AIAA JOURNAL
Vol. 34, No. 7, July 1996

Two-Step Component-Mode Synthesis for the Eigensolution of Large Systems

Anand Ramani* and Charles E. Knight†
Virginia Polytechnic Institute and State University, Blacksburg, Virginia 24061

Component-mode synthesis is a dynamic substructuring technique for approximate eigensolutions of large degree-of-freedom systems divisible into two or more components. System synthesis using component modes results in approximate eigenparameters. The approximation is improved by using more component modes in synthesis. Typically, all component modes up to x times the frequency range of interest are required to ensure reasonable accuracy, where x varies from 2 to 10. A two-step component-mode synthesis approach is presented. The first step involves system synthesis using the minimum number of component modes required to obtain approximate eigenvalues up to a preselected frequency, and the second step uses additional component modes in a convergence scheme operating on the system eigenparameters calculated in the first step. This results in an infinite-series eigenvalue problem that is solved by a modified shooting method. A perturbation approach is adopted to derive the convergence scheme, in which separate iterations are performed for each eigenvalue and eigenvector in each step. Selection of initial values for the convergence scheme is also presented. In a few instances, an eigenvalue nearly reached the convergence tolerance, but then wandered away and converged to an adjacent value. The algorithm was modified to automatically detect this problem and abort iteration with a small penalty in accuracy. This two-step approach provides the analyst with some insight into the convergence and accuracy of the eigenparameters without resolving an increasingly larger system over and over with more component modes. Numerical examples are included to demonstrate the applicability of the method.

Nomenclature

I	= identity matrix
K	= system stiffness matrix
M	= system mass matrix
m	= mass submatrix
N	= functional λ matrix
$N'(\lambda)$	= derivative of N with respect to λ
$\boldsymbol{p}$	= system degree-of-freedom vector
$\boldsymbol{q}$	= transformed system degree-of-freedom vector
ε	= ordering parameter
Λ	= diagonal matrix of eigenvalues, matrix of all eigenvalues of the initial system
λ	= eigenvalue
Φ	= matrix of all eigenvectors of the initial system

Subscripts

c	= connection coordinate, correction
e	= extra normal-mode coordinate
n	= normal-mode coordinate

Superscripts

k	= iteration counter
T	= transpose

Introduction

EIGENPROBLEM solution of large degree-of-freedom (DOF) systems has been of considerable interest for several years. Even with the increasing efficiency of computers and reducing computing and storage costs, dynamic analysis of very large DOF systems can pose a stiff challenge to analysts. Component-mode synthesis is a dynamic condensation technique for the eigensolution of large DOF systems; these methods have the managerial advantage of allowing different design groups to work with different parts of a large structure. This is particularly advantageous if a structure can be divided into natural components. Each component is analyzed independently for natural frequencies and mode shapes. The component mode shapes are then assembled along with other modes and are used to transform the original displacement degrees of freedom to generalized DOF. With a truncated set of component normal modes, this transformation reduces the size of the final eigenvalue problem, which can be solved economically.

Hurty[1] is credited with first presenting the method of component-mode synthesis in 1965. Craig and Chang[2] introduced the concept of attachment modes in substructure coupling. Hintz[3] defined two statically complete interface mode sets—the attachment mode set and the constraint mode set—and discussed how they can be combined with free and fixed-interface normal modes to give dynamic mode supersets. MacNeal[4] developed a hybrid method of component-mode synthesis with some interface degrees of freedom fixed and some free. Rubin[5] used free-interface normal modes augmented with a low-frequency account for the contribution of residual modes. Craig and Bampton[6] modified Hurty's method. Craig[7] presented a review of time-domain and frequency-domain component-mode synthesis methods in 1976. Herting[8] developed a general-purpose program for component-mode synthesis and defined a type of inertia-relief mode. Craig and Chang[9] put forth the generalized procedure of substructure coupling, which is widely used today. The methodology of component-mode synthesis is also described in these references. Several types of component modes and mode supersets are used for synthesis and are discussed in these references.

