Interaction and Multiscale Mechanics, Vol. 1, No. 2 (2008) 211-230
DOI: http://dx.doi.org/10.12989/imm.2008.1.2.211

Advances in solution of classical generalized eigenvalue problem

P. Chen*, S. L. Sun‡, Q. C. Zhao, Y. C. Gong, Y. Q. Chen and M. W. Yuan‡†

LTCS, Department of Mechanics and Aerospace Engineering, College of Engineering, Peking University, Beijing, 100871, China

(Received July 1, 2007, Accepted May 12, 2008)

Abstract. Owing to the growing size of the eigenvalue problem and the growing number of eigenvalues desired, solution methods of iterative nature are becoming more popular than ever, which however suffer from low efficiency and lack of proper convergence criteria. In this paper, three efficient iterative eigenvalue algorithms are considered, i.e., subspace iteration method, iterative Ritz vector method and iterative Lanczos method based on the cell sparse fast solver and loop-unrolling. They are examined under the mode error criterion, i.e., the ratio of the out-of-balance nodal forces and the maximum elastic nodal point forces. Averagely speaking, the iterative Ritz vector method is the most efficient one among the three. Based on the mode error convergence criteria, the eigenvalue solvers are shown to be more stable than those based on eigenvalues only. Compared with ANSYS's subspace iteration and block Lanczos approaches, the subspace iteration presented here appears to be more efficient, while the Lanczos approach has roughly equal efficiency. The methods proposed are robust and efficient. Large size tests show that the improvement in terms of CPU time and storage is tremendous. Also reported is an aggressive shifting technique for the subspace iteration method, based on the mode error convergence criteria. A backward technique is introduced when the shift is not located in the right region. The efficiency of such a technique was demonstrated in the numerical tests.

Keywords: aggressive shifting; generalized eigenvalue problem; finite element analysis.

1. Introduction

In the Finite Element Analysis (FEA) of engineering problems, some of the lowest eigenpairs of the following generalized eigenvalue problem are often required:

$$\mathbf{K}\varphi - \lambda\mathbf{M}\varphi = 0 \tag{1}$$

For eigenvalue problems with the total number of degrees of freedom greater than 10, 000, the subspace iteration, Ritz vector and Lanczos methods are recognized as the most efficient algorithms of solution. Although most commercial FEA packages offer different algorithms for the solution of eigenvalue problems, the results in general are not satisfactory due to the following problems:

1. Missing of eigenvalues;

* E-mail: chenpu@pku.edu.cn
‡ Corresponding Author, Email: sunsl@mech.pku.edu.cn
‡† E-mail: yuanmw@pku.edu.cn

2. Generation of false eigenvalues;
3. Generation of incorrect free modes; and
4. Convergence is achieved for the eigenvalues, but not for the eigenvectors.

Traditionally, the solution of Eq. (1) depends on the solution method used for the following equation:

$$(\mathbf{K} - \mu\mathbf{M})x = \mathbf{L}\mathbf{D}\mathbf{T}^T x \tag{2}$$

In general, the $\mathbf{L}\mathbf{D}\mathbf{T}^T$ decomposition of the shifted matrix $\mathbf{K} - \mu\mathbf{M}$ consumes the major portion of computing time, say 70 to 90%. In this paper a well developed cell storage sparse solution strategy is incorporated, instead of using the conventional skyline scheme. Very high efficiency is achieved using such an approach.

The subspace iteration method emerged in the 1970s as a new method for the solution of generalized eigenvalue problem. At that time, the size of structures considered is generally less than 10,000 degrees of freedom and the number of the lowest eigenpairs required is less than 20. In the 1980s, some direct solution algorithms appeared, among which the most efficient two are the Lanczos method and WYD Ritz vector method proposed by Wilson *et al.* (1982). The subspace iteration method was considered rather inefficient, even though it is generally stable and accurate. Nowadays, the size of structures considered is in the scale of millions degrees of freedom and even larger, and the number of eigenpairs required can be 200 or even more. For such problems, the accuracy of the eigenvalues and eigenvectors desired cannot be guaranteed by the direct methods. This is the main reason for the iterative methods to come back.

If the number of eigenpairs required is less than 20, direct methods such as the Ritz vector and Lanczos methods can be used with high efficiency. By average, they can show a speed of 4~10 times that of the subspace iteration method, see Gong *et al.* (2005). However, the convergence criteria for the standard Ritz vector and Lanczos are only given in terms of eigenvalues, which sometimes cause problems.

Conventionally, the convergence criteria for the subspace iteration is given in terms of the relative error for the adjacent iterations:

$$\frac{\left|\lambda_k^{(l)} - \lambda_k^{(l+1)}\right|}{\left|\lambda_k^{(l+1)}\right|} \le \varepsilon_\lambda \tag{3}$$

Meanwhile, the Lanczos method uses the following inequality as the convergence criterion:

$$\left|\lambda_k^{-1} - \nu_i\right| \le \left|\beta_q s_{qi}\right| < \varepsilon_\nu \tag{4}$$

In practice, if the tolerances are chosen as $\varepsilon_\lambda = \varepsilon_\nu = 10^{-5}$, the accuracy of part of the eigenvectors may not be satisfied by both the subspace iteration and Lanczos methods, and especially for the latter.

Improved results can be obtained if the following mode-error criteria are used:

$$\frac{\left|\mathbf{K}\tilde{\varphi}_k - \tilde{\lambda}_k\mathbf{M}\tilde{\varphi}_k\right|}{\left|\mathbf{K}\tilde{\varphi}_k\right|} \approx \frac{\left|\mathbf{K}\tilde{\varphi}_k - \tilde{\lambda}_k\mathbf{M}\tilde{\varphi}_k\right|}{\left|\tilde{\lambda}_k\mathbf{M}\tilde{\varphi}_k\right|} \le \varepsilon_\varphi \tag{5}$$

The physical meaning for the preceding criterion is as follows. $\mathbf{K}\varphi_k$ represents the maximum elastic nodal forces corresponding to the eigen mode φ_k, and $\lambda_k\mathbf{M}_k\varphi_k$ the associated maximum inertial nodal forces. The left side of Eq. (5) is the ratio of the unbalanced nodal forces to the maximum elastic nodal forces, and the middle term of (5) is the ratio of the unbalanced nodal forces to the maximum inertial forces (see Bathe and Ramaswamy 1980).

Many numerical examples showed that using Eq. (5) as the convergence criterion, more stable convergence characteristics can be achieved. With such a consistent mode error criteria, the efficiency of all the three algorithms, i.e., iterative Ritz vector, iterative Lanczos and subspace iteration, are generally similar. The iterative Ritz vector is the fastest, and the subspace iteration the slowest. A comparison with the subspace iteration in ANSYS version 8.1 and blocked Lanczos is given by Gong *et al*. (2005).

The classical subspace iteration method (SIM) was developed and so named by Bathe and Ramaswamy (1980), and Bathe (1996). This method simultaneously combines the inverse iteration method and a Rayleigh-Ritz procedure on the subspace as outlined in Appendix. The aim of the subspace iteration method is to find the p lowest eigenpairs for the general eigenvalue problem in Eq. (1), where the real matrices **K** and **M** are assumed to positive definite and positive semi-definite, respectively and (λ_i, φ_i) is the i^{th} eigenpair. If the order of **K** and **M** is N, i.e., **K**, **M** $\in R^{N\times N}$, there are N eigenpairs as follows:

$$0 < \lambda_1 \le \lambda_2 \le \lambda_3 \le \ \dots \ \le \lambda_N$$

$$\varphi_1, \ \varphi_2, \varphi_3, \dots, \ \varphi_N \tag{6}$$

In order to speed up the efficiency of the subspace iteration method, many techniques have been developed in the past two decades, namely, the shifting techniques by Bathe and Ramaswamy (1980), Chebyshev polynomials by Yamamoto and Ohtsubo (1976), over-relaxation method by Akl *et al*. (1979, 1982), multiple inverse iteration steps by Bertolini and Lam (1995), alternative projection by Rajendran and Liew (2003), Newton-Raphson acceleration by Lee and Kim (2005), etc. Among these techniques, the shifting technique corresponding to constant size subspace deserves special mentioning. Without the shifting technique, it is difficult to extract a relatively large number of eigenpairs. In our experience of computation for the 200 lowest eigenvalues of Harwell-Boeing automobile shock absorber assembly model, the classical supspace iteration method (SIM) without shifting showed very poor efficiency. The CPU time recorded for SIM without shifting was 232.84 sec compared with 38.73 sec for the case with shifting. However, quite a few improvements on SIM ignored the shifting technique because of its associated computational cost. In fact, the shifting technique is no longer expensive for applications to structures with up to 300,000 degrees of freedom, due to the fact that the property of sparse matrix in finite element analysis has been utilized for 15 years.

