A METHOD FOR DETERMINING EIGENSOLUTIONS OF LARGE, SPARSE, SYMMETRIC MATRICES BY THE PRECONDITIONED CONJUGATE GRADIENT METHOD IN THE GENERALIZED EIGENVALUE PROBLEM

一般固有値問題における前処理付共役勾配法による
大次元スパース対称行列の固有解の一算定法

Mitsuhiro KASHIWAGI*

柏木 光博

This paper presents a conjugate gradient method for the eigensolutions of large, sparse and symmetric matrices using Sylvester's law of inertia and the quadratic form. Since the proposed method is a PCG-based method, it is particularly effective when a sequence of either the smallest or largest eigensolutions of a large and sparse matrix are required. Moreover, the proposed method does not find unnecessary solutions, and it thus minimizes the required computational memory capacity. The proposed method is a method requiring only a relatively small computational time. The accuracy and stability of this method are confirmed by considering several numerical examples. The numerical results are of high accuracy even for the systems with multiple eigenvalues.

Keywords : *Generalized eigenvalue problems; Large, sparse and symmetric matrices; conjugate gradient method; preconditioning; Sylvester's law of inertia*

一般固有値問題, 大次元スパース対称行列, 共役勾配法, 前処理, シルベスター慣性則

1 INTRODUCTION

The development of computational science and computational engineering is closely related to the development in performance of computers. With the rapid, exponential improvements in the performance of modern computers, including computational speed, capacity and so on, numerical simulations in computational dynamics and so on are now of practical use^{1),4),6),9),10),13),15)}. Furthermore, analytical methods used in many fields lead to problems requiring the solution of large-dimensional sets of simultaneous linear equations or eigenvalue problems. Recently, solution techniques for these problems, including the iterative PCG method applied to the conjugate gradient method, with preconditioning by the ICCG or SCG methods, and the AMG and CG methods, and also the implementation of these techniques appropriate for parallel processing, have been attracting attention. Since these methods can significantly reduce computational memory requirements, they provide a powerful analytical technique for large, sparse and symmetric matrices, which frequently arise from the application of finite element methods for example.

For eigenvalue problems, there exist classic methods such as those

based on the Householder transformation in conjunction with the power method, or the subspace method. However, the Householder transformation applied to generalized eigenvalue problems creates dense matrices, and does not exploit the sparse nature of the matrices that arise from finite element applications. Thus high-dimensional problem analysis is difficult by this approach. The subspace method³⁾ yields solutions for near-degenerate and degenerate eigenvalue problems relatively easily, and has benefited from a number of studies investigating acceleration techniques, many of which are now used in practical applications. This method can yield a sequence of several eigensolutions (that is eigenvectors corresponding to the eigenvalues) for a corresponding sequence of the largest or, alternatively, the smallest eigenvalues. However, it is impossible to confirm the existence of desired solutions in an iteration method such as the PCG method.

The Lanczos method^{2),6),11),12)}, which is effective for large, sparse and symmetrical eigenvalue problems, can reduce a matrix to a tridiagonal form in a similar way to the Householder method, but it also gives condensed or spurious eigenvalues due to rounding error or a break in the conjugacy.

* Prof., Dept. of Architecture, School of Industrial Engineering, Tokai Univ., Dr. Eng.

東海大学産業工学部建築学科 教授・工博

The Block Lanczos Method ^{5),6)} was introduced in order to avoid these problems, but there still remain number of difficulties including the complexity involved in setting the block size and the number of blocks, and the increase in the required memory capacity and computational time. Furthermore, it is impossible to confirm existence of the desired solutions by using this method. The restarted-Lanczos method ²⁾ iterates subspace eigenvalue problems in order to obtain eigensolutions, and has similar properties to those of the subspace methods. If a block matrix is used for an eigenvalue analysis by means of minimizing the Rayleigh quotient ²⁾, the necessary computational time is small, but the existence of solutions cannot be proven. The Jacobi-Davidson method ¹⁴⁾ is an effective method which, unlike Krylov subspace iteration ^{2),6)} based methods such as the Lanczos method, returns accurate values for eigenvalues even in near-degenerate cases. However, the algorithm of the Jacobi-Davidson method ²⁾ is relatively complicated compared to those of the subspace and Lanczos methods, and it is not very efficient for eigenvalue problems arising from finite element analyses, whose diagonal dominance is rather weak.

This article proposes a method to calculate eigensolutions of large, sparse and symmetric matrices. While preserving the advantages that conjugate gradient methods possess for large, sparse linear problems, the proposed method ensures that the number of eigenvalues is theoretically countable because, by Sylvester's law of inertia, the number of signs of the object matrix can be counted. If the method is free from numerical errors, it yields eigensolutions in the same way as the Householder bisection method, but it does not require the solution of subspace eigenvalue problems, in contrast to iterative methods based on Krylov subspaces. However, in practice, except for the case when the dimension of the problem is extremely low, it is impossible to correctly count the number of changes of signs in the neighborhood of an eigenvalue due to a break in the conjugacy caused by rounding error. To deal with this problem, a method was proposed from which an ascending sequence of eigensolutions can be obtained rapidly and reliably, requiring only a limited memory capacity, by using the quadratic form of a matrix expressed by its eigenvectors, and exploiting knowledge of whether or not the matrix is positive-definite ⁷⁾.

