

## 평면골조의 최적설계를 위한 병렬 O.C. 알고리즘 Parallel O.C. Algorithm for Optimal design of Plane Frame Structures

김철용\*  
Kim, Cheol Yong

박효선\*\*  
Park, Hyo Seon

박성무\*\*\*  
Park, Sung Moo

### ABSTRACT

Optimality Criteria algorithm based on the derivation of reciprocal approximations has been applied to structural optimization of large-scale structures. However, required computational cost for the serial analysis algorithm of large-scale structures consisting of a large number of degrees of freedom and members is too high to be adopted in the solution process of O.C. algorithm. Thus, parallel version of O.C. algorithm on the network of personal computers is presented in this paper. Parallelism in O.C. algorithm may be classified into two regions such as analysis and optimizer part. As the first step of development of parallel algorithm, parallel structural analysis algorithm is developed and used in O.C. algorithm. The algorithm is applied to optimal design of a 54-story plane frame structure.

### 1. 서론

고성능 연산기를 포함한 전산관련 기술의 급속한 발전은 구조공학분야에 다양한 형식의 영향을 미치고 있다. 대표적인 예로 슈퍼컴퓨터 또는 대형병렬컴퓨터 등을 이용한 병렬연산 또는 분산연산기법의 구조공학분야에의 적용을 들 수 있다. 특히 구조해석 및 설계분야에서는 기존의 직렬구조해석 및 설계알고리즘이 병렬 또는 분산알고리즘의 형태로 개발되어 연구 및 실무에 적용되고 있다. 그리고 기존 구조해석 및 설계용 소프트웨어 또한 병렬화하고 있다. 이러한 경향은 반복적 구조해석을 기본으로 하는 구조설계의 경우 더욱 적용성이 높은 것으로 평가된다. 그러므로 최근 들어 구조설계 자동화의 기본적 구성요소인 구조최적화분야에 병렬 및 분산연산 기법이 활발하게 도입되고 있다.<sup>1)</sup> 이러한 병렬최적화기법은 자유도 및 부재의 수가 많은 초고층 구조물을 포함한 대형구조물의 효율적 설계를 위하여 개발되어야 한다.

구조최적화알고리즘은 다양한 형식으로 분류할 수 있으나 최적해를 탐색하는 과정에 요구되는 구조물의 반응 및 구조반응에 대한 민감도계수의 활용 유무에 따라 두가지로 분류할 수 있다. 최적화과정에 민감도계수가 이용되지 않은 유전자알고리즘, Simulated Annealing Algorithm, 인공신경망기본 최적화기법 등은 최적화 알고리즘의 안정성 그리고 전역적최소해의 탐색 가능성 등 여러 가지 장점을 가지고 있으나 대부분의 경우 많은 수의 반복적 구조해석을 요구하는 단점을 가지고 있다. 그리고 민감도계수를 이용하는 구조최적화 알고리즘은 중 대형구조물의 최적화에 적합한 기법은 Optimality Criteria (O.C.) 알고리즘으로 대표된다.

그러므로 본 논문에서는 주위에서 쉽게 접할 수 있는 개인용 컴퓨터를 이용하여 가상의 병렬 연산 시스템

\* 영남대학교 건축공학과 석사과정  
\*\* 연세대학교 사회환경·건축공학부 조교수  
\*\*\* 영남대학교 건축공학과 교수

을 구성하여 고성능 연산기의 성능을 구현하고 이에 적합한 병렬 O.C. 알고리즘을 개발하고자 한다. 가상의 병렬 연산 시스템이란 저가로 쉽게 구입할 수 있는 다수의 개인용 컴퓨터를 근거리 네트워크(LAN)로 연결하여 하나의 시스템을 구성하는 것이며, 각 컴퓨터간 통신을 통하여 데이터를 주고 받는 Message Passing<sup>2)</sup> 방식을 이용한다. 이렇게 구성된 가상의 병렬 컴퓨터를 이용한 골조의 최적설계를 위한 알고리즘 병렬화는 크게 두 부분으로 나눌 수 있다. 해석부분의 병렬화와 최적화 부분의 병렬화가 그것이다. 본 논문은 병렬최적화 알고리즘 개발을 위한 1단계 작업으로써, 기존에 개발된 병렬 해석 알고리즘으로 해석 부분만을 병렬하여 최적화 과정을 수행하며, 향후 2단계 수치 해석적인 방법을 이용하는 Optimality Criteria의 병렬화를 위한 준비작업으로 삼고자 한다.