The shifting technique is to replace an inverse iteration

$$\mathbf{K}\bar{X}_k \ = \ \mathbf{M}X_{k-1} \tag{7}$$

by a shifted inverse iteration

$$(\mathbf{K} - \mu\mathbf{M})\bar{X}_k \ = \ \mathbf{M}X_{k-1} \tag{8}$$

In the subspace iteration method, the best choice for the convergence criterion of eigenvalues is the computable error bound proposed by Matthies based on an estimatd Rayleigh quotient of the approximated eigenvalue with its expression in the subspace, which guarantees convergence of both eigenvalue and eigenvector and is computable with minimal operations. However, its original form cannot be used for the case of nonzero shift with positive semi-definite mass matrix. In this paper, the original form without shift is generalized to nonzero shift. This generalized error bound can also be used as a convergence criterion of eigenpairs with nonzero shift.

It has been demonstrated by Wilson and Itoh (1983) that the rate of convergence for the progressive evaluation of eigenpairs by using a constant size subspace and Gram-Schmidt orthogonalization can be greatly improved through application of an optimum shifting strategy. The iteration with shifting not only produces several converged eigenvalues, but also allows to estimate the next group of eigenvalues desired (see Wilson and Itoh 1983). However, the shifting strategies proposed by Bathe and Ramaswamy (1980) and Wilson and Itoh (1983) place the shift at the left of not-yet-converged eigenvalues, which thus take advantage of only the right side of the spectrum transformation. Ideally, a more aggressive approach is to place a shift into the region of the next group of eigenvalues desired so as to yield rapid convergence. However, such an aggressive shifting strategy may encounter troubles if the traditional convergence criterion, i.e., relative difference of eigenvalues of consecutive iterations, is used. Firstly, the ghost eigenvalue(see Hughes 1987) may be encountered during iteration. Secondly, the shift may be located beyond the range of convergence of lower eigenvalues in the group, resulting in missing of eigenvalues. In this paper, we shall present an aggressive shift with a convergence criterion based on the computable error bound.

2. Three iterative algorithms and their comparisons

2.1 Subspace Iteration

Subspace iteration is an extension of the inverse power method, which was first proposed by Clint and Jennings (1970). Then Bathe and Wilson (1972) incorporated the Rayleigh-Ritz procedure in the subspace to improve the convergence speed. The steps are as follows:

I. Initialization
   1. Determine the dimension of the subspace, q
   2. Select an initial vector matrix $\mathbf{X} \in R^{N \times q}$
   3. Set the maximum number of iterations for each shifting, $I_{\max}$

II. Shifting and Sturm Sequence Check
   1. Calculate the shifting μ, ensure that μ is not the eigenvalue
   2. Decomposite $\mathbf{K} - \mu\mathbf{M} = \mathbf{LDL}^T$
   3. Perform Sturm sequence check

III. Performing $I_{\max}$ iterations, go to I
   1. M-orthonormalize $\mathbf{X}$
   2. Solve $\mathbf{X}_1 = (\mathbf{LDL}^T)^{-1}\mathbf{MX}$
   3. Calculate the projects of $\mathbf{K}$ and $\mathbf{M}$ on $\mathbf{X}_1$, $\mathbf{K}^* = \mathbf{X}_1^T\mathbf{KX}_1$, $\mathbf{M}^* = \mathbf{X}_1^T\mathbf{MX}_1$
   4. Solve the q by q generalized eigenvalue problem $\mathbf{K}^*\mathbf{\Phi}^* = \mathbf{M}^*\mathbf{\Phi}^*\mathbf{\Lambda}^*$
   5. Form the updated eigenvectors $\mathbf{X} = \mathbf{X}_1\mathbf{\Phi}^*$
   6. Judge the convergence based on the criterion of "mode-error", remove the converged eigenvectors and add the random vectors in $\mathbf{X}$ or reduce the size of subspace

As it is usual with the subspace iteration method, the dimension of the subspace is assumed to be q, and q initial vectors are assumed to start the iteration simultaneously. If p convergent eigenvectors are required, then q is taken as $q = \min(2p, p + 8)$. If p is very large, this cannot be realized. Experience shows that we can take $q = \max(\sqrt{s}, 4)$, here s is the average number of the nonzeroes in one row of $\mathbf{L}$, which can be determined by taking steps II and III into account.

2.2 Iterative Ritz vector method

The iterative Ritz vector method was proposed by Wilson *et al*. (1982), and also named the WYD-Ritz vector method. Initially, this method was used to get a better basis than eigenvectors for the direct dynamic response. Later, it was used by Yuan *et al*. (1989) to calculate the eigenvectors of large systems, and demonstrated to be efficient for solving eigenpairs.

By introducing iteration to the Ritz vector method, the accuracy of eigenpairs can be improved. The steps are as follows:

I. Initialization

1) Determine the size of the block q, and generate the number of steps r

2) Select the initial vector matrix $\mathbf{Q}_0 \in R^{N\times q}$

3) Set the maximum number of iteration for shifting, $I_{\max}$

II. Shifting

1) Calculate the shifting μ and ensure that it is not the eigenvalue

2) Decomposite the shifted stiffness matrix $\mathbf{K} - \mu\mathbf{M} = \mathbf{LDL}^T$

3) Perform Sturm sequence check

III. Iterate until reaching $I_{\max}$, then go to II

1) Solve $\mathbf{LDL}^T \tilde{Q}_{k+1} = \mathbf{MQ}_k$ for $k = 0,1,...,r\text{-}1$. Then ***M***-orthonormalize the converged eigenvectors $\mathbf{Q_1}$, $\mathbf{Q_2}$,…,$\mathbf{Q_k}$ by using $\tilde{\mathbf{Q}}_{k+1}$, and form $\mathbf{Q}_{k+1}$.

2) Calculate the projection of $\mathbf{K}$ on $\mathbf{Q} = (\mathbf{Q_1}, \mathbf{Q_2},\ldots,\mathbf{Q_r})$, $\mathbf{K}^* = \mathbf{Q}^T\mathbf{KQ}$

3) Solve the $q\times r$ standard eigenvalue problem $\mathbf{K}^*\mathbf{\Phi}^* = \mathbf{\Phi}^*\mathbf{\Lambda}^*$

4) Form the new eigenvector $\mathbf{X} = \mathbf{Q\Phi}^*$

5) Judge the convergence based on the mode-error criterion. Then remove the converged eigenvectors.

6) If the required number of eigenvalues has been obtained, exit; otherwise, take the unconverged first q vectors as the initial vectors and go to the next iteration.

2.3 Iterative Lanczos method

Lanczos method was presented in the 1950s. It transforms a symmetric matrix into a tri-diagonal matrix, thereby making the procedure more efficient. This method was recognized to be unstable before the 1970s and was not used widely. Paige proved in 1972 that the sufficient and necessary condition for losing the orthogonality is that the eigenvalues of the projected matrix converge to the eigenvalues of the original matrix. Afterwards Wilkinson proposed the strategy of re-orthogonalization, Golub (1972), Underwood (1975) proposed the Block Lanczos method, and Underwood proposed the iterative Lanczos method (see Golub 1972, Underwood 1975).

In this paper, an iterative block Lanczos with re-orthogonalization is used. Steps I and II for this method are the same as the iterative Ritz vector method. The difference is that the Lanczos method used the orthonormalizing coefficients in generation of the Ritz vector. Step III is as follows:

III. Iterate until reaching $I_{\max}$, go to II

1) Solve $\mathbf{LDL}^T \tilde{Q}_{k+1} = \mathbf{MQ}_k$ for $k = 0,1,...,r$. Then ***M***-orthonormalize the converged eigenvectors $\mathbf{Q_1}$, $\mathbf{Q_2}$,…,$\mathbf{Q_k}$ by using $\tilde{\mathbf{Q}}_{k+1}$, and form $\mathbf{Q}_{k+1}$. In this procedure form T^*.

2) Solve the $q\times r$ standard eigenvalue problem $T^*\mathbf{\Phi}^* = \mathbf{\Phi}^*\mathbf{\Lambda}^*$

2.4 Implementation

In the implementation, the cell storage sparse solver developed by Chen *et al.* (2003) will be used in the triangular decomposition of the shifted matrix, forward reduction and back substitution. The efficiency of such a scheme is about 1 to 2 orders faster than that of the skyline solver. The unrolling technique is also used in the orthonormalization, etc., to accelerate the calculations. The double orthogonalization will be processed if the ratio of the norms of a vector before and after orthogonalization exceeds a certain threshold, which means twice orthogonalization will be implemented. In the following tests, the threshold is taken as 10^{12}.