The proposed method uses an iterative technique within a conjugate gradient method system preconditioned by the ICCG or SCG method, or a shifting inverse power method with the bisection method, to yield eigensolutions for large, sparse and symmetric matrices. In particular, a distinguishing feature is that, by using the inverse power method, an ascending or descending sequence of eigensolutions can be obtained by using the same numerical technique. However, at present it is impossible to precisely estimate eigenvalues in the middle of the spectrum, because the number of signs of the matrix cannot be calculated accurately. In contrast to the inverse power or subspace methods, it is possible to avoid the calculation of unnecessary eigenvalues by the proposed method, so it can minimize the memory requirement of the computation. Avoiding the calculation of unnecessary eigenvalues, together with the property of the method which exploits the information of whether or not the matrix is

positive definite, also ensures that the necessary computational time is significantly reduced. For the case of multiple roots, the convergence of methods to determine eigensolutions often becomes slower. However, in contrast, the proposed method actually improves this weak point. Furthermore, the proposed method possesses a stability preserving quality, since the obtention of eigensolutions in either ascending or descending order is guaranteed. Theoretical aspects and a description of the algorithm of the proposed method are explained in the following sections, along with a consideration of its efficiency through the application to a standard eigenvalue problem with multiple roots, and a generalized eigenvalue problem derived from finite element analysis.

2 EIGENSOLUTIONS OBTAINED BY THE CONJUGATE GRADIENT METHOD

2.1 Changes of sign by Sylvester's Law of Inertia

Consider the set of simultaneous linear equations given by

$$Ax = b, \quad (1)$$

where A is a $n \times n$ real symmetric matrix, x is an unknown vector, and b is a known vector. Let C be the incomplete inverse of A . For example, $C = [LDL^T]^{-1}$ where LDL^T is the incomplete modified Cholesky decomposition, L is an incomplete lower triangular matrix, D is an incomplete diagonal matrix and the superscript T denotes the transposition matrix operation. When C is a positive definite symmetric matrix, a preconditioned conjugate gradient method (PCG method) can be described as follows:

for integers $i = 0, 1, 2, \dots$,

$$\alpha_i = \frac{1}{p_i^T A p_i}, \quad (2)$$

$$x_{i+1} = x_i + \alpha_i p_i, \quad (3)$$

$$r_{i+1} = r_i - \alpha_i A p_i, \quad (4)$$

$$p_{i+1} = p_i + \frac{C r_{i+1}}{r_{i+1}^T C r_{i+1}}, \quad (5)$$

and

$$r_0 = b - A x_0 \quad (x_0 : \text{arbitrary}), \quad (6)$$

$$p_0 = \frac{C r_0}{r_0^T C r_0}, \quad (7)$$

where 0 represents the initial step, i denotes the i -th iteration step, r_i is the residual vector, p_i is the conjugate direction vector, x_i is an approximate solution vector and α_i is a scalar. In the conjugate gradient method, the conjugate direction vectors $p_0, p_1, \dots$ and a conjugate direction matrix $P = [p_0, p_1, \dots, p_{n-1}]$ satisfy the following relations:

$$p_i^T A p_j = 0 \quad (i \neq j), \quad (8)$$

$$P^T A P = \text{diag}(p_0^T A p_0, p_1^T A p_1, \dots, p_{n-1}^T A p_{n-1}). \quad (9)$$

From equations (8) and (9), the number of signs of the matrices A and $P^T A P$ is the same as Sylvester's law of inertia, and the number of changes in the signs is the same as the number of negative signs of $\text{sign}(p_i^T A p_i) \quad (i = 0, 1, \dots, n-1)$.

2.2 Application to the CG Method

Consider a generalized eigenvalue problem in the form

$$[\tilde{A} - \lambda \tilde{B}] \phi = 0, \quad (10)$$

where $\tilde{A}$ is an $n \times n$ real symmetric matrix, $\tilde{B}$ is an $n \times n$ real, symmetric and positive definite matrix, λ is an eigenvalue and ϕ is the corresponding eigenvector. Solution of free vibration and buckling problems by the Finite Element Method give rise to typical examples of the eigenvalue problems arising from the application of this method. In a free vibration problem, $\tilde{A}$ represents the stiffness matrix K , and $\tilde{B}$ is the mass matrix M . In a buckling problem, $\tilde{A}$ is the stiffness matrix, while $\tilde{B}$ represents the geometrical stiffness matrix K_G . If $\tilde{B} = I$ (the identity matrix), then the above problem is a standard eigenvalue problem.

