Comparison of Preconditioned Conjugate Gradient Methods for Adaptive Finite Element Analysis

유한요소 적응분할해석을 위한 선조정 공액구배법들의 비교연구

朱 寬 正*
Joo, Kuan Jung

요 약

유한요소 적응분할 해석을 행할경우 강도매트릭스의 요소배열 형태는 밴드 현상이 아닌 성긴 현상을 갖게된다. 그러한 성긴 현상의 평형방정식들을 풀기 위해서는 컴퓨터 주기억 장소의 가용량이 중요한 관건이 된다. 따라서, 주기억 장소의 사용량을 최소로 줄이고 수렴속도가 높은 반복법에 의한 해를 구하는 알고리즘이 요구된다. 본 논문에서는 '불완전 Cholesky 분해'를 이용한 선조정 공액구배법을 다른 종류의 선조정 구배법들과 비교, 연구해 본다.

Abstract

Adaptive refinements yield a large sparse system of equations. In order to solve such a system, the core storage requirement is an important consideration. Accordingly, an iterative method which minimizes the core storage and provides a high rate of convergence is called for. In this paper the conjugate gradient algorithms with various preconditionings including the incomplete Cholesky decomposition are examined.

1. INTRODUCTION

At each level of refinement during the adaptive finite element analysis the equilibrium equations lead to a linear system.

$$Ax=b \qquad (1)$$

where A is a $n\times n$ symmetric, nonsingular, large and sparse stiffness matrix, and b is a $n\times 1$ load vector. In general, solution strategies of Eq.(1) fall into two classes: direct methods and iterative methods. The solution strategy to be employed depends largely on the type of problem to be solved.

In the problem considered here, the sparsity of the system causes the direct method to suffer from fill-in when A is factored. Even using typical adaptation strategies to exploit sparsity by the use of orderings, permutations, and special pivotings that minimize the fill-in [1, 2], the direct method increases the number of operations dramatically

* 정회원, 삼성항공 산업주식회사 CAD/CAM실, 공학박사

□ 이논문에 대한 토론은 1989년 3월31일까지 본학회에 보내주시면, 그결과를 1989년 9월호에 게재하겠읍니다.

as the size of the system grows larger. Moreover a large system of equations cannot be solved within a limited core memory. To avoid this difficulty, the substructure techniques [23] or the blocked active column method [14, 15] in which the coefficient matrix is stored in a compacted block form on a secondary storage can be utilized. However, the penalties of these methods are the overhead caused by the associated data transfer among substructures or blocks, and the complexity in the implementation.

In contrast to the direct method, the iterative method which is self-correcting repeats the corrections over and over. Iterative methods are attractive because their implementation requires storing only A, b, x and a few other vectors. For example, two vectors need to be stored in conjugate gradient method. Moreover if special acceleration algorithms and conditioning matrices [6, 13, 20] are applied, the iterative method can lead to a solution faster than the direct method.

Recently, multigrid methods have been proposed as efficient schemes, especially for the solution of system resulting from finite element or finite difference mesh refinements [19, 24]. The advantage of the multigrid methods can be realized by forming the coefficient matrix in a special way. It is constructed by hierarchical formulation which is called the absolute relative displacement formulation [16-18, 21]. Without cycling through the multigrid iterations, solution of Eq.(1) after multilevel refinements can approximate the long and the short wave components of the solution.

Here, a conjugate gradient method is reviewed briefly and the incomplete Cholesky conjugate gradient method is compared with other various preconditioned conjugate gradient algorithms.

2. CONJUGATE GRADIENT METHOD

This method was introduced by Hestenes and Stiefel [5] to solve a symmetric, positive definite linear system, and it was regarded as an inefficient iterative scheme when compared with SOR(successive over-relaxation) method. It was considered as a direct method, because it yields a solution within n iterations in exact arithmetics. However, it might take more iterations due to round off error. Becasue the method depends on the properties of A, it converges slowly when the matrix A is ill-conditioned.

If the method is applied to a preconditioned A the method, in general, yields a good approximation after a small number of iterations. Recently, with the use of matrix splitting. $A=M-N$, as of matrix preconditioning, it has proven to be an efficient method[6-13, 20]. The preconditioned conjugate gradient algorithm and the minimum residual are detailed in reference [4] in addition to numerical study of various other methods.

Although the conjugate gradient method can be used only with M, which is symmetric positive definite, it can be applied to unsymmetric equation after they are transformed into normal equations. This type of application can be also be accelerated by preconditions [4, 12].