Theoretical Development

Following the usual procedure of modal transformation using component modes and assembly using constraint equations for interface compatibility, the equations of motion for free vibrations of an undamped system are derived as

$$M\ddot{p} + Kp = 0 \tag{1}$$

where $\boldsymbol{p}$ is the system generalized displacement vector. The structure of the system matrices depends on the mode superset used in the synthesis. For synthesis using constraint mode supersets consisting

Received Jan. 3, 1995; revision received March 10, 1996; accepted for publication March 18, 1996. Copyright © 1996 by the American Institute of Aeronautics and Astronautics, Inc. All rights reserved.

*Ph.D. Student, Department of Mechanical Engineering.

†Professor, Department of Mechanical Engineering.

of fixed-interface normal modes (mass normalized) and constraint modes, the system matrices are given by

$$K = \begin{bmatrix} K_{cc} & 0 \\ 0 & \Lambda_{nn} \end{bmatrix} \quad \text{and} \quad M = \begin{bmatrix} M_{cc} & M_{cn} \\ M_{nc} & I_{nn} \end{bmatrix} \tag{2}$$

Λ_{nn} is a diagonal matrix of component eigenvalues and I_{nn} is the identity matrix of order n. The system displacement vector is given by

$$p = \begin{Bmatrix} p_c \\ p_n \end{Bmatrix} \tag{3}$$

The common approach to solve for system modes up to a cutoff frequency ω_c begins by solving for the normal modes of each component up to a multiple of the cutoff frequency $x\omega_c$, where, in general, $x > 2$. Conventional methods of component-mode synthesis may use values of x up to 10 to ensure accuracy in the calculated system modes. When all of the calculated component modes are assembled, the system eigenproblem is solved in one step, usually for all of the modes with a frequency less than or equal to ω_c. Thus, in this one-step solution, the accuracy of the calculated system modes cannot be estimated. To study the convergence of the calculated system modes, additional component modes will have to be introduced. This also means that the component-mode synthesis problem has to be reformulated using the additional modes. The result is a larger eigenvalue problem that has to be solved. It will be better if the system solution can be done using fewer of the component modes followed by the introduction of additional component modes in a convergence scheme operating on the initially calculated system modes. This is possible when constraint mode supersets are used in system synthesis. To converge the initial system eigenvalues and eigenvectors, let a set of e extra component normal modes be introduced. The resulting system stiffness and mass matrices can be partitioned as

$$K' = \begin{bmatrix} K_{cc} & 0 & 0 \\ 0 & \Lambda_{nn} & 0 \\ 0 & 0 & \Lambda_{ee} \end{bmatrix} \quad \text{and} \quad M' = \begin{bmatrix} M_{cc} & M_{cn} & M_{ce} \\ M_{nc} & I_{nn} & 0 \\ M_{ec} & 0 & I_{ee} \end{bmatrix} \tag{4}$$

where Λ_{ee} is a diagonal matrix of the extra component eigenvalues and I_{ee} is the identity matrix of order e. The system displacement vector is now given by

$$p' = \begin{Bmatrix} p_c \\ p_n \\ p_e \end{Bmatrix} \tag{5}$$

where $\boldsymbol{p}_e$ is the system generalized displacement vector corresponding to the extra component modes.

In the two-step component-mode synthesis method presented here, the initial eigenvalue problem with the matrices given by Eq. (2) is first solved. To reform the equations to include the extra modes, as in Engels,[10] the transformation

$$p = \Phi \boldsymbol{q} \tag{6}$$

is introduced. The transformation

$$\begin{Bmatrix} p_c \\ p_n \\ \cdots\cdot \\ p_e \end{Bmatrix} = \begin{bmatrix} \Phi & 0 \\ 0 & I_{ee} \end{bmatrix} \begin{Bmatrix} q \\ p_e \end{Bmatrix} \tag{7}$$

is used to obtain the system eigenvalue problem with the additional modes as

$$\left(\begin{bmatrix} \Lambda & 0 \\ 0 & \Lambda_{ee} \end{bmatrix} - \lambda \begin{bmatrix} I & m_{ce} \\ m_{ce}^T & I_{ee} \end{bmatrix} \right) \begin{Bmatrix} q \\ p_e \end{Bmatrix} = \begin{Bmatrix} 0 \\ 0 \end{Bmatrix} \tag{8}$$

where

$$m_{ce} = \Phi^T \begin{bmatrix} M_{ce} \\ 0 \end{bmatrix} \tag{9}$$

and λ is the system eigenvalue including the extra modes. Solution of this eigenvalue problem for the lowest system eigenvalues up to the cutoff frequency results in improved eigenvalues compared to those obtained by the first solution. This would, however, require solution of the larger system matrices. Alternatively, if a method for convergence of the initial eigenvalues using the e additional component modes can be developed that does not handle the larger system matrices, the method would be more attractive.