In order to obtain eigensolutions, A should be defined as follows:

$$A = \tilde{A} - \lambda \tilde{B}, \quad (11)$$

where λ is a trial eigenvalue. The number of signs of the matrix A is determined by the PCG method to which Sylvester's law of inertia is applied. However, except the case when the dimension of the matrix is very low, rounding errors destroy the conjugate property of the matrix A with conjugate direction vectors as the computation progresses. Thus, when a trial eigenvalue computed by the suggested method approaches a true eigenvalue, the number of changes of sign cannot be correctly obtained. In order to avoid this instability, the following method is introduced under the assumption that the sign of the quadratic form of A can be correctly found by computational numerical analysis. The quadratic form of a matrix A is expressed as follows:

$$f = x^T A x. \quad (12)$$

From equations (1), (11) and the references 9, the above equation can be written in a same form to equation (13),

$$\begin{aligned} f &= b^T [\tilde{A} - \lambda \tilde{B}]^{-1} b \\ &= b^T \sum_{m=1}^n \frac{\phi_m \phi_m^T}{\lambda_m - \lambda} b \\ &= \sum_{m=1}^n \frac{(b^T \phi_m)^2}{\lambda_m - \lambda}, \end{aligned} \quad (13)$$

in which ϕ_m is the m -th eigenvector, where $m=1$ denotes the smallest eigenvalue.

[Calculating a sequence of eigensolutions ascending from the minimal eigensolution]

When extracting l eigensolutions which start from the minimal eigensolution, equation (13) is expressed by the following form:

$$f = \sum_{m=l+1}^n \frac{(b^T \phi_m)^2}{\lambda_m - \lambda}. \quad (14)$$

When $\lambda < \lambda_{l+1}$ and $b^T \phi_m \neq 0$, $f > 0$, that is A is a positive definite

matrix, the number of signs is equal to 0, and the computation by the proposed method is stable. On the other hand, when $\lambda > \lambda_{l+1}$ and $b^T \phi_m \neq 0$, $f < 0$, that is A is non-positive definite. In this case, except for when the dimension is very small, it is difficult to identify the number of signs correctly due to the break in conjugacy. Therefore it is not possible to proceed with the computation. If there is just a single change in sign, then the number of eigenvalues is not 0, and the computation by the PCG method stops. The number of iterations of the PCG method is usually very small when the computation stops. Therefore the required computational time is relatively small. Furthermore, using a method such as the bisection method, the mean of the trial eigenvalue for the case when A is positive definite (hereafter referred to as the left-trial eigenvalue since it is smaller than the true eigenvalue) and the trial eigenvalue for the case when A is non-positive definite (hereafter referred to as the right-trial eigenvalue since it is larger than the true eigenvalue) are used as the left or right trial eigenvalue for the next iteration step. Upon iterating, the eigensolution can be obtained to the desired accuracy. The PCG method is particularly effective in reducing the required computational time for the case of multiple roots. Because, on setting the left-trial eigenvalue to just left of a multiple root, the eigenvector dimension can be reduced, and the right-trial eigenvalue usually causes the calculation to become non-positive definite. Whether or not the problem has multiple roots, if the size of the difference between the left and right trial eigenvalues is maintained, the shifting inverse power method yields more accurate eigensolutions. Although the presence of multiple roots generally leads to a slower convergence, the convergence of the present method is, in contrast, improved for this situation, and so the number of iterations of the PCG method is substantially reduced. In turn, this can reduce the necessary computational time. It may be noted that, by reducing the dimension of the eigenvectors, the present method must find the eigenvectors one by one.

By orthogonalizing x_i and p_i by the Gram-Schmidt Method, smaller eigensolutions can be obtained, and the effect of preconditioning is fully exploited with the PCG method using an incomplete inverse matrix. However, when λ is sufficiently close to the true eigenvalue, the matrix A in equation (11) approaches a singular form, and computations by the PCG method are unstable. The number of iterations in this case increases substantially, and the solution may not converge. In this case, the preconditioner matrix should be positive definite and stable. In particular, for finite element analysis of an elastic body, it has been confirmed that computations can be stabilized by using preconditioner

matrices of $\tilde{A}$ which do not require the consideration of the $\lambda \tilde{B}$ term of equation (11), since $\tilde{A}$ is always a positive definite symmetric matrix. Furthermore, for the case when the $\text{sign}(p_i^T A p_i)$ is positive for all i , (i.e. A is a positive definite matrix) the PCG method can reach its completion, and so an approximate solution vector found by the present method is an approximate eigenvector. It is necessary to retain this vector in the computer's memory as the newest approximate vector, because it will be used either in the right-hand side of the next step, or as an approximate eigenvector for accuracy analysis by the shifting inverse power and bisection method.

[Calculating a sequence of eigensolutions descending from the maximal eigensolution]

Fig.1 f values for trial eigenvalues

Fig.1 shows a plot of f (a quadratic form of matrix A (13)) in 9 dimensions (i.e. in the case when there are 9 eigensolutions) of a standard eigenvalue problem which arises from the discretization of the 2-dimensional Helmholtz equation over a rectangular domain (this problem will be described in more detail later). If calculating a sequence of eigensolutions ascending from the minimal eigensolution, the matrix A is positive definite, whereas calculating a sequence of eigensolutions descending from the maximal eigensolution, the eigensolutions are non-positive definite, and so these two problems exhibit opposite properties. Thus, a different method must be employed for the calculation of the latter sequence of eigensolutions in comparison with the former sequence. This point is easy to see by rotating the plots about the f axis. Clearly what is required here is to force A to become positive (in the present program for example, A was changed to $-A$). Hence, when a sequence of eigensolutions descending from the maximal eigensolution is required, it is possible to obtain eigensolutions in a similar way to those found when it is required to obtain a sequence of eigensolutions ascending from the minimal eigensolution by the inverse power method. It is obviously important to pay close regard to the necessary changes required in making the incomplete Cholesky decomposition which is positive definite, and to the changes in coefficient matrices and products of conjugate direction vectors induced by the change from A to $-A$.

