AN ALTERNATIVE SUBSPACE ITERATION METHOD USING DYNAMIC CONDENSATION MATRIX

Yangki Jung and Zu-Qing Qu

Bell 4190, University of Arkansas, Fayetteville, AR, 72701
Phone: 501-575-4628; Fax: 501-575-7168
E-mail: yjung @ engr.uark.edu

ABSTRACT

An alternative subspace iteration method is developed to reduce the computational effort of the classical subspace iteration. The proposed subspace consists of a dynamic condensation matrix and an identity matrix. The dynamic condensation matrix is defined as the relations of the eigenvectors or responses between two sets of degrees of freedom, called the master and the slave degrees of freedom. Using this new subspace, the Rayleigh-Ritz becomes partially unnecessary during the iteration. Another advantage of the present approach is that the reduced matrices defined in the subspace have their physical meaning. They can be used in the finite element modeling, test-analysis model correlation, etc.

NOTATION

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>$K$</td>
<td>stiffness matrix</td>
</tr>
<tr>
<td>$M$</td>
<td>mass matrix</td>
</tr>
<tr>
<td>$\Lambda$</td>
<td>diagonal matrix with the system eigenvalues</td>
</tr>
<tr>
<td>$\lambda_i$</td>
<td>$i$th eigenvalue</td>
</tr>
<tr>
<td>$\Phi$</td>
<td>eigenvector matrix of the projected model</td>
</tr>
<tr>
<td>$Q$</td>
<td>eigenvector matrix</td>
</tr>
<tr>
<td>$\phi_i$</td>
<td>$i$th eigenvector</td>
</tr>
<tr>
<td>$I$</td>
<td>unit matrix</td>
</tr>
<tr>
<td>$X$</td>
<td>subspace vector</td>
</tr>
<tr>
<td>$R$</td>
<td>dynamic condensation matrix</td>
</tr>
<tr>
<td>$X^{(i+j)}_{nm}$</td>
<td>subspace at the master degree of freedom</td>
</tr>
<tr>
<td>$X^{(i+j)}_{sm}$</td>
<td>subspace at the slave degree of freedom</td>
</tr>
</tbody>
</table>

INTRODUCTION

The eigensolution method has been very important in a dynamic study. Since the prediction accuracy of structural dynamic characteristics is required, the number of degree of freedom applied in finite element analysis has rapidly become larger. This causes accordingly high computational cost and memory size requirement. To solve these problems, many solution methods have been developed for eigenvalue analysis.

Among these methods the classical subspace iteration method (Bathe [1], Bathe and Wilson [2]) has well been known to be very efficient for solving large eigenvalue problem. The subspace iteration method is used for the computation of a few smallest eigenvalues and the corresponding eigenvectors of large eigenproblems. They developed the subspace iteration method based on the Rayleigh-Ritz analysis and inverse iteration. Convergence was greatly improved since one of the benefits of using the inverse iteration method was that it transformed the poorly separated eigenvalues into well separated eigenvalues. Also, memory size requirement is greatly reduced.

The main goal of the subspace iteration method is to solve for the smallest $p$ eigenvalues and the corresponding eigenvectors satisfying the generalized eigenproblem:

$$K\Phi = M\Phi \Lambda$$

where $K$ and $M$ are the stiffness and mass matrices of the discrete degree of freedom system,
\( \Lambda \) represents a diagonal matrix with the system eigenvalues and the columns of \( \Phi \) are the corresponding eigenvectors. The matrices \( K \) and \( M \) are positive definite and symmetric. If the order of \( K \) and \( M \) is \( n \), the \( n \) eigenpairs are ordered such as:

\[
0 < \lambda_1 \leq \lambda_2 \leq \ldots \leq \lambda_n
\]

\( \phi_1, \phi_2, \phi_3, \ldots, \phi_n \)

However, when the number of eigenvalues is large, the convergence of the required eigenvalues can be slow and the computational effort can be high. Also, if the starting iteration vectors are poorly selected, some eigenvalues and eigenvectors might be missed. For overcoming the drawbacks of the classical subspace iteration method, various schemes have been developed such as over-relaxation method, shifting, improvement of starting vector, matrix power acceleration, selective repeated inverse iteration, Chebyshev polynomials, substructures and parallel computing.