So far, the only efforts directed toward converging initially calculated system modes have been made by Hasselman and Hart[11] and Engels.[10] Methods developed in these references require that the set e of extra component modes include all of the remaining modes of all of the remaining components. This is not practical in a component-mode synthesis scheme intended for large DOF systems.

Consider the eigenvalue problem described by Eq. (8). Solving for $\boldsymbol{p}_e$ from the lower partition and substituting in the upper partition produces

$$N(\lambda)\boldsymbol{q} = 0 \tag{10}$$

where

$$N(\lambda) \equiv N = \left(\Lambda - \lambda I - \lambda^2 m_{ce}[\Lambda_{ee} - \lambda I]^{-1} m_{ce}^T\right) \tag{11}$$

The element in the ith row and jth column of the matrix N, given by

$$N(i, j) = \Lambda(i, i)\delta_{ij} - \lambda\delta_{ij} - \lambda^2 \sum_{r=1}^{e} \frac{m_{ce}(i, r) m_{ce}(j, r)}{\Lambda_{ee}(r, r) - \lambda} \tag{12}$$

where

$$\delta_{ij} = \begin{cases} 1 & \text{for} \quad i = j \\ 0 & \text{for} \quad i \neq j \end{cases}$$

and N is symmetric. The third term can be expanded in terms of powers of λ, so that $N(\lambda)$ can be written as

$$N(\lambda) = N_0 + \lambda N_1 + \lambda^2 N_2 + \lambda^3 N_3 \cdots \tag{13}$$

where

$$N_0(i, j) = \Lambda(i, i)\delta_{ij} \tag{14}$$

$$N_1(i, j) = -\delta_{ij} \tag{15}$$

and

$$N_s(i, j) = -\sum_{r=1}^{e} \frac{m_{ce}(i, r) m_{ce}(j, r)}{\Lambda_{ee}(r, r)^{s-1}} \qquad s = 2, 3, \ldots \tag{16}$$

The series is convergent if $\lambda < \min\{\operatorname{diag}(\Lambda_{ee})\}$.

Several methods exist in the literature for solving the nonlinear eigenvalue problem described by Eq. (10). However, these methods either involve expensive factorizations, forward and backward substitutions, or a truncation of the infinite series up to a small finite power of λ, along with the formation of block-companion matrices.

Some of the algorithms in the literature use the vanishing determinant property of the λ matrix at the exact eigenvalues. Kublanovskaya[12] uses a unitary transformation to obtain a scalar equation in terms of the eigenvalue without evaluating the determinant explicitly, followed by the application of the Newton–Raphson iteration for root finding. Leung's algorithm[13] is also based on a similar approach, with the eigenvectors obtained by inverse iteration. An algorithm by Thurston[14] is based on linearization of the nonlinear eigenvalue about an approximate root and solving many linear eigenvalue problems till convergence is obtained. Each eigenvalue is converged separately. Ruhe[15] studied algorithms that were extensions of those used for linear eigenvalue problems, such as inverse iteration and the QR algorithm. He also considered an algorithm that reduces the nonlinear eigenvalue problem into a series of linear problems. Rajakumar[16] uses a Lanczos two-sided recursion algorithm with bi-orthogonal transformations for quadratic eigenvalue problems.

Moler and Stewart[17] and Fricker[18] have developed algorithms based on block-companion matrices. These matrices are obtained using the coefficient matrices of the nonlinear matrix eigenvalue problem and essentially reduce the nonlinear problem to a linear problem of larger order. Moler and Stewart[17] solved the resulting linear problem using a QZ algorithm. Fricker[18] defined master and slave coordinates for a reduction transformation based on a cutoff frequency. The resulting problem is cast in a block-companion matrix form for solution using a linear algorithm. Jain et al.[19] developed an algorithm for finding only the eigenvalues through a triangularization scheme for calculating the determinant. Yang[20] also used a similar factorization for evaluating the determinant. Both methods were able to calculate the eigenvalues only. Osborne and Michaelson[21] developed a method of iteration similar to the method to be developed here, but their method involves coupling between the eigenvalue and eigenvector. This requires factorizations and forward and backward substitutions at each stage of the iteration.