The basic algorithm for the present method is the same as that which is appropriate for calculating a sequence of eigensolutions

ascending from the minimal eigensolution, thus the same program can be used with minor modifications -- it is not necessary to construct two different programs for the two cases. As a consequence of the above discussion, expression (13) can be described for the case when extracting a sequence of l eigensolutions descending from the maximal eigensolution as follows:

$$f = \sum_{m=1}^{n-1} \frac{(b^T \phi_m)^2}{\lambda_m - \lambda}. \quad (15)$$

3 ALGORITHM

[An Example Algorithm for the Proposed Method]

[1] $K=0$ (the initial condition)

RL (left-trial eigenvalue) and RR (right-trial eigenvalue) can be calculated from Gershgorin's Theorem. For the case when $\tilde{A}$ is a positive definite symmetric matrix, RL is set to be zero. Note that in practice the value $RL=0$ is adequate for the case of the calculation of a sequence of eigenvalues ascending from the minimal eigensolution, while the value of RR given by Gershgorin's Theorem is adequate for the calculation of a sequence of eigenvalues descending from the maximal eigensolution.

[2] $K=K+1$ (calculation of the k -th eigenvalue)

① $KK=0$ (calculation of an approximate eigenvalue)

Prepare the eigenvectors obtained by the calculations thus far $\Phi_l = [\phi_1, \phi_2, \dots, \phi_l]$, a vector b (for example $b = u_0$, where the vector u_0 , normalized as $u_0^T u_0 = 1$, is the $(l+1)$ -th randomly generated initial approximate eigenvector), λ , an eigenvalue from the Rayleigh quotient, a matrix $\tilde{A}$, and $RC = (RL + RR)/2$.

② $KK=KK+1$

Solve the simultaneous linear equations by the PCG method.

(If $\text{sign}(p^T A p)$ is negative, the computation stops. This algorithm will be introduced later.) If all of $\text{sign}(p^T A p)$ are positive, approximate eigenvectors are defined as normalized solution vectors, while if $\text{sign}(p^T A p)$ is always negative, a similar process is carried out once every couple of steps. If calculating a sequence of eigenvalues ascending from the minimal eigensolution, if all of $\text{sign}(p^T A p)$ are positive, then $RL=RC$. If one or more of $\text{sign}(p^T A p)$ is negative, then $RR=RC$. If calculating a sequence of eigenvalues descending from the maximal eigensolution, if all of $\text{sign}(p^T A p)$ are positive, then $RR=RC$. If at least one value of $\text{sign}(p^T A p)$ is negative, then $RL=RC$. Set $RC = (RL + RR)/2$. The first convergence decision is taken at this stage; if the solution is not yet converged, repeat ②.

③ $KKK=0$ (clear-up of eigensolutions)

For the initial displacement of the origin, set RL when calculating a sequence of eigenvalues ascending from the minimal eigensolution, or RR for a sequence of eigenvalues descending from the maximal eigensolution.

④ $KKK=KKK+1$

Solve the simultaneous linear equations by the PCG method. (Even if $\text{sign}(p^T A p)$ is negative, continue iterations of the PCG method.) Extract eigenvectors from the solution vectors. Normalize these eigenvectors, and calculate an approximate eigensolution by means of the Rayleigh quotient. If the approximate solution does not lie in the range

from RL to RR, then repeat ②. If calculating a sequence of eigenvalues ascending from the minimal eigensolution, if all of $\text{sign}(p^T A p)$ are positive, then RL=RC. If at least one value of $\text{sign}(p^T A p)$ is negative, then RR=RC. If calculating a sequence of eigenvalues descending from the maximal eigensolution, if all of $\text{sign}(p^T A p)$ are positive, then RR=RC. If at least one value of $\text{sign}(p^T A p)$ is negative, then RL=RC. Calculation of the sift of origin

The second convergence decision is taken at this stage; if convergence is not yet sufficient, then repeat ④.

[3] If a sufficient number of eigensolutions have not yet been obtained, return to [2].

[An Example Algorithm for the PCG Method of the Proposed Method]

[1] Calculate the initial approximate solution vector x_0 , the initial residue vector r_0 and the initial conjugate direction vector p_0 . In this numerical experiment, an incomplete inverse matrix of $\tilde{A}$ is denoted by C . In the ICCG method $C=[LDL^T]^{-1}$, while in the SCG method $C=A_d^{-1}$ (where A_d is a matrix composed of the diagonal elements of A). Computational speed at this stage can be improved by using a parallel implementation of the SCG method.

[2] In order to extract eigenvectors obtained so far, calculate

$$p_i - \sum_{m=1}^i (\phi_m^T p_i) \phi_m \quad \text{for } i=0,1,2,\dots, \text{ and define the results to be the new } p_i.$$

(Note that this process is an addition to the conventional PCG method.)

[3] Check the Changes of Signs

If $\text{sign}(d_i)$ is negative, then terminate the computation. The matrix A is non-positive definite at this point. (Note that this process is also an addition to the conventional PCG method.)

[4] Calculate α_i , x_{i+1} and r_{i+1} .