## 2. 문제의 정식화<sup>3),4)5)</sup>

최적화 문제는 구조물의 중량을 최소화시키는 문제로 정식화되며, 제약조건은 부재에 걸리는 응력과 각 층간변위 그리고 부재의 최소크기를 제약하게 된다. 설계변수는 단면적이 연속형 변수로 사용되었다. 문제를 수학적으로 정식화하여 표현하면 다음과 같다.

$$\text{Minimize : } W = \sum_{i=1}^m \rho_i L_i a_i \quad (1)$$

$$\text{Subject to : } (\delta_j - \delta_{j-1})/h_j \leq D_j^U \quad (j=1, 2, 3, \dots, ns) \quad ns = \text{층수} \quad (2)$$

$$\frac{\sigma_i}{\bar{\sigma}} \leq 1 \quad (i = 1, 2, 3, \dots, m) \quad m = \text{부재수} \quad (3)$$

$$a_i \leq a_i \quad (4)$$

여기서  $\delta_j$ 는 j층의 층변위,  $h_j$ 는 j층의 층고,  $\bar{\sigma}$  = 허용응력,  $a_i$  = 최소단면, 그리고  $D_j^U$ 는 허용 층간 변위 계수(story drift index)를 각각 나타낸다.

수치적인 접근으로 도함수를 규정하기 위해 가상일의 원리를 적용한다. 이때 j층의 변위  $\delta_j$ 는 다음의 식으로 표현 될 수 있다.

$$\delta_j = \sum_{i=1}^m \int_0^{L_i} \frac{M_i(x) m_i^j(x)}{E I_i} dx + \sum_{i=1}^m \frac{F_i f_i^j L_i}{E a_i} \quad (5)$$

$M_i$ 는 실제하중에 의해 i 부재에 발생하는 모멘트이고,  $m_i^j$ 는 j 층에 가상하중이 작용할 때 i 부재에 발생하는 모멘트,  $F_i$ 는 실제하중에 의해 i 부재에 발생하는 축력,  $f_i^j$ 는 j 층에 가상하중이 작용할 때 i 부재에 발생하는 축력,  $I_i$  = i 부재의 단면 2차 모멘트,  $E$ 는 탄성계수,  $m$ 은 부재,  $L_i$ 는 i 부재의 길이를 각각 나타낸다.

$C_{ij}$ 와  $C_{ij}'$ 를 다음과 같이 두면,

$$C_{ij} = \int_0^{L_i} \frac{M_i(x) m_i^j(x)}{E} dx, \quad C_{ij}' = \frac{F_i f_i^j L_i}{E} \quad (6)$$

j층에서의 변위는 식(7)과 같이 다시 쓸수 있으며,

$$\delta_j = \sum_{i=0}^m \left( \frac{C_{ij}}{I_i} + \frac{C_{ij}'}{a_i} \right) \quad (7)$$

(7)식으로부터 구해진 변위로부터 층간변위계수(story drift index)  $D_j$ 는

$$D_j = (\delta_j - \delta_{j-1}) / h_j = \sum_{i=1}^m \left( \frac{C_{ij}}{I_i} + \frac{C_{ij}'}{a_i} \right) = \sum_{i=1}^m d_{ij} \quad (8)$$

$$\text{여기서 } d_{ij} = \frac{C_{ij}}{I_i} + \frac{C_{ij}'}{a_i} \quad (9)$$

$$C_{ij} = \{(C_{ij} - C_{i,j-1})\} / h_j \quad (10)$$

$$C_{ij}' = \{(C_{ij}' - C_{i,j-1}')\} / h_j \quad (11)$$

$C_{ij}$ 와  $C_{ij}'$ 는 한번 해석에 재설계 과정동안 변하지 않는 것으로 가정하고, 1회 재설계가 끝나서 다음 해석을 하게 되면 바뀌는 것으로 한다. (8)에 의해 구해진 식으로부터 (2)의 변위에 대한 제약식을 다음과 같이 바꾸어 쓸 수 있다.

$$\text{Subject to } \sum_{i=1}^m d_{ij} \leq D_j^U \quad (12)$$

### 3. 단면 재설계

#### 3.1 응력 제약

응력제약은 허용응력설계법을 적용하였으며, 부재에 좌굴은 없는 것으로 가정하여 축력과 모멘트에 의한 응력을 제약하였다. 응력 제약은 응력비에 의한 방법을 이용하여 구한다. 부재 내부에 발생하는 축력과 모멘트의 최대값을 각각  $F_i^{\max}$ 와  $M_i^{\max}$ 로 표현하게 되면 이때  $i$ 부재내에 발생하는 최대응력  $\sigma_i^{\max}$ 는

$$\sigma_i^{\max} = \frac{F_i^{\max}}{a_i} + \frac{M_i^{\max}}{z_i} \text{ 이 되며} \quad (13)$$