Methods involving the formation of block-companion matrices are not useful in our case because we need to find exact solutions to the infinite-series eigenvalue problem. Methods involving factorization and forward and backward substitutions are expensive to implement and inherit all of the numerical difficulties associated with the factorization of matrices, such as ill-conditioning and near-zero pivots.

There is a need for a solution scheme with iteration expressions that do not involve coupling between the eigenvalue and the eigenvector, because such a coupling would require matrix factorizations during iteration. Here, the objective is to develop a simple method that takes advantage of the special structure of the matrices N_0 and N_1 in $N(\lambda)$. Also, if the eigenvalue–eigenvector dependence is decoupled during iteration, the solution algorithm would not require matrix factorization during each iteration.

Convergence Scheme

The convergence scheme is developed using a perturbation approach to obtain a basic equation from which iteration expressions are to be derived. Eigenvalue–eigenvector dependence is decoupled in the derivation of the iteration expressions, and this results in two iteration expressions for convergence, one each for the eigenvalue and the eigenvector.

Assume that we know approximate values λ_0 and $\boldsymbol{q}_0$. Then, in general, λ_0 and $\boldsymbol{q}_0$ will not satisfy Eq. (10). We then seek a correction λ_c to λ_0 and $\boldsymbol{q}_c$ to $\boldsymbol{q}_0$ that will satisfy Eq. (10). Let

$$\lambda = \lambda_0 + \varepsilon\lambda_c, \qquad \boldsymbol{q} = \boldsymbol{q}_0 + \varepsilon\boldsymbol{q}_c \tag{17}$$

Using ε for ordering indicates the relative magnitude of each term. Ordering Eq. (13) for N results in

$$N(\lambda) = N_0 + \lambda N_1 + \varepsilon\lambda^2 N_2 + \varepsilon^2\lambda^3 N_3 \cdots \tag{18}$$

Substituting for λ and $\boldsymbol{q}$ from Eq. (17) in Eq. (10) and using the expansion for N given by Eq. (18) results in

$$\begin{aligned}&\left[N_0 + \lambda_0 N_1 + \varepsilon\lambda_0^2 N_2 + \varepsilon^2\lambda_0^2 N_3 + \cdots\right][\boldsymbol{q}_0 + \varepsilon\boldsymbol{q}_c]\\ &\quad + \varepsilon\lambda_c\left[N_1 + 2\varepsilon\lambda_0 N_2 + 3\varepsilon^2\lambda_0^2 N_3 + \cdots\right][\boldsymbol{q}_0 + \varepsilon\boldsymbol{q}_c]\\ &\quad + \varepsilon^2\lambda_c^2\left[\varepsilon N_2 + 3\varepsilon^2\lambda_0 N_3 + \cdots\right][\boldsymbol{q}_0 + \varepsilon\boldsymbol{q}_c] + \cdots = 0\end{aligned} \tag{19}$$

Neglecting all terms with powers of ε greater than one,

$$N(\lambda_0)\boldsymbol{q}_0 + \varepsilon N(\lambda_0)\boldsymbol{q}_c + \varepsilon\lambda_c N'(\lambda_0)\boldsymbol{q}_0 = 0 \tag{20}$$

where $N'(\lambda)$ is given by

$$N'(\lambda) = N_1 + 2\varepsilon\lambda N_2 + 3\varepsilon^2\lambda^2 N_3 \cdots \tag{21}$$

Equation (20) represents a system of equations with one more unknown than the number of equations. This is the case with the solution of any eigenvalue problem, because the eigenvector can be multiplied by an arbitrary scale factor. The problem is resolved by choosing to normalize the eigenvector $\boldsymbol{q}$ in a specific manner described later. To decouple the dependence between the eigenvalue and the eigenvector during convergence, two different iteration expressions are derived. One is for the eigenvalue, assuming that the eigenvector is known, and the other is for the eigenvector, assuming that the eigenvalue is known.

