1995年度 定期總會 및 秋季學術大會 論文集 1995.11.18

전기기계 설계용 병렬솔버 개발

김형수, 김형중*, 최경**, 이향범***, 정현교***, 한송엽***
* 강원대학교 제어계측공학과, ** 강원대학교 전자공학과, *** 서울대학교 전기공학과

Parallel Linear Solver under PVM for Electric Machine Design

Hyung Soo Kim, Hyoung Joong Kim*, Kyung Choi**, Hyang Beom Lee***, Hyun Kyo Jung***, Song-yop Hahn***
* Dept. of Control and Instrumentation Eng., KWNU, ** Dept. of Electronic Eng., KWNU, *** Dept. of Electrical Eng., SNU

Abstract-전기기계의 설계시에 얻어지는 대형 선형방정식을 풀기 위해서는 엄청난 메모리가 필요하고 계산시간이 요구된다는 점을 고려해야 한다. 병렬계산은 이러한 문제를 해결할 수 있다. 우리는 PVM 환경에서 병렬계산이 가능한 솔버를 설계하였다. 선형방정식을 푸는 방법으로 QMR을 사용하였으며, 수렴속도를 향상시키기 위하여 대각 프리컨디셔너를 사용하였다. 실행결과는 본 논문에서 구현한 솔버가 실제로 발생하는 문제에 대하여 아주 훌륭하게 동작하는 것을 보여준다.

1. 서 론

불균일한 물체가 놓여있는 전자레인지 공진기 내의 파동방정식을 유한요소법으로 근사화하면 매우 큰 선형방정식이 만들어진다. 이렇게 만들어진 시스템은 너무 크기 때문에 하나의 워크스테이션을 사용하여 해를 구하는 것이 불가능하다. 실제로 30만개의 비지수가 있는 시스템의 계수행렬을 저장하기 위해서는 수백 기가바이트(GByte)의 메모리가 필요하다. 계수행렬이 성긴 경우에도 수백 메가바이트(MByte)의 메모리가 필요하다. 이러한 문제를 해결하기 위해서는 가상메모리(virtual memory) 기법을 이용하면 되겠지만 연산시간을 고려할 경우 좋은 방법이라고 할 수 없다. 그래서 병렬처리가 불가피하다.

현재 병렬프로그래밍 환경을 가능하게 하는 소프트웨이 가운데 하나가 PVM(Parallel Virtual Machine)[6]이다. PVM은 이기종 컴퓨터를 네트워크로 연결하여 병렬계산이 가능하게 한다. PVM은 높은 이식성을 가지고 있으며 과학계산용에 적합한 인터페이스를 제공한다.

선형방정식의 해를 구하기 위해서 현재 가장 많이 사용하는 방법이 CG(Conjugate Gradient)[5]이다. 이러한 경우 계수행렬은 Hermitian이고 positive-definite이어야만 한다. 만일 시스템이 Hermitian이 아닐 경우에는 CG를 이용할 수 없다. 그렇지만 QMR(Quasi-Minimal Residual) 방법은 이러한 경우에도 잘 수렴한다[2,3].

우리는 전자레인지 공진기 내에 비균질 물체가 놓여있을 때 전계의 분포를 계산하기 위하여 유한요소법으로 근사화한 후 QMR 방법으로 풀었다. HP735 워크스테이션 9대를 사용하였으며 PVM 하에서 병렬로 계산이 수행될 수 있도록 프로그래밍 하였다. 이식성을 고려하여 코드는 C언어로 작성하였으며 계산시간을 줄이기 위하여 워크스테이션간에 통신이 가능한 적게 이루어지도록 설계하였다.

2. QMR 알고리듬

전자레인지 공진기 내부가 비균질 물질로 차있을 경우 파동방정식을 유한요소법으로 근사화하면 다음과 같은 시스템 방정식이 만들어진다.

$$Ax = b \tag{2.1}$$

여기서 계수행렬 A는 복소수대칭(complex symmetric)인 특성을 가지고 있다.