According to experience, for the subspace iteration,

$$I_{\max} = \max(0.5Ns^2/(3Nq^2 + 2Nq + 10q^3), 6) \tag{9}$$

For iterative Ritz vector and iterative Lanczos, take $q = 4$, $r = 6$ and

$$I_{\max} = \max(0.5Ns^2/(3Nq^2r^2 + 2Nqr + 10q^3r^3), 4) \tag{10}$$

Firstly take the shifting as $\mu = 0.5(\tilde{\lambda}_{k+1} + \tilde{\lambda}_{k+2})$. If this fails, then take $\mu = \lambda_k + 0.98(\tilde{\lambda}_{k+1} - \lambda_k)$, where λ_k is the last converged eigenvalue and $\tilde{\lambda}_{k+1}$, $\tilde{\lambda}_{k+2}$ are the next eigenvalues to be converged.

2.5 Numerical examples

Several practical problems are used to compare the above three methods. Part of them are listed in Table 1 and Table 2, in which the PKUSTK series have been used in our previous papers (see Chen *et al*. 2003, Chen and Sun 2005).

2.6 Comparison of three algorithms

For comparison, all the numerical examples are calculated in core, 512 MB memory. All the results are generated by SAP84 and transferred to Harwell-Boeing RSA format (see Yuan *et al.* 2005).

In Table 1, *neq* means the number of equations, and |**K**| and |**L**| are the memory occupied by stiffness and its triangularized factor, respectively. In eight examples of Tables 1 and 2, PKUSTK03, PKUSTK13 and MM-08 have a closed distribution of eigenvalues.

Table 1 Description of the numerical examples

Examples	Description	*neq*	Size (MB)	
			|K|	|L|
PKUSTK01	Beijing Botanical Garden	22,044	3.86	15.97
PKUSTK03	Group Storage	63,336	12.37	65.94
PKUSTK11	Cofferdam	87,804	20.54	214.53
PKUSTK12	Jijian Plaza	94,653	30.07	127.06
PKUSTK13	Machine part	94,893	26.09	234.21
MM-08	Stadium	160,074	25.68	175.56
GUIZHEN	Shijitang	6,840	1.48	6.82
PALACE	Palace for Children	10,860	2.28	12.82

Table 2 CPU time (sec), number of shiftings and iterations to calculate 10 and 80 eigenpairs by the three methods

Examples	No. of eigen-pairs	Subspace Iteration			Iterative Ritz vector			Iterative Lanczos		
		CPU (sec)	No. of shiftings	No. of iterations	CPU (sec)	No. of shiftings	No. of iterations	CPU (sec)	No. of shiftings	No. of iterations
PKUSTK01	10	6.41	2	22	4.27	1	4	3.89	1	4
	80	70.83	9	169	35.78	7	21	39.03	7	25
PKUSTK03	10	53.53	3	41	28.95	2	7	28.88	2	8
	80	330.84	11	201	154.42	8	26	133.85	9	30
PKUSTK11	10	51.14	1	6	48.52	1	3	47.67	1	3
	80	945.62	8	145	399.32	7	26	330.17	6	21
PKUSTK12	10	39.70	1	14	32.66	1	4	40.92	2	5
	80	651.35	11	201	237.76	6	23	257.73	7	28
PKUSTK13	10	73.20	1	11	50.91	1	3	50.86	1	4
	80	504.18	4	75	375.18	6	21	416.45	7	25
MM-08	10	129.64	2	30	78.64	2	6	79.64	2	6
	80	1098.28	10	185	418.78	7	27	523.98	10	35
GUIZHEN	10	1.30	1	11	1.05	1	3	0.95	1	3
	80	19.90	7	129	7.25	5	20	7.12	6	22
PALACE	10	3.59	2	15	2.70	1	4	3.61	2	5
	80	39.01	7	126	19.93	7	24	19.84	7	25

In the calculation, the tolerance is taken as $\varepsilon_\varphi = 10^{-4}$ for PKUSTK03, and as $\varepsilon_\varphi = 10^{-3}$ for others. The first 6 digits of all the eigenvlues calculated by the three methods are consistent. The maximum relative error is 2.56×10^{-5} and the average error is 1.56×10^{-6}. This means the reliabilities of the three methods. Averagely speaking, the iterative Ritz vector method is the fastest one. It is about 2 times faster than the subspace iteration. The iterative Lanczos is a bit slower than the iterative Ritz vector and faster than the subspace iteration.

In the tests with subspace iterations, a tolerance of $\varepsilon_\varphi = 10^{-3}$ for mode convergence criterion is adopted for the eigenvectors, which is roughly equivalent to $\varepsilon_\lambda = 10^{-7}$ for the eigenvalue convergence criterion; also, a tolerance of $\varepsilon_\varphi = 10^{-4}$ is about equivalent to $\varepsilon_\lambda = 10^{-9}$. In this paper the procedures of the iterative Lanczos and iterative WYD-Ritz vector methods are quite similar, so the results obtained by them are basically consistent.

2.7 Comparison with ANSYS

For comparison, the three examples described in Table 3 are solved by the subspace iteration, iterative Ritz vector and iterative Lanczos of this paper and the subspace iteration and block Lanczos of ANSYS 8.1. The CPU times spent by each method is listed in Table 4. In the tests, we take $\varepsilon_\varphi = 10^{-3}$ as the convergence criterion for all the three methods. The convergence criterion in Eq. (3) is used with $\varepsilon_\lambda = 10^{-5}$ for the subspace iteration in ANSYS.

The eigenvalues for the example TABT are not dense. The example BUAA1 has rigid modes and closed eigenvalues. The results appear to be consistent for 6 digits for the eigenvalues solved by ANSYS and this paper, except BUAA2. The example BUAA2 is an axisymmetric structure with 6

Table 3 Examples for comparison with ANSYS

Example	Description	*neq*	$\|K\|$ (MB)
TABT	Tall building, 8800 SHELL63 element, 1000 BEAM188 element	60,108	11.05
BUAA1	Gear of helicopter, 2840 SOLID95 element	46,080	24.91
BUAA2	Gear of helicopter, 5360 SOLID95 element	82,800	55.26

Table 4 Comparison of CPU time (sec) solved by present methods with ANSYS

Example	No. of calculated modes	ANSYS		Present		
		Subspace Iteration	Lanczos	Subspace Iteration	Iterative Lanczos	Iterative Ritz vector
TABT	10	68.11	15.87	18.16	16.30	12.66
	20	162.53	23.20	68.83	37.91	25.77
	40	254.92	29.44	146.23	49.19	41.66
	80	533.22	53.45	289.31	93.81	84.03
BUAA1	10	822.16	72.42	42.14	23.92	24.16
	20	1738.20	92.67	69.91	47.73	34.19
	40	4752.39	132.28	166.83	84.00	87.41
	80	5496.28	221.50	378.72	185.28	169.36
BUAA2	10	405.44	282.78	83.09	53.47	55.78
	20	> 9000	345.89	174.27	95.55	102.33
	40	No convergence	484.58	317.70	220.81	222.25
	80	No convergence	649.20	586.47	448.75	481.95

rigid modes, multiple and closed eigenvalues. In the calculation, the Lanczos method in ANSYS failed to recognize 6 rigid body modes correctly. The subspace iteration in ANSYS failed to find all the multiple eigenvalues. Four multiple eigenvalues were lost in calculation of the 10 eigenvalues, resulting in the failure of iterations. Also, no convergence was achieved in calculation of the 20 eigenvalues.

Comparing the subspace iteration with ANSYS, the subspace iteration algorithm in this paper is much more efficient and stable. Comparing the block Lancozs in ANSYS, the iterative Ritz vector and iterative Lanczos methods in this paper are relatively conservative and safer. The efficiency in this paper by using the cell sparse algorithm is a bit better than ANSYS. The iterative Lanczos in this paper is more effective than the block Lanczos in ANSYS when there exist multiple eigenvalues.

3. Discussion of convergence criteria

There are generally three distinct convergence criteria in the implementation of SIM for an approximated eigenpair $(\lambda_i^{(k)}, \varphi_i^{(k)})$ at iteration k:

A. The relative difference of approximated eigenvalues in terms of consecutive iterations:

$$tol_1 = \left| \frac{\lambda_i^{(k)} - \lambda_i^{(k-1)}}{\lambda_i^{(k)}} \right| < \varepsilon_\lambda \tag{11}$$

B. The relative mode error or error measure with 2-norm through the eigenpair $(\lambda_i^{(k)}, \varphi_i^{(k)})$:

$$tol_2 = \frac{\left|K\varphi_i^{(k)} - \lambda_i^{(k)} M\varphi_i^{(k)}\right|_2}{\left|K\varphi_i^{(k)}\right|_2} < \varepsilon_\varphi \tag{12}$$

C. The error bound of eigenvalue through the corresponding eigenpair $(\lambda_i^{(k)}, q_i^{(k)})$ in the subspace:

$$\min_j \left|\frac{\lambda_j - \lambda_i^{(k)}}{\lambda_j}\right| \le tol_3 = \left\{1 - \frac{(\lambda_i^{(k)})^2}{(q_i^{(k)})^T q_i^{(k)}}\right\}^{1/2} < \varepsilon_q \tag{13}$$

Consider a SIM conducted with a half-bandwidth stiffness matrix. The count of operations required for the above three criteria is summarized in Table 5. Criterion A is the most popular one for implementation. It is simple and clear, but can only control the accuracy of eigenvalues and sometimes fails to identify incorrect eigenvectors and even eigenvalues. Besides, this criterion does not allow a shift to the center of eigenvalues of concern. Using this criterion, a shift to the center of two eigenvalues may result in the so-called ghost eigenvalue, for which an example was given by Hughes (1987).