2.1 Review of Derivation

When we solve $Ax=b$ iteratively it has a residual,

$$r_k=b-Ax_k \tag{2}$$

Hence, x_k is improved at each iteration by a contribution of the residual r_k. The improvement can in general be written as

$$x_{k+1}=x_k+Br_k \tag{3}$$

where the matrix B is, for example, $(D+L)^{-1}$ for Gauss Siedel iteration and D^{-1} for Jacobi iteration where $A=L+D+L^T$

The iterative procedure reduces this residual successively in the direction of Br_k to reach a solution x which minimize a norm of the error $\|x-A^{-1}b\|$ or a norm of the residual $\|b-Ax\|$. Thus, the rate at which the iterated solution

x_{k+1} converges to the solution x is reflected by the rate that an iterative scheme minimizes the following functional:

$$J(x)=(x, Ax)-2(x, b) \tag{4}$$

where(...) is defined by the inner product of two vectors.

Suppose the functional J(x) is to be minimized in the direction of p for any nonzero residual which satisfies k-th iteration:

$$J(x_{k+1}) < J(x_k) \tag{5}$$

where

$$x_{k+1}=x_k+\alpha_k p_k \tag{6}$$

Then the optimal α_k can be obtained by minimizing the functional $J(x_k+_1)$ with respect to α_k.

Thus, assuming A is symmetric and positive definite, Eq.(6) can be rewritten as

$$x_{k+1}=x_k+\frac{(p_k, r_k)}{(p_k, Ap_k)}\ p_k \tag{7}$$

The next step is to choose the direction vector p_k. If p_k is chosen to be the same as r_k, which is equivalent to the steepest descent, $-\nabla J(x_k)$, then it causes the functional $J(x_k+\alpha_k p_k)$ to be maximally reduced from $J(x_k)$. Then the residual at the k−th step can be written as

$$r_{k+1}=\Pi_k r_k \tag{8}$$

where

$$\Pi_k=I-\alpha_k A \tag{9}$$

At each iteration, the residual r_k is projected to r_{k+1}. The convergence rate of the steepest descent method at the k−th step is dependent upon how much r_k is reduced to r_{k+1}. The projection matrix Π_k, however, is very sensitive to the condition number $K_2(A)=\frac{\lambda_{max}}{\lambda_{min}}$. That is, if A is ill-conditioned the level curve of J(x) becomes more elongated hyperellipsoid and the search direction is not closely directed toward the minimum point. To avoid slow convergence due to the above, the conjugate procedure chooses the A-conjugate search direction which satisfies

$$(p_k, Ap_i)=0 \quad (i=1, \cdots, k-1) \tag{10}$$

Therefore, assuming p_k as

$$p_k=r_k+\beta_{k-1}p_{k-1} \tag{11}$$

the scalar β_{k-1} can be obtained from Eq.(10) and Eq.(11). Finally α_k and β_k can be writtern as

$$\alpha_k=\frac{(r_k, r_k)}{(p_k, Ap_k)} \tag{12}$$

and

$$\beta_k=-\frac{(p_k, Ar_{k+1})}{(p_k, Ap_k)} \tag{13}$$

2.2 Preconditioned Conjugate Gradient Algorithm

The error bound in the conjugate gradient algorithm for the initial guess x_1,

$$\| x-x_k \| \le 2\ \| x-x_1 \|_A \left[\left(\frac{\sqrt{k}-1}{\sqrt{k}+1}\right)^k+\left(\frac{\sqrt{k}+1}{\sqrt{k}-1}\right)^k\right]^{-1} \tag{14}$$

where $\| x \|_A$ is A-norm defined by (x, Ax) and $k=\frac{\lambda_{max}}{\lambda_{max}}$ is given in reference [22].

Fixing the initial error, the condition number of A can imply the convergence rate of the system. A sharp error bound which indicates some local convergence rate is presented in reference [20] while Eq.(14) is used in global sense.

Hence to get a faster convergence it may be desirable to lower the condition number f A. If a good preconditioning of A is used, most of the eigenvalues of the preconditioned matrix are clustered around 1. Precisely speaking, a good approximation can be obtained within a small number of iterations if the eigenvalues corresponding to the first small number of iterations are clustered around 1. Many preconditioning matrices have been proposed for this purpose [6-13, 20]. If A is split into M-N, then a preconditioned matrix $Q^{-1}AQ^{-T}$ can be used to approximate A and it is better to use conditioned A. The matrix Q can

be obtained as long as $M(=Q^2)$ is positive definite.

The preconditioned system becomes

$$(Q^{-1} AQ^{-T})\ (Q^T x)=Q^{-1}b \tag{15}$$

or

$$\tilde{A}\tilde{x}=\tilde{b}$$

The following algorithm solves the modified problem implicitly without calculating $\tilde{A}$ and $\tilde{x}$.