Solution for the eigenvalue assuming that the eigenvector is known: If, in Eq. (20), it is assumed that the eigenvector $\boldsymbol{q}$ is known, i.e., $\boldsymbol{q}_0 = \boldsymbol{q}$ and $\boldsymbol{q}_c = 0$, then the equation reduces to

$$N(\lambda_0)\boldsymbol{q} + \varepsilon\lambda_c N'(\lambda_0)\boldsymbol{q} = 0 \tag{22}$$

Premultiplying by $\boldsymbol{q}^T$ and solving for λ_c, we obtain

$$\lambda_c = -\frac{\boldsymbol{q}^T N(\lambda_0)\boldsymbol{q}}{\varepsilon\boldsymbol{q}^T N'(\lambda_0)\boldsymbol{q}} = -\frac{\boldsymbol{q}^T N(\lambda_0)\boldsymbol{q}}{\boldsymbol{q}^T N'(\lambda_0)\boldsymbol{q}} \tag{23}$$

where the ordering parameter ε is removed. Therefore,

$$\lambda = \lambda_0 - \frac{\boldsymbol{q}^T N(\lambda_0)\boldsymbol{q}}{\boldsymbol{q}^T N'(\lambda_0)\boldsymbol{q}} \tag{24}$$

is a better approximation to the actual eigenvalue corresponding to Eq. (10). Hence, if $\boldsymbol{q}$ is known, we can iterate for λ using the above equation and progressively find better eigenvalues. Used in an iterative scheme,

$$\lambda^{(k+1)} = \lambda^{(k)} - \frac{\boldsymbol{q}^T N\left[\lambda^{(k)}\right]\boldsymbol{q}}{\boldsymbol{q}^T N'\left(\lambda^{(k)}\right)\boldsymbol{q}} \tag{25}$$

Note, however, that in Eq. (25), $N'[\lambda^{(k)}]$ has to be computed in every iteration. This can be avoided by substituting for $N'(\lambda)$ from Eq. (21) in Eq. (22) to get

$$N(\lambda_0)\boldsymbol{q} + \varepsilon\lambda_c\left(N_1 + 2\varepsilon\lambda_0 N_2 + 3\varepsilon^2\lambda_0^2 N_3 \cdots\right)\boldsymbol{q} = 0 \tag{26}$$

and assuming in the parenthetical expression in the second term that the contribution of terms with powers of ε is small compared to N_1, the equation

$$N(\lambda_0)\boldsymbol{q} + \varepsilon\lambda_c N_1\boldsymbol{q} = 0 \tag{27}$$

is obtained. Premultiplying Eq. (27) by $\boldsymbol{q}^T$ and solving for λ_c,

$$\lambda_c = -\frac{\boldsymbol{q}^T N(\lambda_0)\boldsymbol{q}}{\boldsymbol{q}^T N_1\boldsymbol{q}} \tag{28}$$

where, again, ε has been removed. Now, if $\boldsymbol{q}$ is always normalized such that $\boldsymbol{q}^T\boldsymbol{q} = 1$, then

$$\lambda_c = \boldsymbol{q}^T N(\lambda_0)\boldsymbol{q} \tag{29}$$

because $N_1 = -I$, the identity matrix, and $\boldsymbol{q}^T N_1\boldsymbol{q} = -\boldsymbol{q}^T\boldsymbol{q} = -1$. Hence, the iteration scheme for λ can be modified as

$$\lambda^{(k+1)} = \lambda^{(k)} + \boldsymbol{q}^T N\left(\lambda^{(k)}\right)\boldsymbol{q} \tag{30}$$

and this is more attractive than Eq. (25) because it avoids the computation of $N'[\lambda^{(k)}]$.

Solution for the eigenvector, assuming that the eigenvalue is known: If the eigenvalue λ is assumed to be known, i.e., $\lambda = \lambda_0$ and $\lambda_c = 0$, then Eq. (20) becomes

$$N(\lambda)\boldsymbol{q}_0 + \varepsilon N(\lambda)\boldsymbol{q}_c = 0 \tag{31}$$

Solving for $\boldsymbol{q}_c$,

$$\boldsymbol{q}_c = -\{\varepsilon N(\lambda)\}^{-1} N(\lambda)\boldsymbol{q}_0 \tag{32}$$