Criterion B is a good choice for the convergence of eigenpairs, and has clear physical meaning. The vector $K\varphi_i^{(k)}$ represents the elastic nodal forces and $\lambda_i^{(k)} M\varphi_i^{(k)}$ the inertia nodal forces when the finite element assemblage is vibrating in the mode $\varphi_i^{(k)}$. Thus, tol_2 is related to the out-of-balance nodal forces. Unfortunately, this criterion requires either larger floating point operations or extra memory in implementation.

Bathe (1996) suggested measuring the final accuracy with tol_2, but not in the SIM procedure. Nguyen *et al.* (2000) implemented the convergence criterion B for both the subspace iteration method and Lanczos method. Gong *et al.* (2005) examined the efficiency of several general eigenvalue algorithms for natural vibrations with tol_2.

Criterion C is an expression of the computable error bound proposed by Matthies (1985) and can be effectively computed in the subspace with only negligible $(q + 2)$ multiplications, where q is the constant size of the subspace. In contrast, criterion B requires at least $N(q + 3)$ multiplications. Although it is advantageous in operations, the followings are not clear from the monograph on FEA (see Bathe 1996):

a. Whether tol_3 can be used for iteration with a nonzero shift μ.

b. Whether tol_3 represents the convergence of eigenvectors.

c. Whether tol_3 can be used, if the mass matrix is only positive semi-definite.

The above three questions limit the applications of tol_3 as an efficient convergence criterion in the subspace iteration method. Based on the evaluation by Bathe (1996) and Matthies (1985), a computable error bound during iterations for the Rayleigh quotient

Table 5 Count of operations for the three convergence criteria

Criterion	Count of operations
A	1 division
B	1 division, 1 square root, $N(2b + 3)$ multiplications[1]
	1 division, 1 square root, $N(q + 3)$ multiplications[2]
C	1 division, 1 square root, $(q + 1)$ multiplications

[1] b is the half-bandwidth of the stiffness matrix

[2] extra array memory $X(N,q)$ for the previous subspace $X_{k\text{-}1}$ is required

$$R(\bar{x}) = \frac{\bar{x}^T \mathbf{K} \bar{x}}{\bar{x}^T \mathbf{M} \bar{x}} \tag{14}$$

can be represented as

$$\min_j \left| \frac{\lambda_j - R(\bar{x})}{\lambda_j} \right| \leq \left\{ 1 - \frac{R^2(\bar{x})}{\hat{x}^T \mathbf{M} \hat{x} / \bar{x}^T \mathbf{M} \bar{x}} \right\}^{1/2} \tag{15}$$

Here, the vectors $\bar{x}$ and $\hat{x}$ satisfy

$$\mathbf{K} \bar{x} = \mathbf{M} \hat{x} \tag{16}$$

and λ_j ($j = 1,2,\ldots, n$) are eigenvalues of Eq. (1). A proof of Eq. (15) can be found in the literature (see Yuan *et al*. 1989, Akl *et al*. 1982).

If the shifting $\mu = 0$, the right hand side of Eq. (15) can be further simplified and implemented as a computable error bound for the subspace iteration (see Bathe 1996):

$$tol_3 = \left\{ 1 - \frac{R^2(\bar{x})}{\hat{x}^T \mathbf{M} \hat{x} / \bar{x}^T \mathbf{M} \bar{x}} \right\}^{1/2} = \left\{ 1 - \frac{(\lambda_i^{(k)})^2}{(q_i^{(k)})^T q_i^{(k)}} \right\}^{1/2} \tag{17}$$

where the Rayleigh quotient $\lambda_i^{(k)}$ is the calculated eigenvalue approximation and $q_i^{(k)}$ is the corresponding eigenvector in the subspace, i.e., $(\lambda_i^{(k)}, q_i^{(k)})$ is an eigenpair.

From Eqs. (15) and (17), the quantity $tol_3 = \{1 - (\lambda_i^{(k)})^2 / [(q_i^{(k)})^T q_i^{(k)}]\}^{1/2}$ evaluated in the corresponding subspace can be used to control the convergence of eigenvalues. It should be noted that the second equal sign in Eq. (17) is not valid for the case $\mu \neq 0$. However, the extension of tol_3 to $\mu \neq 0$ requires just a slight modification of the procedure and the computation cost remains almost unchanged.

Let us consider an eigenvector approximation of SIM in its k^{th} iteration:

$$\bar{x} = \bar{\mathbf{X}}_k q_i^{(k)} \tag{18}$$

with

$$\bar{x}^T \mathbf{M} \bar{x} = (q_i^{(k)})^T \bar{\mathbf{X}}_k^T \mathbf{M} \bar{\mathbf{X}}_k q_i^{(k)} = (q_i^{(k)})^T \mathbf{M}^* q_i^{(k)} = 1 \tag{19}$$

Implicitly we can derive from Eq. (8)

$$\mathbf{K} \bar{\mathbf{X}}_k q_i^{(k)} = \mathbf{M} (\mathbf{X}_{k-1} + \mu \bar{\mathbf{X}}_k) q_i^{(k)} \tag{20}$$

Denote

$$\hat{x} = (\mathbf{X}_{k-1} + \mu \bar{\mathbf{X}}_k) q_i^{(k)} \tag{21}$$

Thus we have symbolically

$$\begin{aligned} \hat{x}^T \mathbf{M} \hat{x} &= (q_i^{(k)})^T \mathbf{X}^T{}_{k-1} \mathbf{M} \mathbf{X}_{k-1} q_i^{(k)} + \mu^2 (q_i^{(k)})^T \bar{\mathbf{X}}_k^T \mathbf{M} \bar{\mathbf{X}}_k q_i^{(k)} + 2\mu (q_i^{(k)})^T \bar{\mathbf{X}}_k^T \mathbf{M} \mathbf{X}_{k-1} q_i^{(k)} \\ &= (q_i^{(k)})^T q_i^{(k)} + \mu^2 + 2\mu (q_i^{(k)})^T \bar{\mathbf{X}}_k^T (\mathbf{K} - \mu \mathbf{M}) \bar{\mathbf{X}}_k q_i^{(k)} \\ &= (q_i^{(k)})^T q_i^{(k)} + \mu^2 + 2\mu (\lambda_i^{(k)} - \mu) \\ &= (q_i^{(k)})^T q_i^{(k)} + \mu (2\lambda_i^{(k)} - \mu) \end{aligned} \tag{22}$$

so that Eq. (17) is modified to

$$tol_4 = \left\{1 - \frac{R^2(\bar{x})}{\widehat{x}^T\mathbf{M}\widehat{x}\,/\bar{x}^T\mathbf{M}\bar{x}}\right\}^{1/2} = \left\{1 - \frac{(\lambda_i^{(k)})^2}{(q_i^{(k)})^T q_i^{(k)} + \mu(2\lambda_i^{(k)} - \mu)}\right\}^{1/2} \tag{23}$$

Combining Eq. (21), the new quantity tol_4 can be used as a convergence criterion for the subspace iteration method with a nonzero shifting μ, i.e., the convergence criterion C can now be updated as follows (see Chen *et al*. 2007):

$$\min_j \left|\frac{\lambda_j - \lambda_i^{(k)}}{\lambda_j}\right| \le tol_4 = \left\{1 - \frac{(\lambda_i^{(k)})^2}{(q_i^{(k)})^T q_i^{(k)} + \mu(2\lambda_i^{(k)} - \mu)}\right\}^{1/2} < \varepsilon_q \tag{24}$$

Obviously, Eq. (24) is an extension of Eq. (15) and can be computed with just a small number of multiplications. When $\mu = 0$, tol_4 degenerates to tol_3. For convenience, we consider that tol_4 and tol_3 are synonym in the following discussion, and take Eq. (24) as the convergence criterion C.