Given x_1

$r_1=b-Ax_1$

Solve $Mz_1=r_1$

$p_1=z_1$

For $k=1, \cdots$ until $r_k<\varepsilon$

$$\alpha_k=\frac{(z_k,\ r_k)}{(p_k,\ Ap_k)}$$

$x_{k+1}=x_k+\alpha_k p_k$

$r_{k+1}=r_k-\alpha_k Ap_k$ (16)

Solve $Mz_{k+1}=r_{k+1}$

$$\beta_k=\frac{(z_{k+1},\ r_{k+1})}{(z_k,\ r_k)}$$

$p_{k+1}=z_{k+1}+\beta_k p_k$

3. INCOMPLETE CHOLESKY CONJUGATE GRADIENT METHOD

In order to be practical, an algorithm must be robust and efficient; i.e. it must be stable, have a high rate of convergence, be simple to be implemented and have small storage requirecent. For the incomplete Cholesky conjugate gradient method originally proposed by Meijerink and Van der Vorst [6], several versions have been proposed [9-11]. The preconditioning strategy involves an incomplete Cholesky decomposion of A as follows.

$$\tilde{L}\tilde{L}^T=A+C \tag{17}$$

where C is used as a means of disregarding the fill-in caused when A is factored. If $\tilde{L}$ is exact decomposition of A, C becomes null matrix. The matrix C is not computed explicitly. The simplest method is to set $l_{ij}=0$ when a_{ij} is zero. Subsequently, the algorithm (16) is applied to

$$(\tilde{L}^{-1} A\tilde{L}^{-T})\ (\tilde{L}^T x)=\tilde{L}^{-1}b \tag{18}$$

However, when A is factored by incomplete Cholesky decomposition, it may not be stable, because the following calculation is required:

$$\hat{A}=\hat{L}\hat{D}\hat{L}^T=\tilde{L}\tilde{L}^T \tag{19}$$

where

$$\hat{A}=A+C \tag{20}$$

$$\tilde{L}=\hat{L}\,\mathrm{diag}(\hat{d}_1^{1/2}, \cdots, \hat{d}_n^{1/2})$$

Eq. (19) can be rewritten in partitionec form by

$$\hat{A}=A_1=\begin{bmatrix}\alpha_1 & v_1^T\\ v_1 & B_1\end{bmatrix}$$

$$=\begin{bmatrix}1 & 0\\ w_1 & I\end{bmatrix}\begin{bmatrix}\alpha_1 & 0\\ 0 & A_2\end{bmatrix}\begin{bmatrix}1 & w_1^T\\ 0 & I\end{bmatrix} \tag{21}$$

where $w_1=v_1/\alpha_1$ and $A_2=B_1-v_1v_1^T/\alpha_1^2$.

Next, let

$$A_2=\begin{bmatrix}\alpha_2 & v_2^T\\ v_2 & B_2\end{bmatrix}$$

and follow the procedure outlined above. It is noted that A_i, $i=1, \cdots, n$, may not be positive definite, i.e. some $\hat{d}_i$ are less than and equal to zero due to the incomplete Cholesky decomposition. Thus the stability is not guaranteed even for a symmetric positive definite matrix A. Two sufficient conditions for the stability of the incomplete Cholesky decomposition are known: The one is that A should be a diagonally dominant (hence positive definiteness is assured since we consider symmetric A) and the other is that A should be a Stieltjes matrix [6, 9]. However, the stiffness matrix is only guaranteed to be symmetric positive definite. Thus, there is a need for an algorithm which avoids those limitations and yields positive definitess for the matrix $\hat{A}$ of Eq.(19).

A special decomposition such as incomplete banded shifted LU decomposition [3] can be considered if the matrix A has a special form of nonzero elements.

3.1 Shifted Incomplete Cholesky Factorization

The convergence of conjugate gradient iteration is sensitive to the spectrum of a preconditioned matrix [20] and the diagonal scaling does not change the spectrum. Noting that the procedure of conjugate gradient iteration is invariant to the diagonal scaling, Manteuffel [9] used the matrix **M** as $D=\mathrm{diag}(a_{11}, \cdots, a_{nn})$ where a_{ii} is a diagonal element of **A**; The preconditioned matrix is written by

$$\tilde{A}=D^{-1/2}\ AD^{-1/2}$$

where $\mathrm{diag}(\tilde{A})=I$

If $\tilde{A}$ is split as

$$\tilde{A}=I+B \qquad (23)$$

then **B** is the same as $\tilde{A}$ except **B** has zero diagonal elements.

To make $\tilde{A}$ diagonally dominant, **B** is multiplied by a scalar γ.

$$\bar{A}=I+\gamma B \qquad (24)$$

$\bar{A}$ is factorized by the incomplete Cholesky decomposition. $\tilde{\bar{L}}\tilde{\bar{L}}^T=\bar{A}+C$ and then it is used with the algorithm (16). Note that $\bar{A}$ becomes diagonally dominant as the factor γ decreases, and that if γ approaches 1 this scheme asymtotically yields the Jacobi splitting [9]. The difficulty is how to determine an optimal γ, on which the convergence depends, since it depends on the structure of matrix **A**.