Note that $N(\lambda)$ is singular (because λ is the exact eigenvalue) and hence $\boldsymbol{q}_c$ cannot be determined. Even if λ were not known, this equation would give $\boldsymbol{q}_c = -\boldsymbol{q}_0$ and this is undesirable. Recognize that from the expansion for $N(\lambda)$, $\varepsilon N(\lambda)$ can be written as

$$\varepsilon N(\lambda) = \left(\varepsilon N_0 + \varepsilon\lambda N_1 + \varepsilon^2\lambda^2 N_2 + \varepsilon^3\lambda^3 N_3 \cdots\right) \tag{33}$$

and retaining only terms with the first power of ε for the term $\{\varepsilon N(\lambda)\}$ in Eq. (32) gives

$$\boldsymbol{q}_c = -(\varepsilon N_0 + \varepsilon\lambda N_1)^{-1} N(\lambda)\boldsymbol{q}_0 \tag{34}$$

Table 1 Number of initial and additional modes used for examples

Case	Cutoff frequency ω_c, Hz	Total number of component normal modes used for initial solution	Highest frequency of component modes used for initial solution, Hz	(1)[a] Total number of additional modes used for convergence, $2\omega_c$	(1)[a] Highest frequency of additional modes used for convergence, Hz	(2)[b] Total number of additional modes used for convergence, $4\omega_c$	(2)[b] Highest frequency of additional modes used for convergence, Hz
1	560	23	609.1	10	1278	30	2420
2	475	23	557.7	12	1197	31	2282
3	560	23	617.6	9	1229	30	2499
4	475	23	570.2	12	1229	29	2255

[a] Two times the cutoff frequency. [b] Four times the cutoff frequency.

or with the ordering parameter ε removed,

$$q_c = -(N_0 + \lambda N_1)^{-1} N(\lambda) q_0 \tag{35}$$

Therefore,

$$q = q_0 - (N_0 + \lambda N_1)^{-1} N(\lambda) q_0 \tag{36}$$

is a better approximation to the eigenvector. When used iteratively,

$$q^{(k+1)} = q^{(k)} - (N_0 + \lambda N_1)^{-1} N(\lambda) q^{(k)} \tag{37}$$

Because the matrix $(N_0 + \lambda N_1)$ is diagonal, its inverse can be computed easily.

Solution for the eigenvector and the eigenvalue starting from approximate initial values: If both the eigenvalue and the eigenvector are unknown, the above iteration schemes can be used to simultaneously converge both λ and q. Then, an iteration becomes

$$q^{(k+1)} = q^{(k)} - \left(N_0 + \lambda^{(k)} N_1\right)^{-1} N\left(\lambda^{(k)}\right) q^{(k)} \tag{38}$$

$$\lambda^{(k+1)} = \lambda^{(k)} + q^{(k)T} N\left(\lambda^{(k)}\right) q^{(k)} \tag{39}$$

That the above equations will, in fact, lead to convergence of both the eigenvalue and eigenvector can be reasoned as follows.

In the solution scheme, values for λ and q are sought that satisfy Eq. (10). If the entire vector q is considered as a single entity, the solution of Eq. (10) can be interpreted as the search for the zeroes of a function of two variables. For explanatory purposes, consider a scalar function f of two variables x and y, whose zeroes are to be found, i.e., x and y are sought such that

$$f(x, y) = 0 \tag{40}$$

If an approximate solution (x_0, y_0) is known, then

$$f(x, y) = 0 = f(x_0, y_0) + (x - x_0) f_x(x_0, y_0) + (y - y_0) f_y(x_0, y_0) \tag{41}$$

where $f_x = (\partial f/\partial x)$; $f_y = (\partial f/\partial y)$ and the higher-order terms have been neglected. A better value for x is obtained by temporarily disregarding the variation of f with y, i.e., by searching for a solution for Eq. (40) along the plane $y = y_0$, which solves the equation

$$f(x_0, y_0) = (x - x_0) f_x(x_0, y_0) \tag{42}$$

for x. Similarly, by temporarily disregarding the variation of f with x and looking for a solution along the plane $x = x_0$, a better value for y is obtained.