응력제약에 대한 새로운 단면적  $a_i^{nu+1}$ 를 위한 반복 과정은 (14)와 같이 된다.

$$a_i^{nu+1} = a_i^{nu} \left( \frac{\sigma_i^{\max}}{\bar{\sigma}_i} \right)^\eta \quad (14)$$

$\bar{\sigma}_i$ 는  $i$  번째 부재의 허용응력이며  $\eta$ 는 수렴성을 조절하는 매개 변수이다.  $nu$ 는 재설계를 위한 반복수를 나타낸다.

#### 3.2 변위제약

구조최적화 문제는 Lagrangian 함수를 최소화시키는 문제로 다시 쓸 수 있다.

$$L(a_i, \lambda_j) = \sum_{i=1}^m w_i a_i + \sum_{j=1}^m \lambda_j [\sum_{i=1}^m d_{ij} - D_j^U] \quad (\text{put } w_i = \rho_i I_i) \quad (15)$$

Lagrangian 함수를 최소화시키기 위해 (15)식을 단면적으로 미분하여 다음과 같이 얻을 수 있다.

$$\frac{\partial L}{\partial a_i} = w_i + \sum_{j=1}^m \lambda_j \frac{\partial d_{ij}}{\partial a_i} = 0 \quad (16)$$

여기서  $\frac{\partial d_{ij}}{\partial a_i} = -d_{ij}'$ 로 두고 정리하면 윗식은 (17)과 같다.

$$\frac{\sum_{j=1}^m \lambda_j d_{ij}'}{w_i} = 1 \quad - \text{(Optimality Criteria)} \quad (17)$$

여기서 Lagrange Multiplier  $\lambda_j$ 는 j 층의 최적값에 대한 중요도를 측정하는 가중치로써 여겨지며 실제변수가 수렴하기 위해서는 (17)의 조건을 만족해야 함을 뜻한다.

새로운 단면을 찾기 위해서는 Lagrange Multiplier를 먼저 찾아야 하는데 이를 위해서 단면적 변화에 따른 층간변위계수(story drift index)의 변화량을 구해보면,

$$\Delta D_k = \sum_{i=1}^{m-\xi} \left( \frac{\partial d_{ik}}{\partial a_i} \Delta a_i \right) \quad (m-\xi = \text{활성제약변수}) \quad (18)$$

$$\text{put } \Delta D_k = D_k^{nu+1} - D_k^{nu} \quad (nu=\text{재설계 반복수}) \quad (19)$$

$$\sum_{i=1}^{m-\xi} \frac{\partial d_{ik}}{\partial a_i} = \sum_{i=1}^{m-\xi} [-d_{ik}'] \quad (k=\text{층간변위제약}) \quad (20)$$

$$\Delta a_i = a_i^{nu+1} - a_i^{nu} = \frac{a_i^{nu}}{\eta} \left( \sum_{j=1}^m \frac{\lambda_j d_{ij}'}{w_i} - 1 \right)_{nu} \quad (\eta=\text{step size}) \quad (21)$$

(19), (20), (21)식을 (18)식에 대입하여 정리하면,

$$(D_k^{nu+1} - D_k^{nu}) = \sum_{i=1}^{m-\xi} [-d_{ik}'] \frac{a_i^{nu}}{\eta} \left( \sum_{j=1}^m \frac{\lambda_j d_{ij}'}{w_i} - 1 \right) \quad (22)$$

이때 단면적이 nu+1번째 수렴한다고 가정하면,

$$(D_k^U - D_k^{nu}) = \sum_{i=1}^{m-\xi} [-d_{ik}'] \frac{a_i^{nu}}{\eta} \left( \sum_{j=1}^m \frac{\lambda_j d_{ij}'}{w_i} - 1 \right) \quad (23)$$

$$\sum_{j=1}^{ns} \lambda_j \sum_{i=1}^{m-f} \frac{a_i^{nu} d_{ik}' d_{ij}'}{w_i} = \sum_{i=0}^{m-f} a_i^{nu} d_{ik}' - \eta(D_k^U - D_k^{nu}) \quad (24)$$