In fact, the inverse **M**-norm of a vector defined as

$$|x|_M = (x^T\mathbf{M}^{-1}x)^{1/2} \tag{25}$$

is equivalent to the 2-norm, i.e., there exist two positive constants c_1 and c_2, $c_2 > c_1$, so that

$$c_1|x|_2 \le |x|_M \le c_2|x|_2 \tag{26}$$

Simple calculation leads to an equality for an approximated eigenpair $(R(\bar{x}), \bar{x})$:

$$\begin{aligned}\frac{|\mathbf{K}\bar{x} - R(\bar{x})\mathbf{M}\bar{x}|_M}{|\mathbf{K}\bar{x}|_M} &= \left\{\frac{\bar{x}^T\mathbf{K}\mathbf{M}^{-1}\mathbf{K}\bar{x} - 2R(\bar{x})\bar{x}^T\mathbf{K}\bar{x} + R^2(\bar{x})\bar{x}^T\mathbf{M}\bar{x}}{\bar{x}^T\mathbf{K}\mathbf{M}^{-1}\mathbf{K}\bar{x}}\right\}^{1/2} \\ &= \left\{1 - \frac{2R(\bar{x})\bar{x}^T\mathbf{K}\bar{x}\,/\bar{x}^T\mathbf{M}\bar{x} - R^2(\bar{x})}{\widehat{x}^T\mathbf{M}\widehat{x}\,/\bar{x}^T\mathbf{M}\bar{x}}\right\}^{1/2} \\ &= \left\{1 - \frac{R^2(\bar{x})}{\widehat{x}^T\mathbf{M}\widehat{x}\,/\bar{x}^T\mathbf{M}\bar{x}}\right\}^{1/2}\end{aligned} \tag{27}$$

Using Eqs. (26) and (27), we obtain

$$\frac{c_1}{c_2}\frac{|\mathbf{K}\bar{x} - R(\bar{x})\mathbf{M}\bar{x}|_2}{|\mathbf{K}\bar{x}|_2} \le \left\{1 - \frac{R^2(\bar{x})}{\widehat{x}^T\mathbf{M}\widehat{x}\,/\bar{x}^T\mathbf{M}\bar{x}}\right\}^{1/2} \le \frac{c_2}{c_1}\frac{|\mathbf{K}\bar{x} - R(\bar{x})\mathbf{M}\bar{x}|_2}{|\mathbf{K}\bar{x}|_2} \tag{28}$$

Thus, the quantities tol_2 and tol_4, and hence the criteria B and C are equivalent. We have to point out here that, even if **M** is positive semi-definite, the inequality (28) remains valid in the subspace iteration procedure.

4. Aggressive shifting acceleration for subspace iteration

Since the subspace iteration has a better quality of stability and reliability in calculation,

researchers like to use it as the default algorithm in software packages. However, the efficiency is not satisfactory. Shifting provided the key to improve the convergence speed of the subspace iteration. In this regard, how to obtain the value of shifting and to solve the problems caused by shifting has been an issue of great concern. Zhao *et al*. (2007) proposed an "aggressive shifting" strategy for accelerating the convergence rate and for treating the problems caused by shifting.

By denoting the converged eigenvalues as λ and approximated eigenvalues as $\tilde{\lambda}$, Bathe and Ramaswamy (1980) proposed the largest allowable shift:

$$\mu_B = \frac{\lambda_s + \lambda_{s-1}}{2} \text{ and } 1.01\ \lambda_{s-1} \le \mu_B \le 0.99\lambda_s \tag{29}$$

where λ_s is the largest of the eigenvalues which have converged to a given tolerance. If Eq. (29) is not satisfied, decrease the value of s (using $s \leftarrow s - 1$). In order to gain higher convergence rate, Wilson and Itoh (1983) suggested to move the shift even closer to the next approximated eigenvalue $\tilde{\lambda}_{s+1}$, that is,

$$\mu_W = \lambda_s + 0.9(\tilde{\lambda}_{s+1} - \lambda_s) \text{ or } \mu_W = 0.99\lambda_s, \text{ if } (\tilde{\lambda}_{s+1} - \lambda_s) \le 0.01\lambda_s \tag{30}$$

Experience showed that this shifting strategy often leads to singularity of the shifted matrix $(\mathbf{K} - \mu\mathbf{M})$, since μ_W may be quite close to an eigenvalue. Transforming to the reciprocal problem provides the possibility of Sturm's sequence check. Both shifting procedures require a prior Sturm's sequence check, i.e., to check the match of the number of converged eigenvalues s and the number of negative diagonal entries in the matrix factor $\mathbf{D}$ after the triangular factorization of $\mathbf{K} - \mu\mathbf{M} = \mathbf{LDL}^T$. In other words, each shifting and successive Sturm's sequence check involves a triangular factorization. Of course, the computation involved is expensive. The kind of SIM with conservative shifting strategies μ_B or μ_W and a prior Sturm's sequence check has been illustrated in Fig. 1.

As shown in Fig. 1 and Eq. (31), Gong *et al.* (2005) moved the shift beyond the converged eigenvalues to the center of next two approximated Ritz-values $\tilde{\lambda}_{s+1}$ and $\tilde{\lambda}_{s+2}$, that is,

$$\mu_G = \frac{\tilde{\lambda}_{s+1} + \tilde{\lambda}_{s+2}}{2} \text{ and } \tilde{\lambda}_{s+1} + 0.001\lambda_1 < \mu_G < \tilde{\lambda}_{s+2} - 0.001\lambda_1 \tag{31}$$

where a Ritz-value $\tilde{\lambda}_j (j = s+1, s+2)$ is defined as an eigenvalue in the subspace, of which the corresponding computable error bound is not larger than 10^{-1}, i.e.,

$$\min_i \left| \frac{\lambda_j^{(k)} - \lambda_i}{\lambda_i} \right| \le tol_4 \le 10^{-1} \tag{32}$$

This simple improvement averagely improves the speed of computation by up to 17% compared with SIM with the shift μ_B of Eq. (29), if the high performance cell sparse fast direct solver

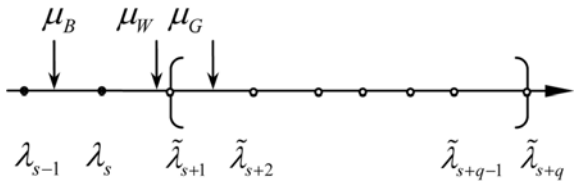

Fig. 1 Shifting strategies

(CSFD) (see Chen *et al*. 2003, Chen and Sun 2005) is used. To guarantee the stability condition in Eq. (31), the Ritz-value $\tilde{\lambda}_{s+1}$ and $\tilde{\lambda}_{s+2}$ can be substituted by their upper bound and lower bound, respectively. Since singularity may occur in the shifting process as an accelerated scheme, the shift value must be carefully chosen to avoid singularity. In this sense, a knowledge of the error bound of eigenvalues is helpful to avoid singularity encountered by any shift, because we can establish a series of small intervals that consist of individual eigenvalues and prevent the shift from being placed in these small intervals (see Bathe 1996).

In their early study, Bathe and Ramaswamy (1980) discussed the allocation of a possible shift under the assumption that the lowest s eigenpairs have converged to certain tolerance. The effort of

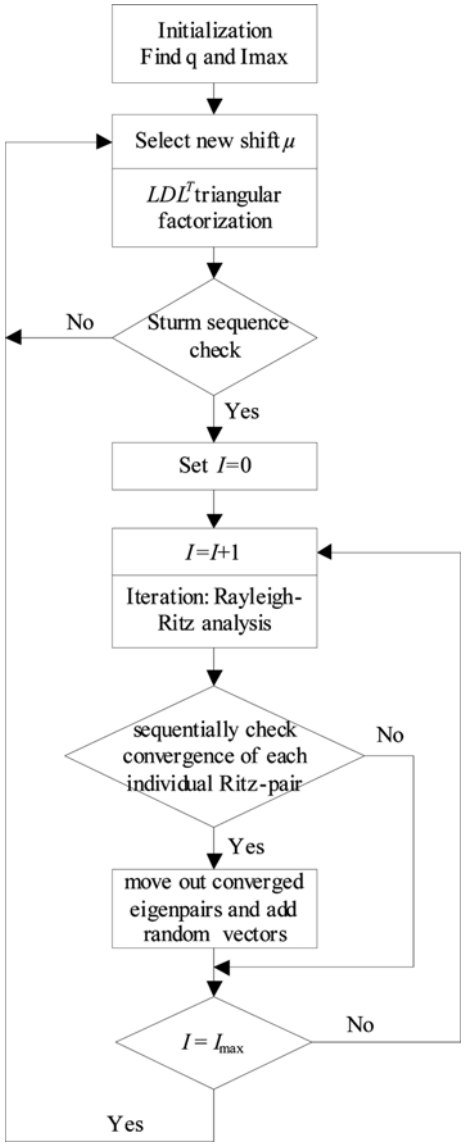

Fig. 2 Flowchart of SIM with conservative shifting and prior Sturm's sequence check

next shift is to accelerate the convergence rate for eigenvalues between λ_{s+1} and λ_{s+q}, allowing further shift. Considering the next subspace iterations, it is important to assure that the iteration vectors do not converge to already calculated eigenvectors, by employing the Gram-Schmidt orthogonalization, such that all q vectors are orthogonalized with respect to the s converged eigenvectors. In order to achieve better performance, i.e., the eigenpairs located at the left to μ should be converged faster than those at the right to μ, the shift μ should satisfy the following condition:

$$\mu - \lambda_s < \tilde{\lambda}_{s+q+1} - \mu \tag{33}$$

which means that μ is in the left half of the eigevalue spectrum $\tilde{\lambda}_{s+1}$ to $\tilde{\lambda}_{s+q}$. After shifting, the new convergence rate for each individual eigenpair is $|\mu - \lambda_j| < |\tilde{\lambda}_{s+q+1} - \mu|$, $j = s+1,\ldots, s+q$. To solve the inequality for λ_s, we have

$$\mu = \lambda_s + \alpha(\tilde{\lambda}_{s+q+1} - \lambda_s) \text{ with } \alpha < 1/2 \tag{34}$$

where λ_s is the converged eigevalue and $\tilde{\lambda}_{s+q+1}$ is an estimated eigenvalue.