QMR 알고리듬은 Krylov 부공간(subspace) 방법으로서 일반적으로 역행렬이 존재하고 Hermitian이 아닌 선형방정식을 풀기위하여 사용한다. QMR 방법은 Lanczos 처리와 준최소나머지(quasi-minimal residual) 조건의 두가지로 구성된다. Lanczos 알고리듬은 Krylov 부공간에 기초한 기저(basis)벡터를 생성하기 위하여 사용한다. 그리고 룩어헤드(look-ahead) 기법은 고전적인 Lanczos 알고리듬에서 발생하는 계산중단을 피할 수 있다. Lanczos 기저벡터가 만들어지면 준최소 나머지 특성은 Krylov 부공간으로부터 QMR 반복을 선택하기 위하여 사용된다. QMR 반복은 항상 잘 정의되어 있으며, 부드럽고 거의 일정하게 수렴하는 특성을 가지고 있다.

최초의 QMR 알고리듬은 Krylov 부 공간의 기저벡터를 생성하기 위하여 3항순환(three-term recurrence)을 사용하였다. 그러나 일반적으로 3항순환에 기초한 벡터의 반복은 수학적으로 등가인 결합 2항순환(coupled two-term recurrence)보다 강

건하지 않다[4].

QMR 알고리듬은 V-W 시퀀스와 P-Q 시퀀스를 반복한다. 이때 w_1은 임으로 선택하였다. 그러나 복소수 대칭인 경우 이것을 v_1과 같게 선택하면, 모든 n에 대하여 $w_n = v_n$, 그리고 $q_n = p_n$이 된다. 그러므로 순환벡터 w_n과 q_n은 생략될 수 있고, 따라서 연산시간과 저장공간이 절약될 수 있다. 여기서 $M = M_1M_2$이며 복소수 대칭인 시스템인 경우 M 역시 대칭이어야 한다. 즉 다음을 만족해야 한다.

$$M = M_1M_2 = (M_1M_2)^T = M^T. \tag{2.2}$$

다음은 복소수대칭인 시스템에 대한 QMR 알고리듬이다.

알고리듬 2.1 복소수대칭인 시스템에 대한 QMR 알고리듬

0. Choose $x_0 \in \mathbf{C}^N$ and set $r_0 = b - Ax_0$.
   Compute $\rho_1 = \|M_1^{-1}r_0\|$ and set $v_1 = r_0/\rho_1$.
   Set $p_0 = d_0 = 0,\ c_0 = \varepsilon_0 = 1,\ \theta_0 = 0,\ \eta_0 = -1$.

For $n = 1, 2, \cdots$ do:
1. If $\varepsilon_{n-1} = 0$ then stop.
   Compute $\delta_n = v_n^T M^{-1} v_n$. If $\delta_n = 0$, then stop.
2. Compute
   $p_n = M^{-1}v_n - p_{n-1}(\rho_n\delta_n/\varepsilon_n)$.
3. Compute $\varepsilon_n = p_n^T A p_n$, $\beta_n = \varepsilon_n/\delta_n$, and
   $\tilde{v} = Ap_n - v_n\beta_n$, $\rho_{n+1} = \|M_1^{-1}\ \tilde{v}_{n+1}\|$.
4. Compute
   $$\theta_n = \frac{\omega_{n+1}\rho_{n+1}}{\omega_n c_{n-1}|\beta_n|},\ c_n = \frac{1}{\sqrt{1+\theta_n^2}},\ \eta_n = -\eta_{n-1}\frac{\rho_n c_n^2}{\beta_n c_{n-1}^2}.$$
   $d_n = p_n\eta_n + d_{n-1}(\theta_{n-1}c_n)^2$, $x_n = x_{n-1} + d_n$.
5. If $\rho_{n+1} = 0$, then stop.
   Otherwise, set
   $v_{n+1} = \tilde{v}_{n+1}/\rho_{n+1}$.

3. 계산환경

PVM은 소프트웨어 시스템으로 이기종 유닉스 컴퓨터들을 네트워크로 연결하여 하나의 대형 병렬컴퓨터처럼 사용할 수 있게 만들어준다. 따라서 성능이 다소 떨어지는 컴퓨터라도 다수를 통합시켜 성능을 향상시키는 방법으로 대형 계산문제를 고속으로 처리할 수 있다.