(24)식을 이용하여  $\lambda_j$ (Lagrange Multiplier)를 구할 수 있다.  $\lambda_j$ 를 구하기 위한 Gauss Seidal반복법은 다음의 식(25)같다.

$$\lambda_k^{ladr+1} = \frac{1}{e_{kk}} (b_k - \sum_{j=1}^{k-1} e_{kj} \lambda_j^{ladr+1} - \sum_{j=k+1}^{ns} e_{kj} \lambda_j^{ladr}) \quad (k=1, 2, \dots, ns) \quad (25)$$

$$\text{여기서 } e_{kk} = \sum_{i=1}^{m-f} \left( \frac{a_i^{nu} d_{ik}' d_{ik}'}{w_i} \right)_{kk} \quad (26)$$

$$e_{kj} = \sum_{i=1}^{m-f} \left( \frac{a_i^{nu} d_{ik}' d_{ij}'}{w_i} \right)_{kj} \quad (27)$$

$$b_k = \sum_{i=0}^{m-f} a_i^{nu} d_{ik}' - \eta(D_k^U - D_k^{nu}) \quad (28)$$

위의 최적성 기준(17)을 이용한 변위 제약에 대한 반복과정은 다음과 같다.

$$a_i^{nu+1} = a_i^{nu} \left\{ \sum_{j=0}^{m-f} \frac{\lambda_j d_{ij}'}{w_i} \right\}^{\frac{1}{\eta}} \quad (29)$$

이항정리를 써서 정리하면,

$$\begin{aligned} a_i^{nu+1} &= a_i^{nu} \left[ 1 + \frac{1}{\eta} \left( \sum_{j=1}^{m-f} \frac{\lambda_j d_{ij}'}{w_i} - 1 \right) \right]_{nu} \\ &= a_i^{nu} + \frac{a_i^{nu}}{\eta} \left( \sum_{j=1}^{m-f} \frac{\lambda_j d_{ij}'}{w_i} - 1 \right)_{nu} \end{aligned} \quad (30)$$

여기서  $\eta$  는 step size parameter로 설계변수의 수렴성을 조절한다.

#### 4. 가상 병렬 연산 시스템의 구현<sup>(7,8)</sup>

본 논문에서 구현하고자 하는 분산 연산시스템 모델은 Windows98 환경의 개인용 컴퓨터들을 Ethernet Network Card로 연결하여 네트워크를 구성하며, 필요한 데이터는 통신을 통하여 주고받는 Message Passing방식을 적용하고 이때 통신은 WMPI(Message Passing Interface for Win32 Platforms) 패키지를 이용한다.

시스템의 구성은 한 대의 Master 컴퓨터를 중심으로 다수의 Slave 컴퓨터들이 Master 컴퓨터에 연결되는 구성을 가진다. 주 연산은 Slave 컴퓨터들이 Master에 의해 할당된 양만큼 수행하게 되며, 이때 기본적으로 각 Slave들의 연산량은 모두 동일하며 계산시간도 역시 동일하게 되나, 부재의 그룹핑으로 인하여 분담량이 달라질 수도 있다. Master 컴퓨터는 각 Slave와의 데이터 송,수신 및 데이터의 입출력을 담당하게 된다.

#### 5. 병렬 O.C알고리즘

그림 5-1은 기존의 O.C 알고리즘의 흐름도를 나타낸 것이고, 그림 5-2는 본 연구에서 제시하고자

하는 O.C 알고리즘의 흐름도를 나타낸 것이다.