Obviously, the stability of subspace iterations requires that μ must be relatively far away from any eigenvalue. The shift μ is therefore chosen to lie midway between two well-spaced Ritz-values, which can be found through the error bound formula tol_4. In the implementation, Bathe and Ramaswamy (1980) and Wilson and Itoh (1983) have made a very conservative choice of μ, as shown in Eqs. (29) and (30), respectively. The condition $\alpha < 1/2$ was only used as an additional control of the shift given in Eq. (29). The success of the shifting strategy in Eq. (31) proposed by Gong *et al.* (2005) encourages us to shift more aggressively, that is, to the interval between $\tilde{\lambda}_{s+u}$ and $\tilde{\lambda}_{s+u+1}$, where s denotes the highest converged eigenvalue and u the depth of the aggressive shifting, satisfying $0 < u \le q - 1$.

The SIM with an aggressive shifting strategy as shown in Fig. 3 and Fig. 4 is proposed as follows:

I. Replace the convergence criterion A by C as described in previous discussion. The convergence criterion C also provides individual upper and lower bounds for each eigenvalue, which are used in identifying the Ritz-value and finding a new shift.

II. The shift μ should be located in the left half of the interval bracketed by two converged eigenvalues: $[\lambda_{s+1}, \lambda_{s+q}]$. In implementation we chose a shift $\mu \approx 0.5(\tilde{\lambda}^{+}_{s+u-1} + \tilde{\lambda}^{-}_{s+u})$, where s denotes the highest converged eigenvalue and $u = \lfloor \alpha q \rfloor$, satisfying $0 \le \alpha < 0.5$. The parameter u is the depth of the aggressive shifting and α is a parameter of the relative depth. The upper bound $\tilde{\lambda}^{+}_{s+u-1}$ and lower bound $\tilde{\lambda}^{-}_{s+u}$ of two Ritz-values can be evaluated from their approximations and error bounds, respectively (Eq. (28)). In this study, we have tested a series of α and found that the optimum value of the parameter α is around 0.4.

III. Perform as many iterations as necessary, so that all eigenpairs located to the left of μ are converged. To ensure that, a posterior Sturm's sequence check is necessary. In addition, for each shifting a maximum number of iterations, say $I_{\max}$, should be carried out.

IV. Since all the above Ritz values are approximated in the iteration procedure, it is not guaranteed that the shift μ is really located in the left half between $[\lambda_{s+1}, \lambda_{s+q}]$ as was required. In this case the convergence rate will be slow down as too many iterations are required. It is better to set an additional control to terminate the iteration of the current shift and turn on a new shift. In this study, if the count of iterations reaches 1.5 $I_{\max}$ and not all eigenpairs in the left of the current shift μ have been obtained, a backward shifting is

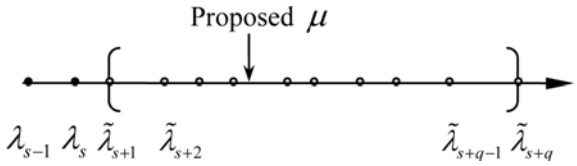

Fig. 3 An aggressive shift

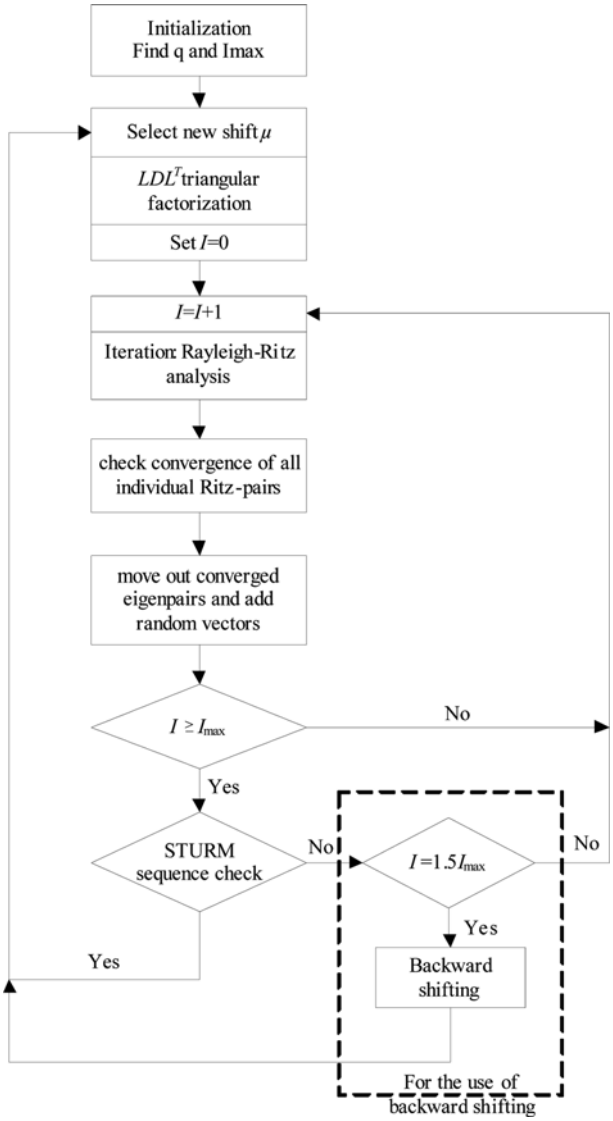

Fig. 4 Flowchart of SIM with aggressive shifting and posterior Sturm's sequence check

conducted, by moving to a point to the left of the lowest eigenvalue approximation in the current subspace. The backward shifting is included in the flowchart of Fig. 4 with a dashed frame.

5. Numerical tests

The SIM with the following shifting strategies will be tested:
BSIM: SIM with the classic shift μ_B of Eq. (29).
ISIMG: SIM with the shift μ_G of Eq. (31).
ISIM0.4: SIM with the proposed aggressive shift ($\alpha = 0.4$).
ISIMB0.4: SIM with the proposed aggressive shift ($\alpha = 0.4$) and backward option.

All the SIMs are conducted with a sparse solver CSFD (see Chen *et al*. 2003, Chen and Sun 2005). The SIM parameter $I_{\max}$ (iterations per shift) and q (optimum subspace size) are selected as proposed by Wilson and Itoh (1983). The convergence criterion C with $\varepsilon_q = 10^{-4}$ for the eigenvectors is roughly equivalent to the tolerance $\varepsilon_\lambda = 10^{-7}$ for the eigenvalues. Structural models with the number of equations between 10,800 and 160,074 are listed in Table 6, with their wireframes plotted in Fig. 5. These problems are selected from our home collection. All test problems have arisen from mechanical and civil engineering applications and have been extensively used in earlier research works. In Table 6, *neq*, and *nzr* denote the number of equations and number

Table 6 Test examples and their descriptions

Problem	Description	*neq*	*nzr*
PALACE	Museum in Shenzhen	10,800	295,302
WENSHI	Botanical exhibition hall	22,044	500,712
JIJIAN	Twin tower tall building	94,653	3,803,485
MM-08	A model of stadium	160,074	3,304,314

