.54921e-05 |
| 50 | 1.98156e-05 | 2.18301e-05 |
| 61 | 3.00164e-05 | 2.89244e-05 |
| 72 | 3.42351e-05 | 3.66051e-05 |
| 83 | 4.17854e-05 | 4.46514e-05 |
| 94 | 4.91358e-05 | 5.27203e-05 |
| 105 | 5.12574e-05 | 6.02585e-05 |
| 116 | 8.52328e-05 | 8.30928e-05 |

표 1. 변위비교

먼저 병렬해석의 효율성을 비교를 위해 요소에 따른 CPU 계산수행시간을 산출하였다. 그림 4에서와 같이 하위도메인의 개수가 늘어날수록 전체적인 수행시간은 줄어드는 것을 볼 수 있다. 또한 클러스터 구조에 따른 병렬해석의 효율성을 알아보기 위해 그림 5와 같이 각각의 클러스터 구조에 대한 Speed-up을 산출하였다. Speed-up은 병렬해석 알고리즘과 단일해석 알고리즘을 비교한 값으로 하나의 CPU가 사용되어 수행된 알고리즘보다 여러 개의 CPU를 사용하여 수행된 알고리즘이 얼마나 빠른지 수치적으로 나타내는 값으로 아래의 식 과 같이 정의되어진다.

$$S_p = \frac{T_1}{T_p}$$

여기서, p 는 CPU의 수, T_1 은 한 개의 CPU가 사용되었을 때의 수행 시간, T_p 는 p 개의 CPU가 사용되었을 때의 수행 시간을 나타낸다.

전체의 요소의 수가 적을 경우, SMP 구조는 DMP 구조에 비하여 약 0.1%정도로 효율성의 차이가 거의 없다. 하지만 요소의 개수가 증가하면서 계산수행시간이 길어질수록 DMP구조에서의 병렬해석이 SMP구조에

서의 병렬해석보다 약 10%정도의 효율성을 나타낸다. SMP 구조는 하나의 메모리를 공유하여 사용하고 DMP 구조는 각각의 노드에 메모리가 분배되기 때문에 계산수행속도의 차이가 난다. 또한 클러스터를 사용한 병렬해석에서 SMP 구조와 DMP 구조에서 병렬해석이 수행되어질 때 가장 이상적인 라이브러리는 서로 다르지만, 본 수치실험에서는 병렬수치해석에서 가장 널리 쓰이고 있는 MPI 라이브러리를 사용하였기 때문에 SMP 구조와 DMP 구조의 효율성에 대한 차이가 발생하게 된다.

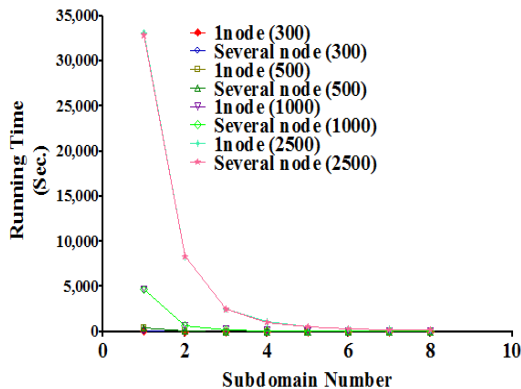


그림 4. 요소에 따른 병렬해석수행시간

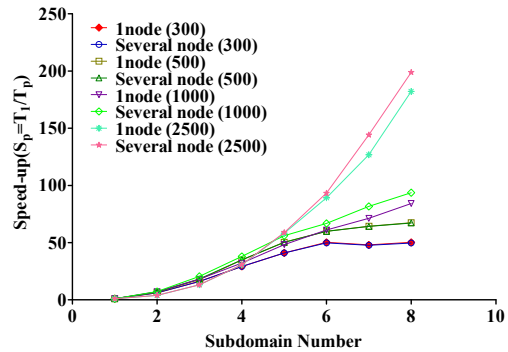


그림 5. 요소에 따른 Speed-up

4. 결론

본 논문에서는 클러스터의 구조적 특성에 따른 병렬해석의 효율성을 알아보기 위한 수치해석을 수행하였다. MPI 라이브러리를 이용한 병렬해석을 수행한 결과, 요소의 개수가 많아질수록 DMP 구조에 대한 효율성이 더욱 높게 나타났다. 복잡하고 대형화된 수치해석의 경우 SMP 구조와 DMP 구조를 동시에 사용되므로 MPI 라이브러리와 OpenMP 라이브러리를 적절히 사용하면 좀 더 높은 효율성을 얻게 된다. 또한, 보다 효과적인 병렬해석을 위해 사용되는 CPU들의 수행시간이 가능한 동일하도록 작업을 분배하여 수행대기시간을 최소화하는 로드밸런싱 기법의 적용이 요구된다. 이를 바탕으로 MPI 라이브러리, OpenMP 그리고 로드밸런싱 기법의 적절한 조합에 대한 연구가 수행되어야 한다.

감사의 글

본 연구는 한국연구재단을 통해 교육과학기술부의 세계수준의 연구중심대학 육성사업(WCU)으로부터 지원받아 수행 되었습니다(R32-2008-000-20042-0). 이에 관계자 여러분께 감사드립니다.

참고문헌

(2010)<http://www.mcs.anl.gov/research/projects/mpich2/>

(2010)<http://gcc.gnu.org/>

C. Farhat. (1991) A Method of Fintie Element Tearing and Interconnecting and Its Parallel Solution Algorithm, *International Journal for Numerical Methods in Engineering*, Vol. 32, pp.1205-1277.