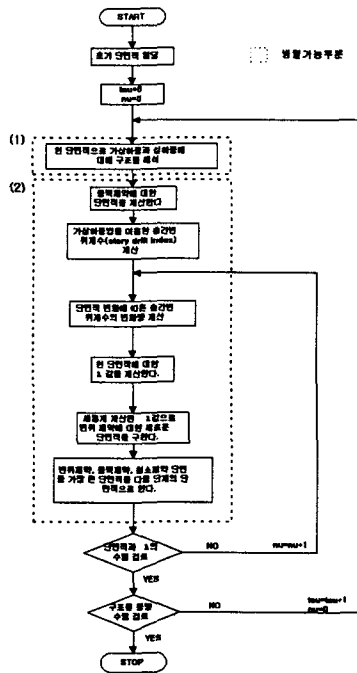


그림 5-1 직렬 O.C 알고리즘

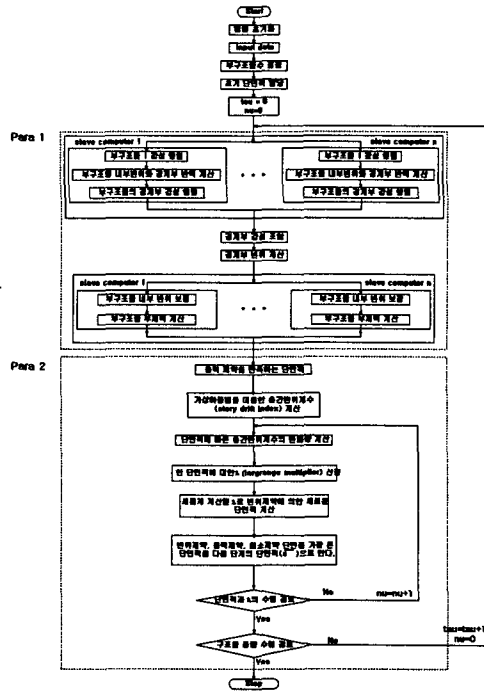


그림 5-2 병렬 O.C 알고리즘

그림 5-1의 (1)과 (2)는 병렬알고리즘으로 바꾸고자 하는 부분을 표시한 것이며, (1)부분은 이미 개발되어 있는 부구조 기법을 이용하여 병렬화한 알고리즘으로 그림 5-2의 Para1으로 표시되고 앞으로 개발되어야 할 부분으로 Para2라고 표시를 하였다.

그림 5-2의 Para1의 흐름을 간략히 설명하면 다음과 같다.<sup>7),8)</sup>

(1) 나누어진 각 부구조물에 대해 강성매트릭스 구성(Slave)

(2) 각 부구조물의 경계부 강성 추출(Slave)

$$K_b^{(n)} = K_{bb}^{(n)} - K_{ba}^{(n)}(K_{aa}^{(n)})^{-1}K_{ab}^{(n)}$$

(3) 각 부구조물에 대한 해석 수행(Slave)

$$[U_i^{(n)}]_{\text{경계부구조}} = K_{ba}^{(n)}(K_{aa}^{(n)})^{-1}P_i^{(n)}$$

$$R_b^{(n)} = K_{ba}^{(n)}(K_{aa}^{(n)})^{-1}P_i^{(n)}$$

(5) 추출된 경계부 강성 매트릭스를 이용하여 경계부 해석 수행 (Master)

$$K_b U_b = Q_b$$

$$Q_b = - \sum_{i=1}^n R_b^{(i)} + P_b$$

$$U_b = K_b^{-1}Q_b$$

(6) 부구조물 내부 변위 수정(Slave)

$$U_i^{(n)} = -(K_{aa}^{(n)})^{-1}K_{ab}U_b + U_i^{(n)}$$

(7) 계산된 변위를 이용하여 부재력을 계산(Slave)

## 6. 예제 적용 및 결과

개발된 1단계 병렬 최적화 알고리즘을 3경간 54층 평면 골조(그림 6-1)에 적용하였다. 예제는 경간은 1000.0 cm, 층고 400.0 cm로 이루어진 구조물이다. 이 예제를 통하여 순수 부구조 기법을 적용한 O.C알고리즘과 해석 부분만을 병렬화시킨 O.C알고리즘의 speedup을 비교해 보고, 최적화 routine의 병렬화의 당위성과 향후 나아갈 방향을 제시해 보고자 한다.

입면상 대칭되는 기둥, 보를 3개층씩 그룹핑하였으며, 하중조건으로는 KS규준에 나오는 풍력 산정법으로 산정하여 각 절점하중으로 작용 시켰고, 각 보에는 50 kg/cm의 등분포하중을 작용시켰으며 구조물 자체의 중량변화는 무시하였다.

용력제약은 부재의 좌굴은 무시하고 축력과 모멘트에 대해 제약을 가하였고, 각 층간변위계수(story drift index)는 사용성 한계를 고려하여 1/400로 제약하였다. 초기 단면은 모든 부재를 700 cm<sup>2</sup>로  $\eta$ 는 3으로 두고 중량변화가 전체 중량의 0.001% 이내일 때 수렴한 것으로 하여 모든 최적화 과정을 종료하였다. 전체 중량은 32번 해석만에 수렴하였으며, 그 이력은 다음의 그림 (6-2)과 같다.