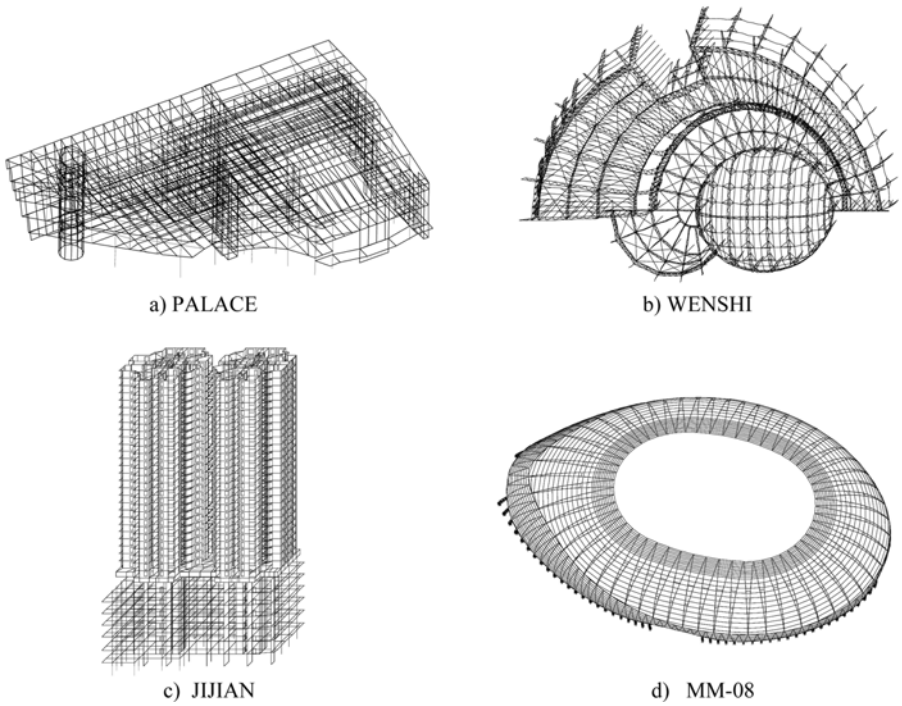

Fig. 5 Examples from home collection. a) PALACE, b) WENSHI, c) JIJIAN, d) MM-08

of non-zeros in the upper parts of the stiffness matrices, respectively. Our home test data are produced by SAP84 (see Yuan et al. (2005)). All the SIMs are conducted on the platform: A Pentium IV 3.0 GHz with 2 GB RAM machine. For the sake of comparison of related computations, no out-of-core strategy is considered for the triangular factorization, forward reduction, back substitution and SIM itself. All programs are compiled by Compaq Visual Fortran 6.5, with compiler options /architecture:p6p /tune:p6p.

Fig. 6 shows the CPU time curves in the solution of 200 eigenpairs with BSIM, ISIMG, and ISIM0.4. In Fig. 6(d) there occurs a jump in the ISIM0.4 CPU time curve around $k = 181$ of MM-08. A check of the iteration history indicates that at the shift $\mu = 0.99484 \times 10^3$ (roughly at the center of λ_{184} and λ_{185}), the convergence of the left eigenvalues λ_{180}, λ_{181} and λ_{182} in the interval $[\lambda_{180} = 0.963499 \times 10^3, \lambda_{192} = 0.100820 \times 10^4]$ requires 25 iterations alone, since there is an eigenvalue jump between λ_{182} and λ_{183}, and an eigenvalue cluster around $\lambda_{192} = 0.100820 \times 10^4$. Hence, the selected shift $\mu = 0.99484 \times 10^3$ is no longer located in the left half of $[\lambda_{180}, \lambda_{192}]$ as

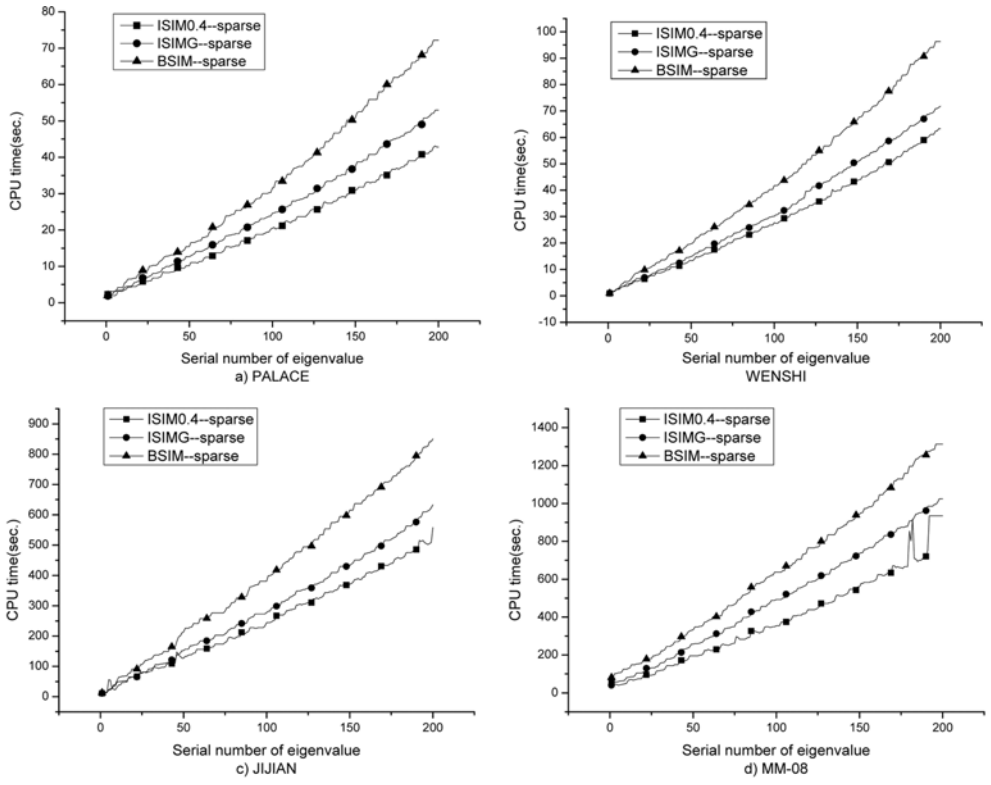

Fig. 6 CPU time of BSIM, ISIMG and ISIM0.4 for extracting 200 eigen pairs. a) PALACE, b) WENSHI, c) JIJIAN, d) MM-08

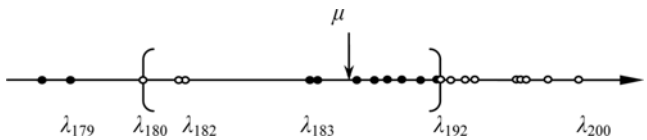

Fig. 7 Spectrum of MM-08 near the shift $m = 0.99484 \times 10^3$

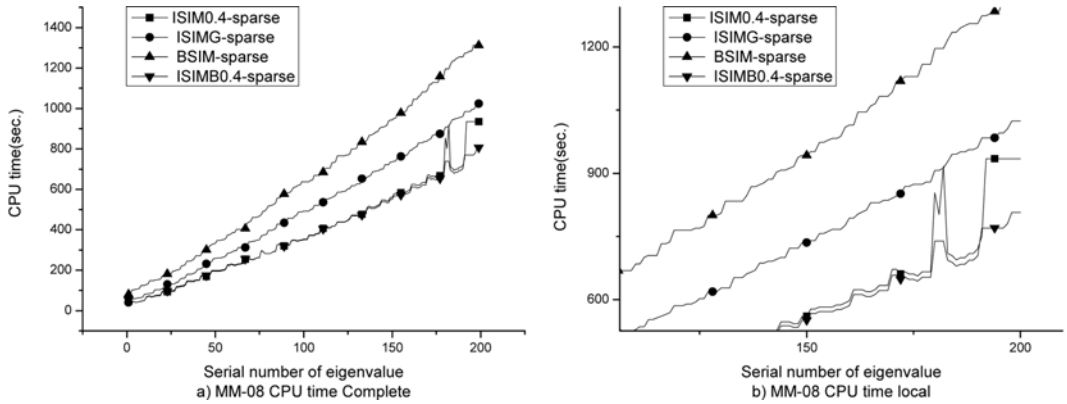

Fig. 8 CPU time of BSIM, ISIMG, ISIM0.4 and ISIMB0.4 for extracting 200 eigen pairs of MM-08. a) Complete CPU time curves, b) Local CPU time curves

initially conceived. After convergence of λ_{183} to λ_{188}, the current shift is still located in the right half of the interval $[\lambda_{180}, \lambda_{196}]$, meaning that the asymptotic convergence rate is quite low. Such a phenomenon was referred to as convergence delay. Fig. 7 shows the spectrum around this shift. In order to overcome such a difficulty, ISIMB moves the shift backwards to the center of $[\lambda_{179} = 0.95276 \times 10^3, \lambda_{180} = 0.963499 \times 10^3]$ after a certain number of iterations, say $1.5I_{\max} = 7$, so that λ_{180} to λ_{182} can converge rapidly. In Fig. 8, the CPU time of BSIM, ISIMG, ISIM0.4 and ISIMB0.4 are compared. It was found that ISIMB0.4 can effectively reduce the convergence delay, and the problem with a too far shifting was resolved.

In fact, this example mirrors a serious difficulty arising from aggressive shifting, i.e., the shift is not located in the left half of an eigenvalue interval as desired. Fortunately, such a problem can be solved by a backward shifting. In addition, the CPU time reduced by aggressive shifting is more significant for SIM with sparse solver than with skyline solver. ISIMB0.4 gains an average speedup of 1.573 in comparison with BSIM in extracting 200 eigenvalues, and of 1.133 in comparison with ISIMG.