Bathe et al. [3], Alk et al. [4] and Rajendran et al. [5] employed over-relaxation method with the subspace iteration method. Bathe et al. [3] combined over-relaxation and shifting techniques. They concluded that this accelerated method is practically more effective than the basic subspace iteration method when the basic subspace iteration method requires many iterations. Also, they reported that this method could be effectively applied to the solution of eigenproblems where the matrices have small or large bandwidths. Alk et al. [4] showed that over-relaxation method could be applied effectively to solve eigenproblems. They showed that the method could reduce 40% in cost of the basic subspace iteration for eigenvalues. Shifting has been used to accelerate the convergence of other eigensolving procedure. Wilson et al. [6] discussed that the strategies of the shifting relied upon establishing stable shifting routines and avoiding the numerical difficulties which may arise through a poor choice of shife point. Cheu et al. [7] investigated the effects of starting iteration vectors accelerating the convergence rate of the basic subspace iteration method. They derived alternative starting iteration vectors from the Guyan reduction method [8]. Cheu et al. [7] showed the improved results for the reduced eigenproblems. Qian and Dhatt [9] studied matrix power acceleration. The selective repeated inverse iteration is proposed by Lam and Bertolini [10]. Both the matrix power acceleration and the selective repeated inverse iteration perform the inverse iteration twice for each Rayleigh-Ritz step. The main difference between these two methods is that the matrix power acceleration uses all trial vectors, but the selective repeated inverse iteration uses partial trial vectors. Yamamoto and Ohtsubo [11] applied Chebyshev polynomials to accelerate the subspace iteration method and showed the improved convergence of the subspace iteration. Jasbir et al. [12] and Xian et al. [13] investigated a simple and efficient method based on the subspace iteration method with substructures. Jasbir et al. [12] presented the subspace iteration method with partitioning a large structure into several substructures. They mentioned that this method was quite compact and required minimum computer storage because it does not require the generation and storage of stiffness and mass matrices for entire structure. Since developing a high efficient computer, the numerical methods using parallel computers have been studied. Zhang and Moss [14] implemented a parallel computing scheme to accelerate the subspace iteration method.

In this paper, a new alternative subspace iteration method is proposed by the repeated inverse iteration with the basic subspace iteration. The Rayleigh-Ritz analysis is not needed in each step in the proposed method. The theory and concept of this method are discussed. Numerical example is presented to verify the effectiveness of this technique.

**CLASSICAL SUBSPACE ITERATION**

The basic subspace iteration method proposed by Bathe [1] combines simultaneous inverse iteration and Rayleigh-Ritz analysis. The basic objective of this method is to solve for the lowest \( p \) eigenvalues and corresponding eigenvectors satisfying equation (1). Also, the orthogonality condition must be satisfied

\[
\Phi^T K \Phi = \Lambda \quad \text{and} \quad \Phi^T M \Phi = I
\]

(2)

where \( I \) is an unit matrix of order \( p \).

The subspace iteration method entails iterating simultaneously with \( p \) linearly independent trial vectors that initially span the starting subspace \( E^{(1)} \), until subspace \( E^{(\infty)} \) is spanned to sufficient accuracy. The following algorithm is used to perform the subspace iterations;

First, determine the starting vector \( X^{(1)} \).