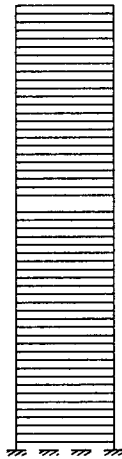


그림 6-1 3경간54층

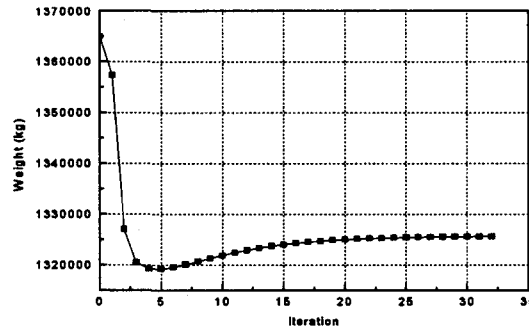


그림 6-2 중량 수렴 곡선

병렬 알고리즘과 직렬 알고리즘 모두 같은 반복과 수렴 이력을 가지며, 이는 병렬화가 해에 미치는 영향이 없음을 알 수 있다. 그에 반해 각 프로세서수에 따른 연산 시간의 변화를 보면 그림 6-3에서 알 수 있듯이 Slave가 증가할수록 선형적으로 연산시간이 감소하는 것이 아니라 Slave가 6개일 때 연산시간이 최소이고 그 다음부터는 오히려 연산시간이 증가하는 것을 알 수 있다. 이는 통신에 필요한 시간이 전체 시간에 영향을 미치기 때문이다. 이는 병렬에서 통신횟수와 연산량의 관계가 적절해야만 최대의 효율을 얻을 수 있음을 뜻한다.

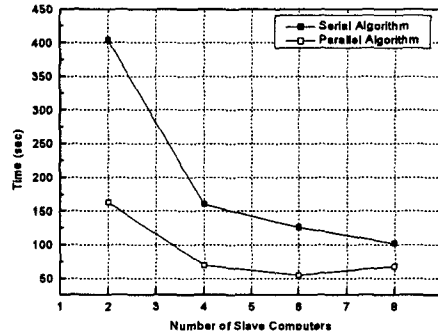


그림 6-3 total elapsed time

## 7. 결론

본 연구는 O.C. 병렬 최적화를 위한 1단계 작업으로 해석 부분만을 병렬화 하였으나, 최적화 과정에 소요되는 연산량의 대부분이 구조해석 부분에 집중되어있으므로 전체적인 연산시간을 많이 향상 시킬 수 있었다. 그리고 자유도의 수가 많은 구조물의 경우 전체 구조최적화 과정에 소요되는 연산 중 해석이 차지하는 비중이 높아지므로 그 효율성 또한 개선될 것으로 판단된다. 그러나 통신에 따른 시간 소비로 인하여 이상적인 선형감소는 나타나지 않았으며 네트워크의 통신성능이 낮은 Ethernet의 경우 오히려 Slave 수가 지나치게 많을 경우 직렬 알고리즘 보다 성능이 떨어질 수 있으므로 보다 통신량을 줄일 수 있는 알고리즘을 개발해야 한다.

## 8. 참고문헌

1. Hyo Seon Park, Hojjat Adeli, "Distributed Neural Dynamics Algorithms for Optimization of Large Steel Structures", *Journal of Structural Engineering*, ASCE, Vol. 123, No. 7, pp. 880-888, July, 1997
2. Marc Sner 외 4, MPI : The Complete Reference, The MIT Press, 1996
3. 정진현, "최적기준에 의한 철골구조물의 최적설계", 울산대학교 대학원 토목, 환경공학부, 석사학위논문, 1997
4. Chun-Man Chan, Donald E. Grierson, "An Efficient Resizing Technique for The Design of Tall Steel Buildings Subject to Multiple drift Constraints", *The Structural Design of Tall Buildings*, v2, 17-32, 1993
5. Chun-Man Chan, Donald E. Grierson, Archibald N. Sherbourne, "Automatic Optimal Design of Tall Steel, Building Frameworks", *Journal of Structural Engineering*, 838-847, May, 1995
6. H. Adeli and O. Kamal, *Parallel Processing in Structural Engineering*, Elsevier Applied Science, 1993
7. 박효선, "대형 구조물을 위한 병렬 구조해석 및 설계", *전산구조공학회지*, 제9권, 제3호, pp 47-53, 1996년 9월
8. 박효선, 박성무, 성창원, 김재홍, "부구조기법을 이용한 PC level 분산구조해석법", *한국전산구조공학회 발표논문집*, 제 11권, 제 2집, pp. 53-60, 1998년 10월