Equation (10) compares to Eq. (40) and the two individual solution schemes for the eigenvalue and eigenvector, assuming that the other is known, are, in reality, a search for a better solution along the hyperplanes $q = q_0$ and $\lambda = \lambda_0$, respectively. In other words, separate iterations are performed in the q-subspace and λ-subspace, respectively. Strictly speaking, the proposed schemes are slightly modified versions of this search, because the expressions have been simplified for better computational efficiency.

Initial Values

Because the first eigenproblem with its eigenvalues and eigenvectors has been solved, the convergence scheme for the ith eigenvalue and eigenvector can be started with the initial values

$$\lambda_i^{(0)} = \Lambda(i, i)$$

$$q_i^{(0)} = \langle 0 \quad 0 \cdots 0 \quad 1 \quad 0 \cdots 0 \quad 0 \rangle^T \tag{43}$$

where $q_i^{(0)}$ has 1 in the ith position. However, $(N_0 + \lambda_i^{(0)} N_1)$ is singular, and so, convergence must be started with the eigenvalue equation to determine $\lambda_i^{(1)}$. Therefore, to start convergence, iteration for the eigenvalue must be performed first. Thereafter, Eqs. (38) and (39) can be used in turn to converge both the eigenvalue and the eigenvector.

Advantages of the Two-Step Method

1) The two-step method for system eigensolution up to a cutoff frequency ω_c uses the smallest possible eigenproblem.

2) Reformulation and resolution of the larger eigenproblem when additional modes are introduced is eliminated in the two-step solution.

3) Indication of convergence of the system eigenvalues and eigenvectors is shown.

4) The method lends itself to easy computer implementation and does not involve expensive matrix decompositions and solutions of a linear system of equations, which are currently used to solve the nonlinear eigenvalue problem described by Eq. (10).

5) If, after the second step, it is found that some eigenvalues do not satisfy the analyst's accuracy requirements, the second step can be repeated with additional component modes.

Examples

The two-step component-mode synthesis method was used to synthesize the eigenparameters of a free–free beam shown in Fig. 1, from three beam components, connected end to end. The second beam component played the role of a connector. Four cases were studied.

Finite element models for the first two cases were constructed with beam elements. Component 1 was discretized into 80 elements, component 2 into 40, and component 3 into 110 elements. For case 1, an elastic modulus of 2×10^{11} Pa and a mass density of 7800 kg/m^3 was used for all of the components. In case 2, the connector component was softened to have an elastic modulus of 2×10^9 Pa and a mass density of 2600 kg/m^3.

Cases 3 and 4 were similar to cases 1 and 2, respectively, but used shell elements instead of beam elements for the finite element models. A regular mesh pattern with six elements across the width of the beam was used; component 1 was discretized into 360 elements, component 2 into 120, and component 3 into 480 elements.

In all cases, a fully assembled finite element model was constructed to provide reference frequencies and mode shapes for verification. This full model was constructed by combining the models of the individual components and hence inherited the same mesh pattern as the components. Subspace iteration was used to obtain the frequencies of the full model with a convergence tolerance of six significant digits. Results from the full model were used for

Table 2 Results for example case 1

No.	Reference frequencies, Hz	Initial frequencies, Hz	Difference from reference, %	(1) Converged frequencies, Hz	Difference from reference, %	(2) Converged frequencies, Hz	Difference from reference, %
1	5.20497	5.20497	0.00	5.20497	0.00	5.20497	0.00
2	14.3482	14.3483	0.00	14.3482	0.00	14.3482	0.00
3	28.1302	28.1306	0.00	28.1304	0.00	28.1302	0.00
4	46.5052	46.5070	0.00	46.5061	0.00	46.5053	0.00
5	69.4829	69.4868	0.01	69.4836	0.00	69.4830	0.00
6	78.0192	78.0561	0.05	78.0272	0.01	78.0202	0.00
7	97.0584	97.0756	0.02	97.0678	0.01	97.0589	0.00
8	129.257	129.291	0.03	129.264	0.01	129.258	0.00
9	166.054	166.129	0.05	166.090	0.02	166.055	0.00
10	204.462	204.533	0.03	204.468	0.00	204.464	0.00
11	207.493	207.653	0.08	207.547	0.03	207.498	0.00
12	214.806	214.973	0.08	214.963	0.07	214.827	0.01
13	253.558	253.922	0.14	253.652	0.04	253.567	0.00
14	304.246	304.581	0.11	304.450	0.07	304.254	0.00
15	359.642	361.341	0.47	359.916	0.08	359.687	0.01
16	408.935	416.086	1.75	409.874	0.23	409.152	0.05
17	419.612	420.139	0.13	420.110	0.12	419.616	0.00
18	420.385	437.729	4.13	421.726	0.32	420.482	0.02
19	484.370	490.201	1.20	485.126	0.16	484.492	0.03
20	553.714	556.974	0.59	554.866	0.21	553.766	0.01