3.2 Practical Modification

The incomplete Cholesky decomposition is a fast but non-exact decomposition exploiting the large sparsity of **A**. Thus, the following algorithm is proposed for the modification of diagonal terms of **L** which effects the instability of the factorization.

For $j=1, \cdots, n$

$$l^*_{jj}=(a_{jj}-\sum_{i=1}^{j-1} l^2_{ji})^{1/2}$$

if $l_{jj}\simeq 0$ or $l_{jj} < 0$

then $l_{jj}=(a_{jj})^{1/2}$

else $l_{jj}=(l^*_{jj})^{1/2}$ (25)

For $i=j+1, \cdots, n$

if $a_{ij}=0$ then $l_{ij}=0$

else

$$l_{ij}=(a_{ij}-\sum_{k=1}^{j-1} l_{ik}\ l_{jk})\ /\ l_{jj}$$

This process is followed by algorithm (16) with the matrix **M** being replaced by $\tilde{L}\tilde{L}^T$ where $\tilde{L}$ is the lower triangular matrix computed as above.

4. COMPARISON OF PRECONDITIONINGS

In this section the proposed practical conjugate gradient method with the incomplete Cholesky decomposition (ICCG) is tested for two model examples to be compared with other preconditionings such as the Jacobi conjugate gradient method (JCG), the SSOR conjugate gradient method(SSOR-CG), the conjugate gradient method with the shifted incomplete Cholesky decomposition (SCG), and the conjugate gradient method without the precondition(CG).

Firstly, the coupled spring model is considered (Figure 1) and secondly the membrance example is tested(Figure 2). The **M** matrix ($M=Q^2$; **Q** is preconditioning matrix) is as follows:

ICCG : $M=\tilde{L}\tilde{L}^T$ (26)

JCG : $M=D$ (where $A=L+D+L^T$) (27)

SSOR-CG: $M=(D+\omega L)D^{-1}(D+\omega L)^T$ (28)

SCG : $M=\tilde{\bar{L}}\tilde{\bar{L}}^T$ (29)

The tolerance of error, $\varepsilon=\|Ax-b\|\ /\ \|b\|$, was set to be 10^{-7} for the tests.

For the spring model, the rotational stiffness k_3 contributes to the diagonal term of the entire stiffness matrix, the coupling stiffness k_1 makes the matrix more ill-conditioned as it grows larger, and k_2 contributes to the tridiagonal terms. In the third case($k_1=10, k_2=1, k_3=1000$) the stiffness matrix is diagonally dominant even without pre-

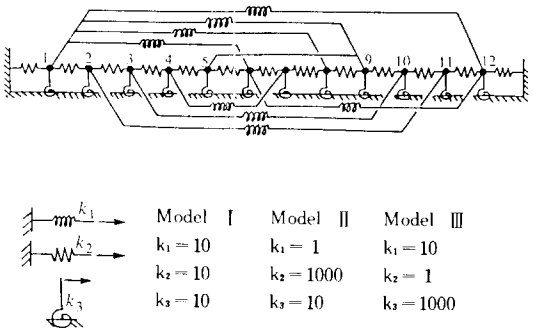


Figure 1. Coupled spring system

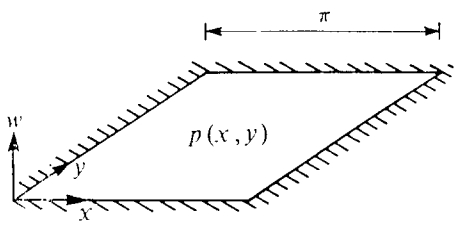


Figure 2. Square membrane

Table 1. Condition Numbers and Spectral Radius

Model	Cond($\tilde{A}$)	Cond($\tilde{A}$)	$\frac{\text{Cond}(A)}{\text{Cond}(\tilde{A})}$	$\rho(\tilde{A})$	$\rho(\tilde{A})$
spring I	7.40	1.58	4.68	83.27	1.15
spring II	502.00	60.45	8.30	5098.03	1.97
spring III	1.03	≈1.00	1.03	1041.06	≈1.00
membrane	48.41	5.13	9.44	7.84	1.18

where
$\tilde{A} = \tilde{L}^{-1} A (\tilde{L}^{-1})^T$
A : Stiffness Matrix
$\tilde{L}$: Lower Triangular Matrix by Incomplete Cholesky Decomposition
ρ : Spectral Radius

conditioning (see Table 1), so the convergence rates for all methods are very high. In the case of ICCG the preconditioned matrix $A(=\tilde{L}^{-1} A \tilde{L}^{-T})$ is very close to the identity matrix I multiplied by scalar as tabulated in Table 1, which produces a high rate of convergence(see Figure 5). Figures 3, 4 and 5 indicate that the advantages of the preconditioning methods: ICCG, SCG and SSOR-CG for three cases tested.