For \( i = 1, 2, \ldots \) iterate from subspace \( E^{(i)} \) to subspace \( E^{(i+1)} \):

\[
K \overline{X}^{(i+1)} = MX^{(i)}
\]

(3)

Determine the projections of the matrices \( K \) and \( M \) in the subspace \( E^{(i+1)} \):

\[
K^{(i+1)} = \overline{X}^{T(i+1)} K \overline{X}^{(i+1)}
\]

(4)
Compute for the $q \times q$ eigensystem of the projected matrices:

$$
M^{(i+1)} = \overline{X}^{T(i+1)} M \overline{X}^{(i+1)}
$$

(5)

Construct an improved approximation for the eigenvectors from $\overline{X}^{(i+1)}$ using the $n \times q$ matrix of Ritz trial vectors, and the $q \times q$ projected system eigenvectors $Q^{(i+1)}$, that is,

$$
X^{(i+1)} = \overline{X}^{(i+3)} Q^{(i+1)}
$$

(6)

Then, if the vector $X^{(i)}$ are not orthogonal to one of the required eigenvectors, $\Lambda^{(i+1)}$ converges to $\Lambda$ and $X^{(i+1)}$ converges to $\Phi$ as $k \to \infty$.

Convergence is measured by the eigenvalue approximations (Bathe [1]). Arranging $\lambda_j^i$ and $\lambda_{i-1}^j$, $j = 1, 2, \ldots, q$ in the increasing order of magnitude, the quantities $tolc_j$ can be calculated as

$$
tolc_j = \left| \frac{\lambda_j^i - \lambda_{i-1}^j}{\lambda_j^i} \right| < tol, \quad j = 1, 2, \ldots, q
$$

(8)

If $tolc_j$ is less than allowable tolerance, $tol$, for every $j$, then the convergence is reached. The allowable tolerance is typically equal to $10^{-6}$. This results in a stable eigensolution. Table 1 shows the algorithm of the classical subspace iteration.

**ALTERNATIVE SUBSPACE ITERATION**

When the number of eigenvalues is large, the Rayleigh-Ritz analysis in each step requires great computational effort. Alternative subspace iteration is proposed by combining the dynamic condensation with the basic subspace iteration. The dynamic condensation matrix is defined as the relations of the eigenvectors or responses between two sets of degrees of freedom, called the master and the slave degrees of freedom. Using this alternative subspace, the Rayleigh-Ritz becomes unnecessary during the iteration.

The following algorithm is used to perform the alternative subspace iteration:

Before the subspace iteration, a set of master degree of freedom needs to be selected. The starting subspace consists of an identity matrix at the master degree of freedom a null matrix at the slave degree of freedom. That is

### Table 1. Classical subspace iteration algorithm

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>a)</td>
<td>Set up starting iteration vectors $X^{(1)}$</td>
</tr>
<tr>
<td>b)</td>
<td>Factorize the stiffness matrix $K : K = LDL^T$</td>
</tr>
<tr>
<td>c)</td>
<td>Main subspace iteration loop</td>
</tr>
<tr>
<td></td>
<td>For $i = 1, 2, \ldots, m$</td>
</tr>
<tr>
<td></td>
<td>1. Compute $Y^{(i)} = MX^{(i)}$</td>
</tr>
<tr>
<td></td>
<td>2. Solve $K \overline{X}^{(i+1)} = Y^{(i)}$</td>
</tr>
<tr>
<td></td>
<td>3. $K^{(i+1)} = \overline{X}^{T(i+1)} K \overline{X}^{(i+1)}$</td>
</tr>
<tr>
<td></td>
<td>4. $M^{(i+1)} = \overline{X}^{T(i+1)} M \overline{X}^{(i+1)}$</td>
</tr>
<tr>
<td></td>
<td>5. Solve $K^{(i+1)} Q^{(i+1)} = M^{(i+1)} Q^{(i+1)} \Lambda^{(i+1)}$</td>
</tr>
<tr>
<td></td>
<td>6. $X^{(i+1)} = \overline{X}^{(i+3)} Q^{(i+1)}$</td>
</tr>
<tr>
<td></td>
<td>7. $\left</td>
</tr>
</tbody>
</table>

If the convergence criterion is not satisfied goto step 1
\[ X^{(1)} = \begin{bmatrix} I_{mm} \\ 0 \end{bmatrix} \] (9)

In practice the arrangement that all the master degrees of freedom come first and then the slave degrees of freedom is unnecessary.