We have tested weaker and stronger convergence condition, i.e., $\varepsilon_q = 10^{-3}$ and 10^{-5}, respectively, for the above examples. For $\varepsilon_q = 10^{-3}$, difficulty arose in Sturm's sequence check if higher eigenpairs are to be extracted for the problem JIJIAN, since the relative differences between eigenvalues are very small. For $\varepsilon_\lambda = 10^{-6}$, Sturm's sequence check failed at the same position. For $\varepsilon_q = 10^{-5}$, an average of 11% more CPU time is required for ISIMG, and of 7% for ISIMB0.4.

6. Conclusions

In this paper we reported some advances in the classical iterative methods for generalized eigenvalue problems. The major conclusions are summarized as follows:

I. After comparison of subspace iteration, iterative Ritz vector and iterative Lanczos methods, the iterative Ritz vector method appears to be the most efficient one.

II. Using cell sparse solution scheme, instead of the traditional skyline solver, considerable improvement in efficiency was obtained.

III. Using the "mode error" convergence criterion, instead of the eigenvalue-only criterion, a more

stable and efficient iteration strategy for all three iterative methods was established.

IV. An aggressive shifting strategy is presented for the subspace iteration method in solution of eigenvalue problems arising from finite element analysis. The extended computable error bound has been conducted with an aggressive shifting strategy to accelerate the convergence of the classic subspace iteration method. Numerical examples showed the proposed combination can tremendously reduce the computational effort in terms of the count of iterations and shifttings, as well as the CPU time. A posterior Sturm's sequence check is necessary. If the aggressive shift is in trouble, then a backward shifting can be adopted to solve the problem.

Acknowledgements

The support from the National Natural Science Foundation of China under grant No. 10572003 is gratefully acknowledged. The authors thank Mr. Bin Chen for his assistance in preparing more than 20 test problems. Part of this paper will be presented in APCOM'07 & EPMESC XI to be held on December 3, 2007 as a plenary lecture.

References

Akl, F.A., Dilger, W.H. and Irons, B.M. (1979), "Over-relaxation and subspace iteration", *Int. J. Numer. Meth. Engg.*, **14**, 629-630.

Akl, F.A., Dilger, W.H. and Irons, B.M. (1982), "Acceleration of subspace iteration", *Int. J. Numer. Meth. Engg.*, **18**, 583-589.

Bathe, K.J. (1996), *Finite Element Procedure*, Prentice-Hall, Englewood Cliffs, New Jersey.

Bathe, K.J. and Ramaswamy, S. (1980), "An accelerated subspace iteration method", *Comput. Meth. Appl. Mech. Engg.*, **23**, 313-331.

Bathe, K.J. and Wilson, E.L. (1972), "Large eigenvalue problems in dynamic analysis", *J. Eng. Mech. Div.*, ASCE, **98**, 1471-1485.

Bertolini, A.F. and Lam, Y.C. (1995), "Accelerated reduction of subspace upper bound by multiple inverse iteration", *Comput. Systems Eng.*, **6**, 67-72.

Chen, P., Gong, Y.C., Chen, Y.Q., et al. (2007), "On the computable eigenvalue error bound in the subspace iteration method", Submitted to *J. of Computational Mechanics* (in revision).

Chen, P. and Sun, S.L. (2005), "A new high performance sparse static solver in finite element analyses", *Acta Mechanica Solida Sinica*, **18**, 248-255.

Chen, P., Zheng, D., Sun, S.L. and Yuan, M.W. (2003), "High performance sparse static solver in finite element analyses with loop-unrolling", *Advances in Engineering Software*, **34**, 203-215.

Clint, M. and Jennings, A. (1970), "The evaluation of eigenvalue and eigenvectors of real symmetric matrices by simultaneous iteration", *The Comput. J.*, **13**, 76-80.

Duff, I.S., Grimes, R. and Lewis, J. (1992), *The User's Guide for the Harwell-Boeing Sparse Matrix Collection* (Release I), Technical Report TR/PA/92/86, CERFACS, Lyon, France.

Golub, G.H. (1972), "Some uses of the Lanczos algorithm in numerical linear algebra", in Miller J.J.H. eds. *Topics in Numerical Analysis*, 173-194.

Gong, Y.C., Zhou, H.W., Yuan, M.W. and Chen, P. (2005), "Comparison of subspace iteration, iterative Ritz vector method and iterative Lanczos method", *J. Vib. Eng.*, **18**, 227-232 (in Chinese).

Grimes, R.G., Lewis, J.G. and Simon, H.D. (1994), "A shifted block Lanczos algorithm for solving sparse symmetric generalized eigenproblems", *SIAM J. Matrix. Anal. Appl.*, **15**(1), 228-272.

Hughes, T.J.R. (1987), *The Finite Element Method-Linear Static and Dynamic Finite Element Analysis*,.

Prentice-Hall, Englewood Cliffs, New Jersey.
Lee, J.H. and Kim, H.T. (2005), "Hybrid method for natural frequency extraction: performance improvement using Newton-Raphson method", *J. Electromagnetic Waves and Applications*, **19**, 1043-1055.
Matthies, H. (1985), "Computable error bounds for the generalized symmetric eigenproblem", *Communications Appl Num Methods*, **1**, 33-38.
Nguyen, D.T., Bunting, C.F., Moeller, M.J., Runesha, H. and Qin, J. (2000), "Subspace and Lanczos sparse eigen-solvers for finite element structural and electromagnetic applications", *Advances in Engineering Software*, **31**, 599-606.
Press, W.H. et al. (1996), *Numerical Recipes in Fortran 90: the Art of Parallel Scientific Computing* (2nd ed), Cambridge University Press, London.
Rajendran, S. and Liew, K.M. (2003), "A variant of the subspace iteration algorithm for generalized eigenproblems", *Int. J. Numer. Meth. Eng*., **57**, 2027-2042.
Underwood, R. (1975), "An iterative block Lanczos method for the solution of large sparse symmetric eigenproblems", Ph.D. Dissertation, Stanford University.
Wilson, E.L. and Itoh, T. (1983), "An eigensolution strategy for large systems", *Comput. Struct*., **16**, 259-265.
Wilson, E.L., Yuan, M.W. and Dickens, J.M. (1982), "Dynamic analysis by direct superposition of Ritz vectors", *Earthquake Eng. Struct. Dyn.*, **10**, 813-821.
Yamamoto, Y. and Ohtsubo, H. (1976), "Subspace iteration accelerated by using Chebyshev polynomials for eigenvalue problems with symmetric matrices", *Int. J. Numer. Meth. Engg*., **10**, 935-944.
Yang, Y. B., Hung, H. H. and Hsu, L. C. (2008), "Ground vibrations due to underground trains considering soil-tunnel interaction", *Interaction and Multiscale Mechanics*, *An Int. J.*, **1**(1), 157-175.
Yuan, M.W., et al. (2005), *SAP84-A General purpose Program on Microcomputer* (version 6.5), Peking University, Beijing, China (in Chinese).
Yuan, M.W., et al. (1989), "The WYD method in large eigenvalue problems", *Eng. Comput*., **6**, 49-57.
Zhao, Q.C., Chen, P., Peng, W.B., Gong, Y.C. and Yuan M.W. (2007), "Accelerate subspace iteration with aggressive shift", *Comput. Struct.*, **85**(19-20), 1562-1578.

Appendix: Flowchart of the classical subspace iteration method with shift.

I. Initialization
  I.1. Determine subspace dimension q
  I.2. Select initial vectors $\mathbf{X}_0 \in R^{N \times q}$
  I.3. Set up maximum iteration number $I_{\max}$ for each $\mathbf{LDL}^T$ decomposition
  I.4. $k = 0$

II. Shifting and Sturm sequence checking
  II.1. Compute shifting μ, which is not an eigenvalue
  II.2. $\mathbf{LDL}^T$ decomposition for $\mathbf{K} - \mu\mathbf{M} = \mathbf{LDL}^T$
  II.3. Perform Sturm sequence check

III. Perform $I_{\max}$ iterations, and then go to II
  III.1. $k = k + 1$
  III.2. Solve trial vectors $\bar{X}_k = (\mathbf{LDL}^T)^{-1}\mathbf{M}X_{k-1}$
  III.3. Find the projection of the matrices $\mathbf{K}$ and $\mathbf{M}$, $\mathbf{K}^* = \bar{X}_k^T\mathbf{K}\bar{X}_k$, $\mathbf{M}^* = \bar{X}_k^T\mathbf{M}\bar{X}_k$,
  III.4. Solve the eigensystem for the projected matrices, $\mathbf{K}^*\ \mathbf{Q}_k^* = \mathbf{M}^*\ \mathbf{Q}_k^*\Lambda^*$, $(\mathbf{Q}_k^*)^T\mathbf{M}^*\ \mathbf{Q}_k^* = \mathbf{I}$
  III.5. Find an improved approximation to eigenvectors $\mathbf{X}_k = \bar{\mathbf{X}}_k\ \mathbf{Q}_k^*$
  III.6. Check convergence, move out converged eigenvectors and if necessary, add random vectors to $\mathbf{X_k}$
  III.7. If the number of the converged eigenpairs has reached the required ones, stop the process