It is noted that CG results in a faster convergence than ICCG for the large off diagonal case(see

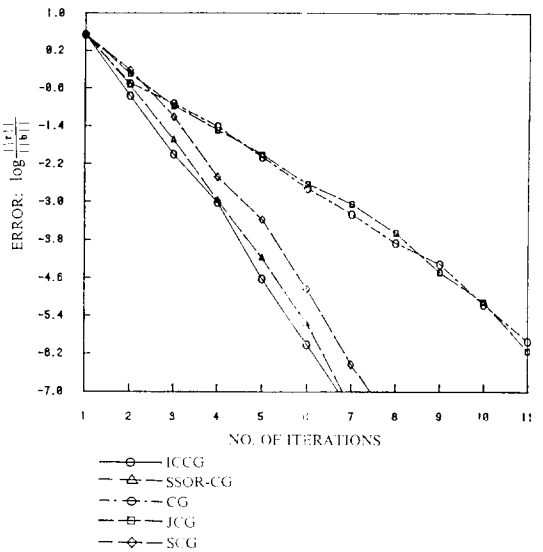


Figure 3. Convergence rates for spring model-I

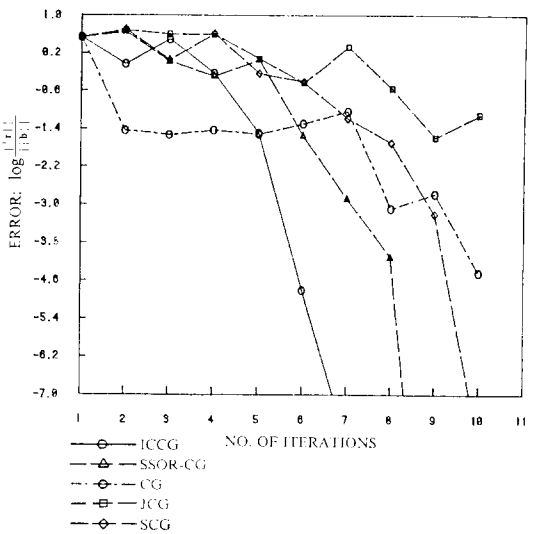


Figure 4. Convergence rates for spring model-II

Figure 4) at the begining of iterations and then it converges slower as the iteration continues. By looking at the Table 2 it can be noted that the convergence rate is dependent on the spectrum ratio of the stiffness matrix. The ratios of spectrum, λ_J / λ_1, are in general reduced by the preconditioning so faster convergence than CG can be achieved. In the case of spring model II, λ_2 / λ_1 has been increased to 28.8938, while λ_2 / λ_1 is 1.1410. That

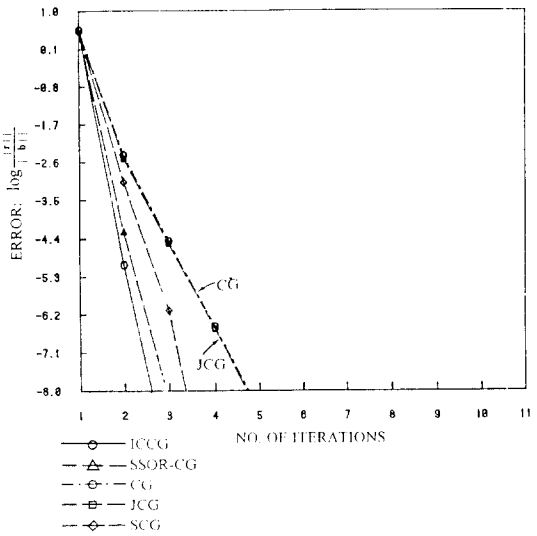


Figure 5. Convergence rates for spring model-Ⅲ

Table 2. Eigenspectrum Ratios of Spring Model

j/i	Model Ⅰ		Model Ⅱ	
	λ_j/λ_i	$\tilde{\lambda}_j/\tilde{\lambda}_i$	λ_j/λ_i	$\tilde{\lambda}_j/\tilde{\lambda}_i$
2 / 1	1.9331	1.2510	1.1410	28.8938
3 / 2	1.2942	1.0300	1.0658	1.0590
4 / 3	1.1753	1.0160	1.0734	1.0039
5 / 4	1.0283	1.0521	37.6216	1.0002
6 / 5	1.1603	1.0192	2.2104	1.0002
7 / 6	1.1652	1.0097	1.9870	1.0001
8 / 7	1.1019	1.0294	1.0006	1.0001
9 / 8	1.1577	1.0266	1.0006	1.0018
10 / 9	1.1088	1.0088	1.2121	1.0021
11 / 10	1.0542	1.0212	1.2344	1.0511
12 / 11	1.2140	1.0426	1.6926	1.8640

$(\lambda_1 \leq \lambda_2 \leq \cdots \leq \lambda_{12})$

causes a very slow convergence for ICCG. Moreover, its effect lasts for several iterations. To investigate the effect of preconditioned ICCG, the condition number and the spectral radius are tabulated in Table 1. Based on these results, it is apparent that there is only global improvement over CG.