Factorize the stiffness matrix using the modified Cholesky decomposition

\[ K = LDL^T \] (10)

For \( i = 0, k, 2k, 3k, \ldots \), begin the iteration.

For \( j = 1, 2, 3, \ldots, k \) perform the following iteration:

Computer a set of new subspace \( X_m^{(i+j)} \) using simultaneous inverse iteration, i.e.,

\[ K\hat{X}_m^{(i+j)} = MX_m^{(i+j-1)} \] (11)

Construct the submatrix \( \hat{X}_{mm}^{(i+j)} \) using the values of matrix \( \hat{X}_m^{(i+j)} \) on the master degrees of freedom and computer the inverse of the submatrix.

\[ \hat{X}_m^{(i+j)} = \begin{bmatrix} \hat{X}_{mm}^{(i+j)} \\ \hat{X}_{sm}^{(i+j)} \end{bmatrix} \] (12)

Compute the new subspace for the next iteration:

\[ X_m^{(i+j)} = \hat{X}_m^{(i+j)} \left( \hat{X}_{mm}^{(i+j)} \right)^{-1} \] (13)

The dynamic condensation matrix \( R^{(i+j)} \) may be obtained by reconstructing the new subspace using the rows at the slave degrees of freedom.

---

**Table 2 Alternative Subspace iteration algorithm**

<table>
<thead>
<tr>
<th>Step</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>a)</td>
<td>Set up starting iteration vectors ( X^{(1)} )</td>
</tr>
<tr>
<td>b)</td>
<td>Factorize the stiffness matrix ( K ): ( K = LDL^T )</td>
</tr>
<tr>
<td>c)</td>
<td>For ( i = 0, k, 2k, 3k, \ldots ), begin the iteration:</td>
</tr>
<tr>
<td>1.</td>
<td>For ( j = 1, 2, \ldots, k )</td>
</tr>
<tr>
<td>1.1</td>
<td>Compute a set of new subspace ( X_m^{(i+j)} )</td>
</tr>
<tr>
<td>1.2</td>
<td>( K\hat{X}_m^{(i+j)} = MX_m^{(i+j-1)} )</td>
</tr>
<tr>
<td>1.3</td>
<td>Construct the submatrix ( \hat{X}_{mm}^{(i+j)} ) using the values of matrix ( \hat{X}_m^{(i+j)} )</td>
</tr>
<tr>
<td>1.4</td>
<td>( \hat{X}<em>m^{(i+j)} = \begin{bmatrix} \hat{X}</em>{mm}^{(i+j)} \ \hat{X}_{sm}^{(i+j)} \end{bmatrix} )</td>
</tr>
<tr>
<td>1.5</td>
<td>Compute the new subspace for the next iteration:</td>
</tr>
<tr>
<td>1.6</td>
<td>( X_m^{(i+j)} = \hat{X}<em>m^{(i+j)} \left( \hat{X}</em>{mm}^{(i+j)} \right)^{-1} )</td>
</tr>
<tr>
<td>2.</td>
<td>Computer the projections of the stiffness matrix ( K ) and mass matrix ( M )</td>
</tr>
<tr>
<td>2.1</td>
<td>( K_R^{(i+k)} = \left( X_m^{(i+k)} \right)^T KK X_m^{(i+k)} )</td>
</tr>
<tr>
<td>2.2</td>
<td>( M_R^{(i+k)} = \left( X_m^{(i+k)} \right)^T XM X_m^{(i+k)} )</td>
</tr>
<tr>
<td>3.</td>
<td>Solve</td>
</tr>
<tr>
<td>3.1</td>
<td>( K_R^{(i+k)} Q_R^{(i+k)} = M_R^{(i+k)} Q_R^{(i+k)} \Lambda^{(i+k)} )</td>
</tr>
<tr>
<td>4.</td>
<td>( \frac{</td>
</tr>
</tbody>
</table>
After k iteration, compute the projections of the stiffness matrix \( K \) and mass matrix \( M \) in the subspace spanned by \( X_m^{(i+k)} \):

\[
K_R^{(i+k)} = (X_m^{(i+k)})^T K X_m^{(i+k)} \quad (14)
\]

\[
M_R^{(i+k)} = (X_m^{(i+k)})^T M X_m^{(i+k)} \quad (15)
\]