[5] Check the convergence

If $\|r_{i+1}\|/\|b\| \leq \varepsilon$ (where ε is the convergence criteria), then stop the iteration; x_{i+1} is an approximate eigenvector at this point.

[6] Calculate p_{i+1} .

[7] Go back to [2].

If A is a positive definite matrix, the necessary computational time is nearly the same as that of the PCG method. However, if A is non-positive-definite, the necessary computational time can be considerably shorter than that of the PCG method.

4 NUMERICAL EXPERIMENT

In order to verify the proposed method, numerical experiments given by the computation of standard and generalized eigenvalue problems were undertaken. Both the bisection and presently proposed methods were used to calculate approximate eigensolutions, with accuracy improved by application of the shifting inverse power and bisection methods. The classic subspace method was also used to compare results with the proposed method, since it is known that the computational time required

by this method does not differ significantly from that of the restarted-Lanczos method by the author's experience.

As described in the algorithm, first the proposed method and the bisection method were used to calculate approximate eigensolutions. The computation stopped when the residual relative error between the approximate eigenvalues of the present and previous steps reaches the value which is not greater than 10^{-4} (Ex.4.1) or 10^{-3} (Ex. 4.2) (the first convergence decision). The convergence criteria of the PCG method was therein defined as 10^{-5} (Ex. 4.1) or 10^{-4} (Ex. 4.2). Next, an eigensolution was found (the second convergence decision point). An eigensolution was found if one of the following two criteria is met. One criterion is satisfied when we have from the shifting inverse power and bisection methods based on the approximate eigensolution, that $\|\tilde{A}x - \lambda \tilde{B}x\|/\|\tilde{A}x\| \leq 10^{-8}$ where x is a approximate eigenvector. The other criterion is satisfied if the residual relative error between the approximate eigenvalues of the present and previous steps divided by the approximate eigenvalue of present step is less than 10^{-12} , for which the convergence criteria of the PCG method was set to be 10^{-11} . For the classic subspace method, an eigensolution was found if eigensolutions were calculated for which either $\|\tilde{A}x - \lambda \tilde{B}x\|/\|\tilde{A}x\| \leq 10^{-8}$, or the residual relative error between the approximate eigenvalues of the present and previous steps is less than 10^{-13} , for which the convergence criteria of the PCG method was set to be 10^{-9} .

All calculations were performed in double-precision arithmetic on an EPSON Endeavor Pro 1100 (Desktop Personal Computer, Pentium4, 1600MHz, RAM512MB, Windows XP, Fujitsu Fortran & C Academic Package Ver.3). A scalar ICCG method was used for the PCG method. Parallelization is possible upon using a parallel-implemented PCG method.

4.1 Standard Eigenvalue Problem (2-dimensional Helmholtz Problem)

This section is concerned with a standard eigenvalue problem derived from a discretized 2-dimensional Helmholtz problem defined on a

rectangular domain. Centered-differences were used in the discretization, and upon fixing the value of the solution to be 0 everywhere on the boundary, a standard eigenvalue problem $\tilde{A}x = \lambda x$ is derived, where $\tilde{A} = [\tilde{a}_{k,l}]$ and each element of $\tilde{A}$ is defined as follows:

$$l = (i-1)n_y + j,$$

$$\tilde{a}_{l,l} = \frac{2}{\Delta x^2} + \frac{2}{\Delta y^2}, \quad (1 \leq i \leq n_x, 1 \leq j \leq n_y),$$

$$\tilde{a}_{l,l+ny} = \tilde{a}_{l+ny,l} = -\frac{1}{\Delta x^2}, \quad (1 \leq i \leq n_x, 1 \leq j \leq n_y),$$

$$\tilde{a}_{l,l+1} = \tilde{a}_{l+1,l} = -\frac{1}{\Delta y^2}, \quad (1 \leq i \leq n_x, 1 \leq j \leq n_y),$$

$$\tilde{a}_{l,m} = 0, \quad l, m \text{ except for the above superscription}, \quad (16)$$

where $n_x + 1$ and $n_y + 1$ are nodal numbers in the x and y directions, respectively, and l_x and l_y are the lengths of the domain in the x and y directions, respectively, with $\Delta x = l_x / (n_x + 1)$, $\Delta y = l_y / (n_y + 1)$ (Cf. Fig.2.). The analytical solution for this coefficient matrix $\tilde{A}$ is given by the following equation:

$$\lambda_{i,j} = \frac{2}{\Delta x^2} \left(1 - \cos \frac{i\pi}{n_x + 1} \right) + \frac{2}{\Delta y^2} \left(1 - \cos \frac{j\pi}{n_y + 1} \right). \quad (17)$$

Table 1 Parameters of the plane model

type	l_x	l_y	n_x	n_y	n
A	1.0	1.0	50	50	2500
B	1.0	1.0	100	100	10000
C	1.0	1.0	150	150	22500
D	1.0	1.0	200	200	40000

The Parameters used in the numerical experiment are shown in Table 1. As shown in the table, the numerical computation was conducted for 4 combinations of n degrees of freedom, and either the 5 smallest or 5 largest eigensolutions were calculated by the proposed method and also by the classic subspace method. The number of eigensolutions calculated by the classic subspace method was determined by min (the number of eigensolutions required $\times 2$, the number of eigensolutions required + 8, n). As can be seen from Table 2, many multiple eigenvalues are present (regarding the upper eigenvalues, the largest eigenvalue is denoted by λ_1).