For the membrane problem, the equilibrium equation becomes

$$-\nabla^2 w(x, y) = \frac{p(x, y)}{S} \qquad (30)$$

where ∇^2 is the Laplacian operator, $p(x, y)$ is the distributed load; S is the force per unit area; and the boundary condition is set to be fixed along the boundary.

Let the matrix A correspond to $-h^2\nabla^2$ on the space of discrete functions in the discretized $(0, \pi)x(0, \pi)$ where $h=\triangle x=\triangle y=\pi/(N+1)$. Then, the above boundary value problem in the 2nd divided difference form yields the following equation.

$$Aw=f \qquad (31)$$

where A is the N^2 by N^2 matrix

All preconditioning methods except JCG produced the faster convergence than CG(see Figure 6) as before. SCG shows especially a good convergence rate.

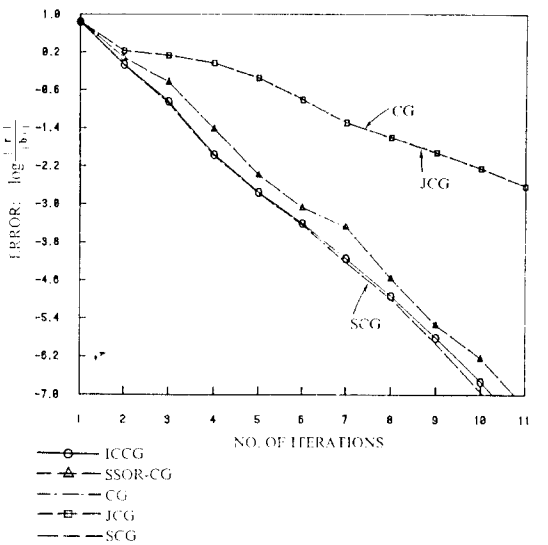


Figure 6. Convergence rates for membrane model

As mentioned for SCG, the determination of the parameter γ is difficult and a priori estimation of it has not been estbalished. As an example, the value of γ is shown in Figure 7 for the spring model. The same difficulty exist in determining the optimum ω. Figure 8 indicates that the optimum ω might depend on the size of the system. For the membrance model(100 degrees of freedom

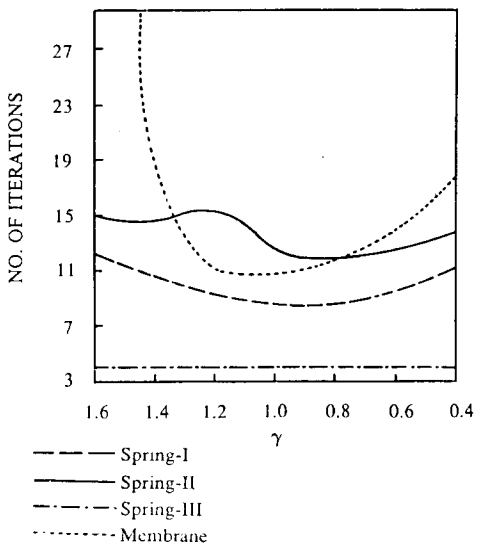


Figure 7. Effects of γ on rate of convergence $\left(\text{for SCG with } \frac{\| r \|}{\| b \|} \leq 10^{-6}\right)$

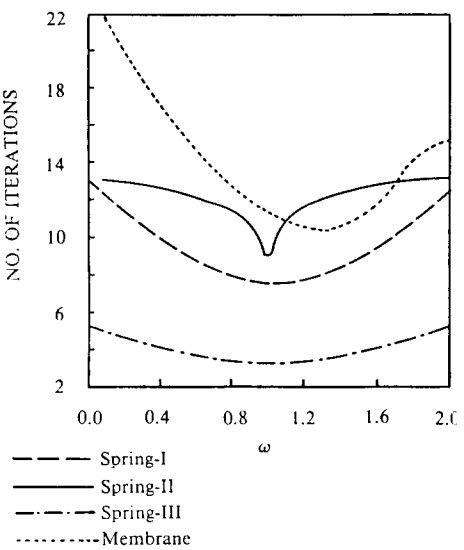


Figure 8. Effects of ω on rate of convergence $\left(\text{for SSOR-CG with } \frac{\| r \|}{\| b \|} \leq 10^{-6}\right)$

(dofs)), the optimum ω ranges between 1.35 and 1.50 and for the spring model I, II and III (16 dofs) between 1.0 and 1.2. Because of the test results stated above and the difficulty of finding optimal parameters of other preconditioning methods, the practical version of ICCG is the obvious choice among the methods examined for the numerical solver in the adaptive finite element analysis.