우리는 그림 1과 같이 네트워크로 연결되어 있는 9대의 HP-735 워크스테이션에 PVM을 사용하여 병렬처리를 가능하게 하였다. 하나의 워크스테이션은 서버로 사용하며, 이것은 초기의 데이터 분배 및 모니터용으로 사용한다. 나머지 8대의 워크스테

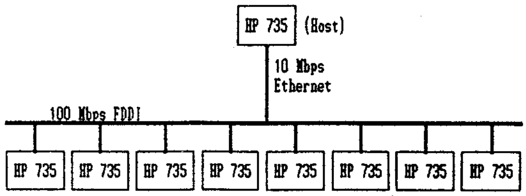


그림 1 HP 735 클러스터 컴퓨팅 환경

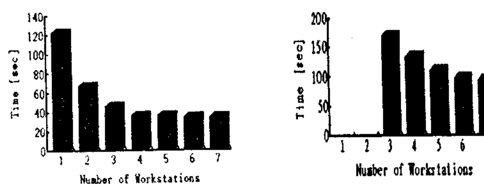
이션은 100 Mbps FDDI로 연결되어 있으며, 각각은 서버와 10 Mbps 이더넷으로 연결되어 있다.

4. 구 현

우리는 메모리부족 문제 및 계산시간의 단축이라는 두가지 목표를 가지고 솔버(solver)를 구현하였다. 먼저 메모리부족 문제를 해결하기 위하여 식 (2.1)과 같은 선형방정식의 A와 b를 각각의 프로세서에 동일한 크기로 분배하였다. 이렇게 하였을 경우 솔버에서 고려하여야 할 사항은 벡터노옴(norm) 및 내적(inner product)의 계산, 그리고 가장 계산시간이 많이 요구되는 행렬-벡터곱(matrix-vector product)의 계산과정이다. 이것을 수행하기 위해서는 특별히 각 프로세서간에 통신의 과정이 필요하다. 여기서 통신이 너무 빈번하게 이루어질 경우 계산시간에 영향을 미치게 된다. 또한 수렴속도를 향상시키기 위하여 프리컨디셔너(preconditioner)[1]를 사용하였다. 그러나 병렬프로그래밍에서 프리컨디셔너를 구현하기란 쉽지 않다. 그래서 우리는 대각프리컨디셔너를 사용하였다.

5. 실행결과

그림 2는 워크스테이션의 수에 따른 연산시간을 비교한 것이다. 비교에 사용된 행렬은 토플리츠행렬(Toeplitz matrix)로서 크기는 각각 1000×1000, 2000×2000이다. 각각의 경우 오차노옴이 1e-10이 될 때까지 계산한 결과이며 반복횟수는 두 경우 모두 93회이다. 여기서 보면 워크스테이션의 수가 많아지면 계산시간이 감소하는 것을 알 수 있다. 또한 미지수의 개수가 많아지면 그 차이는 더욱 명백해 진다.



(a) 1000×1000 (b) 2000×2000

그림 2 워크스테이션 수에 따른 계산시간 비교

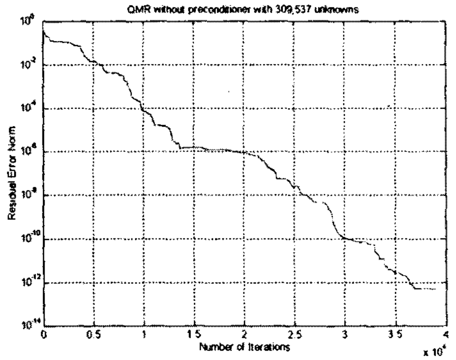


그림 3 QMR 방법의 수렴특성

다음은 실제 전자레인지 내부를 유한요소법으로 근사화하여 풀어본 결과이다. 미지수의 개수는 309,537개 이며 복소수 대칭인 특징을 가지고 있다. 그림 3은 부드럽게 감소하는 QMR의 특성을 잘 보여준다.