Figs.3-6 show the convergence of trial eigenvalues (λ_2 of Type A) and the number of iterations of the ICCG method against each step in the bisection method. For calculating the 5 smallest eigensolutions (λ_2 in Fig.3), when the matrix was non-positive definite, the number of iterations required by the ICCG method for a trial eigenvalue to become sufficiently close to the real eigenvalue was, notably, just 1 (steps 1 -5), and the

domain of existence of the eigenvalue rapidly diminished. Upon approaching the solution, less than 10 iterations were then required. When the matrix was positive definite, it was necessary to iterate the computation by the ICCG method until the convergence decision condition was satisfied, as shown in steps 8,10,12,13,17,18,20,22 and 23, where approximate solution vectors were taken as approximate eigenvectors.

Table 2 Eigenvalues from equation (17) for type A

No.	lower eigenvalues	No.	upper eigenvalues
1	19.73296781979341	1	20788.26703218021
2	49.29499259648689	2	20758.70500740351
3	49.29499259648689	3	20758.70500740351
4	78.85701737318037	4	20729.14298262682
5	98.44041935423965	5	20709.55958064576
6	98.44041935423965	6	20709.55958064576
7	128.0024441309331	7	20679.99755586906
8	128.0024441309331	8	20679.99755586906
9	166.9828226557774	9	20641.01717734422
10	166.9828226557774	10	20641.01717734422

In the case of $\lambda_3 (= \lambda_2, \text{ multiple root})$ in the plot, unlike the case of λ_2 , the value converged from the right-trial eigenvalue, and so the number of iterations required by ICCG method was very small resulting in a shorter computation time. Fig.5 shows that the proposed method, through its use of the bisection method, ensures that the interval between left- and right-trial eigenvalues steadily narrows, and the value reliably tends to the eigenvalue. Furthermore, with reference to Figs.4 and 6, it is clear that results for the case of the calculation of the 5 largest eigensolutions display similar properties. Figs.7-8 display the computational time required by the classic subspace and presently proposed methods. In calculating the 5 smallest eigensolutions, the proposed method requires a factor of 1/39~1/60 (< 40000 degrees of freedom) of the time required by the classic subspace method, while for the calculation of the 5 largest eigensolutions, the proposed method requires a factor of 1/23~1/124 (< 40000 degrees of freedom) of the time required by the classic subspace method. Furthermore, the difference between the required computational times of the two methods increases exponentially in the number of degrees of freedom.

4.2 Generalized Eigenvalue Problem (2-dimensional FEM Frame Problem)

The second example problem is the generalized eigenvalue problem arising from the application of the finite element method (FEM) to the architectural frame with rigid nodes as shown in Fig.9. In the frame, columns and girders are from reinforced concrete, whose Young's modulus and density are $2.058 \times 10^4 \text{ N/mm}^2$ and $2.4 \times 10^{-6} \text{ kg/mm}^3$,

respectively. The cross sections (width $\times$ height) of the columns and girders are 800mm $\times$ 800mm and 400mm $\times$ 800mm, respectively. The material length (floor height and span) are 3000mm and 6000mm, respectively. Each node of the frame has three degrees of freedom (u: displacement along x-axis, v: displacement along y-axis, θ : deflection angle), and the elements of coefficient matrices $\tilde{A}$ (stiffness matrix) and $\tilde{B}$ (mass matrix) have different dimensions. Furthermore, the sizes of the elements of these two matrices differ significantly. The matrix $\tilde{B}$ was defined to be a distributed concentrated mass matrix. In this numerical experiment, the 5 smallest eigensolutions, which are required from an engineering perspective, were calculated both by the proposed method and by the classic subspace method. The parameters used in the numerical experiment are given in Table 3. As shown in the table, computations were conducted for 5 combinations of n degrees of freedom. The process of searching for eigenvalues is similar to that of the previous test problem, and so the explanation is omitted here. Figs.10-11 plots the required CPU time for this problem. The proposed method requires only a factor of 1/5~1/22 of the time required by the classic subspace method, and this difference again increases with increases in the number of degrees of freedom. Thus, for this FEM derived problem, we observe a similar behavior in the performance of the proposed method as for the previous test problem.

Fig.3 Number of ICCG iterations in the bisection method for the lower 5 eigenvalues

Fig.4 Number of ICCG iterations in the bisection method for the upper 5 eigenvalues

Fig.5 Trial lower eigenvalues for λ_2

Fig.6 Trial upper eigenvalues for λ_2

Fig.7 Comparison of CPU time for the lower 5 eigenvalues

Fig.8 Comparison of CPU time for the upper 5 eigenvalues

Fig.9 A plane frame model

Table 3 Parameters of example 2

Type	m_r	n_r	n
A	16	14	720
B	32	29	2880
C	48	44	6480
D	64	59	11520
E	80	74	18000

m_r : numbers of story, n_r : numbers of span, n : degrees of freedom

Fig.10 Comparison of CPU time for the lower 5 eigenvalues for the plane frame model

Fig.11 Comparison of CPU time for the upper 5 eigenvalues for the plane frame model

5 CONCLUSIONS

It is theoretically possible to count the number of eigenvalues while conducting computations by PCG method, because the number of signs of the object matrix can be counted by $\text{sign}(p^T A p)$. If the calculation is free from numerical errors, the proposed method can yield the eigensolutions as the Householder bisection method. However, it is practically impossible to count the number of changes of signs in the neighborhood of eigenvalues due to the round off error in computation. The proposed method overcomes this problem by using equations in the eigenvectors of the quadratic form of a matrix, and by exploiting the information of whether or not the matrix is positive definite. The proposed method yields reliable eigensolutions rapidly. Moreover, while convergence is normally slower for multiple roots for other methods, by contrast, convergence of the proposed method improves under the presence of multiple roots. This result is established theoretically, and is borne out by the numerical experiments.