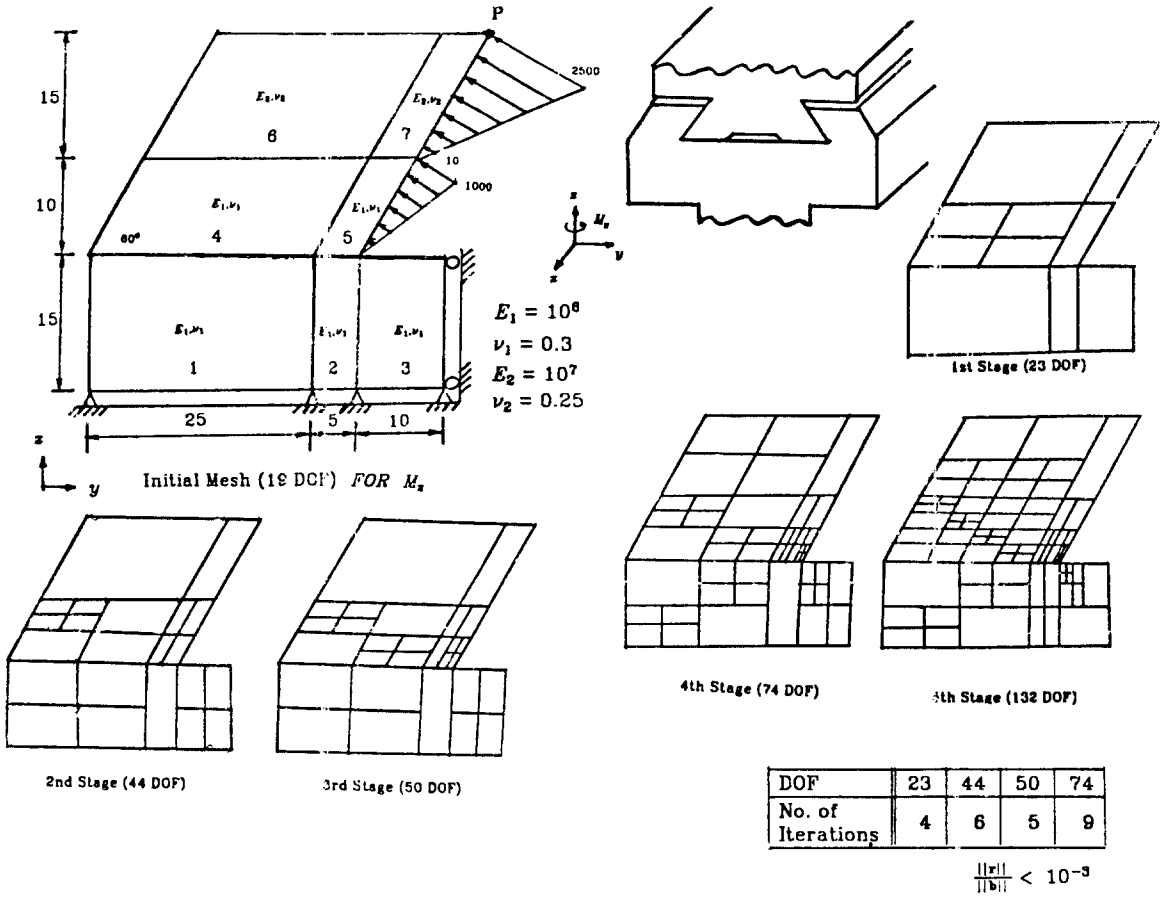
5. CONCLUDING REMARKS

Four preconditionings to the conjugate gradient methods were compared to find a practical preconditioned conjugate algorithm for the linear system of equations arising at the adaptive refinement levels. It is concluded that for the large sparse system the incomplete Cholesky conjugate method is the best solution technique among the chosen comparisons with respect to reliability and convergency.

The condition number of the coefficient matrix does not fully reflect the convergence characteristics of the conjugate gradient algorithm. However, it can be a reliable indication of the global convergence rate. For the local behavior of the convergence rate, the eigen spectrum ratio can be considered. Thus, the incomplete Cholesky conjugate gradient method is also appropriate for the unsymmetric matrix **A** as long as the eigen spectrum ratios of $\tilde{L}^{-1}(A^TA)\ (\tilde{L}^{-1})^T$ are clustered around 1 or the number of iterations to reach the required accuracy is at least less than the number of the first part of eigen spectrum ratios which are clustered around 1.

The incomplete Cholesky decomposition is not always stable even for the symmetric positive definite matrix. However, based on the author's experience, the proposed algorithm has not produced any difficulties; the number of iterations needed to converge to a solution within a required accuracy is less than the number of equation (see Figure 9 as an example).

Hence, the proposed incomplete Cholesky conjugate gradient is recommended as the solution method for the linear sparse system of equations resulting from adaptive finite element analysis.