Compute for the \( q \times q \) eigensystem of the projected matrices:

\[
K_R^{(i+k)} Q^{(i+k)} = M_R^{(i+k)} Q^{(i+k)} \Lambda^{(i+k)} \quad (16)
\]

Convergence is measured by the eigenvalue approximations. Arranging \( \lambda_j^{(i+k)} \) and \( \lambda_j^{(i)} \), \( j = 1,2,...,k \) in the increasing order of magnitude, the quantities \( \text{tolc}_j \) can be calculated as

\[
\text{tolc}_j = \left| \frac{\lambda_j^{(i+k)} - \lambda_j^{(i)}}{\lambda_j^{(i)}} \right| < \text{tol}, \quad (j = 1,2,...,p < m) \quad (17)
\]

If the former \( p \) eigenvalues converge, exit this loop. If \( \text{tolc}_j \) is less than allowable tolerance, \( \text{tol} \), for every \( j \), then the convergence is reached. The allowable tolerance is typically equal to \( 10^{-6} \). Table 2 shows the algorithm of the alternative subspace iteration method.

**NUMERICAL EXAMPLE**

The three-dimensional framed structure and the simply supported square plate used from Jung et al. [14] are applied to verify the effectiveness of the alternative subspace iteration method. The first smallest 15 eigenvalues are calculated by the alternative subspace iteration method. The computational times of these eigenvalues calculated by this method are compared with those of the classical subspace iteration method. Sun Microsystems SPARCstation 20 is used.

Figure 1 shows the geometric configuration of the framed structure and plate. The structure is discretized by using 315 beam elements with a total of 810 degrees of freedom. The material properties are: \( A = 0.2787 \ m^2 \), \( E = 2.068 \times 10^{10} \) Pa, \( G = 7.954 \times 10^9 \) Pa, \( I_x = 8.631 \times 10^{-3} \ m^4 \), \( I_y = 8.631 \times 10^{-3} \ m^4 \), \( I_z = 8.631 \times 10^{-3} \ m^4 \), \( \rho = 5.154 \times 10^2 \) kg/m³.

\( m = 20 \) master degrees of freedom are implemented. The first 15 smallest eigenvalues calculated by using the alternative subspace are presented in table 3. Table 4 shows the comparison of the computational times from these two methods. \( k \) of 1,2, and 5 are used for the alternative iteration method and 10 and 20 iterations are applied for both methods. When \( k \) is 1 for 10 and 20 iterations, the computational times from the proposed method are very similar to that from the classical subspace iteration method, because the proposed method requires the Rayleigh-Ritz for every iteration as same as the classical subspace method requires. However, when \( k \) are 2 and 5 for 10 iterations, the computation times from the proposed method are smaller than that from the classical subspace iteration. The proposed method does not need the Rayleigh-Ritz for every iteration. As shown in table 4, the proposed method provides good improvement compared with the classical subspace iteration method. The proposed method has yielded about 40% reduction in computational time for this problem.
CONCLUSION

An efficient method for improving the basic subspace iteration method in the calculation of the smallest eigenvalues and corresponding eigenvectors of generalized eigenproblems has been introduced. The dynamic condensation matrix is defined as the relations of the eigenvectors or responses between two sets of degrees of freedom, called the master and the slave degrees of freedom. Using this alternative subspace iteration method, the Rayleigh-Ritz becomes unnecessary during the iteration. This method has been implemented and the effectiveness of the method has been demonstrated for the numerical example. The proposed method has yielded about 40% reduction in computational time for the demonstrative problem. Based on the theory, the increase in solution effectiveness depends on the reduction of computing the Rayleigh-Ritz. Therefore, it is concluded that the alternative subspace iteration approach can provide more effective than the basic method. Future research on the stability of this approach is under way.

REFERENCE