Since the proposed method is a PCG-based method, it is particularly effective when a sequence of either the smallest or largest eigensolutions of a large, sparse and symmetric matrix are required. Moreover, the proposed method does not find unnecessary solutions, and it thus minimizes the required computational memory capacity. In short, the proposed method is a method requiring only a relatively small computational time, is of good accuracy and is stable. Finally, it may be noted that it is also possible to accelerate the method's computational speed by parallel processing if a PCG method designed for parallelization is used.

REFERENCES

- 1) Ajiz, M.A. and Jenning, A.: A Robust Incomplete Choleski-Conjugate Gradient Algorithm, International Journal for Numerical Method in Engineering, Vol.20, pp.949-966, 1984.5
- 2) Arbenz, P. and Lehoucq, R.B.: A Comparison of Algorithms for Modal Analysis in the Absence of a Sparse Direct Method, International Journal for Numerical Method in Engineering, Vol.1, pp.1-21, 2003
- 3) Bathe, K.J.: Finite Element Procedures in Engineering Analysis, Prentice-Hall, 1982
- 4) Briggs, W.L., Henson, V.E. and McCormick, S.F.: A Multigrid Tutorial Second Edition, SIAM: Philadelphia, pp.137-161, 2001
- 5) Golub, G.H. and Underwood, R.: The Block Lanczos Method for Computing Eigenvalues, in Mathematical Software 3, Rice J (ed). Academic Press: New York, 1977
- 6) Golub, G.H. and van Loan C.F.: Matrix Computation, Johns Hopkins: Baltimore, MD, 1983
- 7) Kashiwagi, M.: A PCG Method Using Sylvester's Law of Inertia for the Eigensolution of Large, Sparse, Symmetric Matrices, Bulletin of School of Engineering, Kyushu Tokai University, Vol.25, pp.41-47, 1999.3
- 8) Kashiwagi, M., Hirai, I., Katayama, T. and Pilkey, W.D.: Sturm Sequence Recurrence Formula for Eigensolution of Locally Modified Systems, Finite Elements in Analysis and Design, Vol.12, pp.31-36, 1993.4
- 9) Komzsik, L. and Satoh, E.: The Art of the State in Large Scale, Accurate Eigenvalue Computations, Bulletin of the Japan Society for Industrial and Applied Mathematics, Vol.16, pp.58-63, 2006.6
- 10) Meijerink, J.A. and van der Vorst H.A.: Guidelines for the Usage of Incomplete Decompositions in Solving Set of Linear Equations as They Occur in Practical Problems, Journal of Computational Physics, Vol.44, pp.134-155, 1981
- 11) Nour-omid, B., Parlett, B.N. and Laylor, T.R.: Lanczos versus Subspace Iteration for Solution of Eigenvalue Problems, International Journal for Numerical Method in Engineering, Vol.19, pp.859-871, 1983.6
- 12) Paige, C.C.: Computational Variants of the Lanczos Method for the Eigenproblem, J.Inst.Maths Applies, Vol.10, pp.373-381, 1972
- 13) Sakurai, T.: A Parallel Method for Large Scale Generalized Eigenvalue Problems, Bulletin of the Japan Society for Industrial and Applied Mathematics, Vol.13, pp.28-37, 2003.12
- 14) Sleipen, G.L.G. and van der Vorst H.A.: A Jacobi-Davidson Iteration Method for Linear Eigenvalue Problems, SIAM, Vol.17, pp.401-425, 1996
- 15) Tanaka, M.: Studies on Water Wave Turbulence by Large-Scale Direct Numerical Simulations, Bulletin of the Japan Society for Industrial and Applied Mathematics, Vol.17, pp.2-15, 2007.6

和文要約

1 はじめに

コンピュータの計算速度、容量など、その性能は指数関数的に進歩して現在に至っており、計算力学などによる数値シミュレーションは実用化の時代に入ったといえる。そして、それぞれの分野の解析法は、大次元の連立1次方程式あるいは固有値問題に帰着する。現在、ソルバーとしてICCG法、SCG法等の前処理を施した共役勾配法系の反復法(PCG法)やAMG法とCG法を組み合わせた方法などが並列化と共に注目されている。これらは、消去法に比べ記憶領域を極めて少なくできるため、有限要素法などに見られる大次元でスパースな行列に対して有力な解析手段となる。