DOF	23	44	50	74
No. of Iterations	4	6	5	9

$\frac{||r||}{||b||} < 10^{-3}$

Figure 9. Adaptive mesh refinements and no. of iterations iterations via ICCG for a dovetail joint, whose upper part is subjected to the moment about z-axis (Mz) and modeled with 4-node quadrilateral plane strain elements (See Ref. [25] for node constraining)

REFERENCES

1. A. George and J. W. Liu, Computer Solution of Large Sparse Positive Definite Systems, Prentice-Hall, Englewood Cliffs, New Jersey (1981)
2. G.H. Golub and C.F. Van Loan, Matrix Computations, the Johns Hopkins University Press, Baltomore, Maryland (1983)
3. Dao-Qi Chen and R.P. Tewarson, "Use of incomplete decomposition of the coefficient matrix in solving linear equations", Int. J. Num. Meth Eng., 23, 199-208 (1986)
4. C.P. Jackson and P.C. Robinson, "A numerical study of various algorithms related to the preconditioned conjugate gradient method", Int. J. Num. Meth. Eng., 21, 1315-1338 (1985)
5. M.R. Hestenes and E. Stiefel, "Methods of conjugate gradients for solving linear systems", J. Res. Nat. Bur. Stand., 49, 409-436 (1952)
6. J.A. Meijerink and H.A. Van der Vorst, "An iterative solution method for linear systems of which the coefficient matrix is a symmetric M-matrix", Mathematics of Computations, 31, 148-162 (1977)
7. J.A. Meijerink and H.A. Van der Vorst, "Guidelines for the usage of incomplete decompositions in solving sets of linear equations as they occur in practical problems", J. Computational Physics, 44, 134-155 (1981)
8. D.S. Kershaw, "The incomplete Cholesky-conjugate

gradient method for solution of system of linear equations", J. of Computational Physics, 26, 43-65(1978)

9. T.A. Manteuffel, "Shifted incomplete Cholesky factorization", in Sparse Matrix Proceedings, 1978, ed. I.S. Duff and G.W. Stewart, SIAM publications, Philadelphia(1979)
10. M.A. Ajiz and A. Jennings, "A robust incomplete Cholesky conjugate gradient algorithm", Int. J. Num. Meth. Eng., 20, 949-966(1984)
11. A. Jennings and G.M. Malik, "The solution of sparse linear equations by the conjugate gradient method", Int. J. Num. Meth. Eng., 12, 141-158 (1978)
12. Henk A. van der Vorst, "Iterative solution methods for certain sparse linear systems with a mon-symmetric matrix arising from PDE-problems", J. Computational Physics, 44, 1-19(1981)
13. Henk A. van der Vorst, "A vectorizable variant of some ICCG methods", SIAM J. Sci. Stat. Comput., 3, 350-356(1882)
14. E.L. Wilson, "Solution or reduction of equilibrium equations for large complex structural systems", J. Adv. Eng. Soft., 1, 19-25(1978)
15. E.L. Wilson, K.J. Bathe and W.P. Doherty, "Direct solution of large systems of linear equations", Comp. & Stru., 4, 363-372(1974)
16. E. L. Wilson, "Filite elements for foundations, joints and fluids", in Finite Elements in Geomechanics, ed. Gudehus, 319-350, Wiley, New York(1977)
17. E.L. Wilson, "Special numerical and computer techniques for the analysis of finite element systems", in Formulation and Computational Algorithm in Finite Elements Analysis, ed. Bathe, Oden and Wunderich, 3-25, MIT Press(1978)
18. E.L. Wilson, "Solution of sparse stiffness matrices for structural systems", in Sparse Matrix Proceedings, 1978, ed. I.S. Duff and G.W. Stewart SIAM Publications, Philadelphia(1979)
19. A. Brandt, "Multi-level adaptive solutions to boundary value problems", Mathematics of Computations, 31, 339-390(1977)
20. A. Greenbaum, "Comparison of splitings used with the conjugate gradient algorithm", UCRL-80800, Lawrence Livermore Laboratory(1978)
21. O.C. Zienkiewicz, J.P. DE S.R. Gago and D.W. Kelly, "The hierarchic concepts in finite element analysis", Comp. & Stru., 16, 53-65(1983)
22. A.K. Cline, "Several observations on the use of conjugate gradient methods", ICASE, Report No.76-22(1976)
23. D.G. Row and G.H. Powell, A Substructure Technique for Nonlinear Static and Dynamic Analysis, Report No. EERC 78-15, University of California, Berkeley, August(1978)
24. A. Brandt, "Guide to multigrid development", in Multigrid Methods, Proc. conf., Koln-Porz, Nov., 1981, (Ed. by M.H. Hackbusch and U. Trottenberg), Lecture Notes in Mathematics, Spring-Verlag, Berlin, 1982.
25. K.J. Joo and E.L. Wilson," An adaptive finite element techinique for structural dynamic analysis", Computers & Structures(1988)

(접수일자 1988. 10. 18)