그림 4는 프리컨디셔너의 효과를 잘 보여준다. 여기서 푼 문제는 35,966개의 미지수가 있는 시스템으로서 역시 복소수 대칭인 특성을 가지고 있다. 오차노옴이 1e-12으로 떨어지는 데 프리컨디셔너를 사용하지 않은 경우와 비교하여 약 2,000번의 반복횟수가 감소한다.

6. 결 론

대형 선형방정식을 풀기 위해서는 메모리 부족 문제의 해결과 계산시간의 단축이 당면과제이다. 본 연구에서는 실제 전기기계의 설계시에 발생하는 문제를 QMR 알고리듬으로 풀었다. CG가 Hermitian인 시스템인 경우에만 제대로 동작하는 데 비하여 QMR은 모든 시스템에 대하여 잘 수렴하는 특성을 가지고 있다. 또한 PVM은 선형방정식이 대형이기 때문에 하나의 컴퓨터에서 풀 수 없는 문제에 대하여 이것을 해결할 수 있는 훌륭한 병렬계산 환경을 제공한다. 그리고 대각프리컨디셔너를 사용하였을 때 계산시간이 많이 감소한다는 것을 알 수 있다.

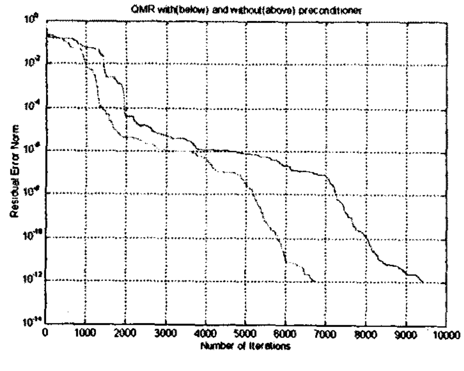


그림 4 대각프리컨디셔너의 효과

선형방정식의 해를 구하기 위해서는 시스템의 특성이 매우 중요하다. 비록 CG가 우리가 풀이본 문제에 대해서는 잘 동작하지 않지만 Hermitian인 경우에는 잘 동작한다. 따라서 Hermitian인 시스템은 CG로, 그렇지 않은 경우 QMR로 푸는 것이 현재 선형방정식의 해를 구하는 좋은 방법이다.

우리가 구현한 솔버는 초기 데이터 분배시 모든 프로세서에 동일한 양의 자료를 저장했다. 그러나 계산시간을 단축시키기 위해서는 프로세서의 성능을 고려하여 최적으로 배분하는 방법을 찾아야 할 것이다. 또한 수렴속도를 향상시키기 위한 병렬 프리컨디셔너도 향후 연구되어야 할 부분이다.

후 기

본 연구는 기초전력공학공동연구소(관리번호 94-001)와 석재복합신소재연구센터(관리번호 95-03-03)의 지원에 의하여 이루어졌음.

참고문헌

[1] R. Barrett, M. Berry, T. Chan, J. Demmel, J. Donato, J. Dongarra, V. Eijkhout, R. Pozo, C. Romine, H. van der Vorst, *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods*, SIAM, 1994.

[2] R. W. Freund, "Conjugate Gradient-type Methods for Linear Systems with Complex Symmetric Coefficient Matrices," *SIAM Journal on Scientific Computing*, vol. 13, no. 1, pp. 425-448, January, 1992.

[3] R. W. Freund and N. M. Nachtigal, "QMR: A Quasi-Minimal Residual Method for Non-Hermitian Linear Systems," *Numerische Mathematik*, vol. 60, pp. 315-339, 1991.

[4] R. W. Freund and N. M. Nachtigal, "An Implemeatation of the QMR Method based on Coupled Two-term Recurrence," *SIAM Journal on Scientific Computing*, vol. 15, no. 2, pp. 313-337, March, 1994.

[5] G. Golub and C. F. Van Loan, *Matrix Computations*, The Johns Hopkins University Press, 1989.

[6] V. Sunderam, "PVM: A Framework for Parallel Distributed Computing," *Concurrency: Practice and Experience*, vol. 2, pp. 315-339, 1990.