提案法はICCG法やSCG法などの前処理を施した共役勾配法系の反復法や原点移動逆ベキ乗法および2分法を利用しながら、大次元スパース行列で小さい方からあるいは大きい方から数個の固有解を求める場合に有効である。大きい方から数個の固有解を求める場合も、逆ベキ乗法で小さい方から数個の固有解を求める場合と同様に数値解析できるのが特徴である。逆ベキ乗法やサブスペース法は要求される固有解以外の解を求める必要があるが、提案法はその必要がないので、計算機の記憶容量を最小限に押さえられると共に、正定値性と非正定値性を利用した方法の採用および固有解以外の解を求める必要がないことにより、計算時間の大幅な短縮化も可能としている。多重根に対しては収束が遅いのが一般的であるが、提案法は逆に早くなる特徴も持っている。また、固有解は小さい方からあるいは大きい方から順番に確実に求められるので安定性を保持した解法ともなっている。

2 共役勾配法による固有解

一般固有値問題は $[\tilde{A} - \lambda \tilde{B}] \phi = 0$ によって与えられる。ここに、 $\tilde{A}$ は $n \times n$ の実対称行列で、 $\tilde{B}$ は $n \times n$ の正値実対称行列であり、 λ は固有値で、 ϕ は対応する固有ベクトルである。シルベスター慣性則を応用したPCG法により行列 A の符号数が決定される。しかし、数値計算で生じる丸め誤差のために、極小次元問題の場合を除いて、共役方向ベクトルに関する行列 A の共役性が演算の進行に伴いずれていく。そのために、提案法によって計算された試行固有値が正しい固有値に接近するとき、符号変化数が正しく求まらない。この不安定性を避ける方法を提案した。最小固有解からいくつかの固有解を求める場合と最大固有解からいくつかの固有解を求める場合について、提案法の考え方を示した。

3 アルゴリズム

提案法のアルゴリズムの1例を本文に示した。

4 数値実験

提案法の検証のために、本法と2分法を使い、近似固有解を求めた後、原点移動逆ベキ乗法と2分法を使って、固有解の精度を向上させる手順により、標準固有値問題および一般固有値問題に対する数値実験を行った。また、提案法との比較のために、サブスペース法を使用した。

最初に、長方形領域における2次元ヘルムホルツ方程式を離散化して生じる標準固有値問題を扱う。Figs.3-6は2分法におけるステップ毎のICCG法の反復回数と試行固有値の収束状況を表している。

小さい方から5個の固有解を求める場合、非正定値行列の時、試行固有値が実際の固有値に近づくまでは、ICCG法の反復回数は1回(ステップ1から5)と極端に少なく、急激に固有値の存在範囲は狭まっている。解に近づいても10回以内と非常に少ない。正定値行列の時、収束判定値を満足するまでICCG法は反復を行っている。Fig.5から、提案法は、2分法を利用することによりleft(左試行固有値)とright(右試行固有値)が次第に狭まり、確実に固有値に収束している。また、Figs.4と6から、大きい方から5個の固有解を求める場合も、小さい方から5個の固有解を求める場合と同様な傾向を示した。Figs.7-8はサブスペース法とのCPU時間の比較を描いているが、小さい方から5個の固有解を求める場合、サブスペース法のおよそ1/39~1/60(<40000自由度)の時間で、また、大きい方から5個の固有解を求める場合、サブスペース法のおよそ1/23~1/124(<40000自由度)の時間で計算されており、自由度の増加に伴いその差は指数関数的に大きくなる傾向を示した。

2番目の計算例は、Fig.9に示すように剛節節点を有する建築骨組についての有限要素法(FEM)による一般固有値問題である。ここでは、工学系で必要とされる小さい方から5個の固有解を提案法とサブスペース法の両方法によって計算した。数値実験に用いたパラメーターをTable3に示す。表に示されるように、5通りの自由度に対して数値計算を行った。固有値の探索は最初の計算例と同様な性状を示しており、ここではその説明を省いた。Figs.10-11はサブスペース法とのCPU時間の比較を描いているが、サブスペース法のおよそ1/5~1/22の時間で計算されており、最初の計算例と同様に自由度の増加に伴いその差は大きくなることを示した。FEMにおいても、同様な傾向を示すものと思われる。

5 結語

理論的には、 $\text{sign}(p^T A p)$ によって対象とする行列の符号数を計算できるので、PCG法の計算をしながら固有値の個数をカウントできる。演算時の誤差の影響を受けなければ、ハウスホルダー2分法のように固有解を求め得る方法であるが、丸め誤差により固有解の近傍では符号変化数を正しくカウントできない場合がほとんどである。その対処法として、行列の2次形式の固有ベクトルによる表現式および行列の正定値性と非正定値性を利用して確かかつ迅速に固有解を求める方法を提案した。多重根を持つ行列に対して収束は遅くなるのが一般的であるが、提案法の理論と数値実験により、特に多重根を持つ行列の方がそうでない行列より早く収束することが示された。本法はPCG法系の解法であるので、大次元スパース行列で小さい方あるいは大きい方から数個の固有解を必要とする場合に有効であり、要求される固有解以外の解を必要としないので、計算機の記憶容量を最小限に押さえられる、計算時間も短い、精度も良い、安定性のある解法と考えられる。また、並列用PCG法を用いれば並列処理による高速化も可能である。

(2007年11月26日原稿受理, 2008年4月2日採用決定)