Table 3 Results for example case 2

No.	Reference frequencies, Hz	Initial frequencies, Hz	Difference from reference, %	(1)[a] Converged frequencies, Hz	Difference from reference, %	(2)[b] Converged frequencies, Hz	Difference from reference, %
1	1.51202	1.51203	0.00	1.51203	0.00	1.51203	0.00
2	12.4811	12.4812	0.00	12.4811	0.00	12.4811	0.00
3	22.6685	22.6695	0.00	22.6688	0.00	22.6686	0.00
4	23.8279	23.8283	0.00	23.8280	0.00	23.8279	0.00
5	39.2530	39.2558	0.01	39.2535	0.00	39.2531	0.00
6	55.5810	55.5902	0.02	55.5823	0.00	55.5813	0.00
7	58.4602	58.4720	0.02	58.4625	0.00	58.4605	0.00
8	87.9478	87.9625	0.02	87.9511	0.00	87.9484	0.00
9	99.6708	99.7298	0.06	99.6789	0.01	99.6729	0.00
10	151.506	151.554	0.03	151.517	0.01	151.509	0.00
11	173.798	173.962	0.09	173.825	0.02	173.804	0.00
12	185.334	185.505	0.09	185.442	0.06	185.352	0.01
13	224.687	224.838	0.07	224.712	0.01	224.695	0.00
14	277.029	277.246	0.08	277.062	0.01	277.039	0.00
15	311.719	312.264	0.17	311.780	0.02	311.739	0.01
16	352.180	358.336	1.75	353.167	0.28	352.307	0.04
17	370.908	375.150	1.14	371.870	0.26	371.024	0.03
18	375.088	377.340	0.60	375.271	0.05	375.121[c]	0.01
19	424.880	426.184	0.31	425.000	0.03	424.937	0.01
20	463.394	469.617	1.34	464.213	0.18	463.541	0.03

[a] Two times the cutoff frequency. [b] Four times the cutoff frequency. [c] Iteration aborted.

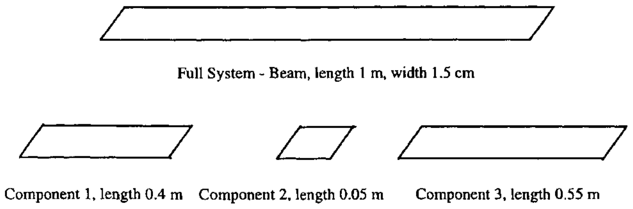
Full System - Beam, length 1 m, width 1.5 cm
Component 1, length 0.4 m
Component 2, length 0.05 m
Component 3, length 0.55 m

Fig. 1 Full system and components used for the examples.

reference and for comparison with those from two-step component-mode synthesis.

Next, two-step component-mode synthesis was performed with the cutoff frequency selected to give the first 20 frequencies in all four cases. The number of component modes used for the initial solution up to the cutoff frequency and the total number of additional modes used for convergence are reported in Table 1. Two different $x\omega_c$ frequencies for the additional modes about two and about four times the cutoff frequency were studied so that convergence could be examined. These frequency limits are designated (1) and (2), respectively, and are also reported in Table 1. A convergence tolerance of six significant digits was used for the iterative scheme.

The initial and converged frequencies are reported in Tables 2–5 along with the reference frequencies for each of the four cases. For verification, conventional component-mode synthesis using the Jacobi method and the subspace iteration method were performed for all four cases using component modes up to about two and four times the cutoff frequencies (the same number that was used in the two-step method), and the frequencies thus obtained were almost identical (maximum difference, 0.0008%).

A small problem occurred in the iterative process, which caused one or two eigenvalues to converge to an adjacent value. The eigenvalue would approach the correct value closely but then wander off to the adjacent value before the six-digit tolerance was reached. The eigenvalue strayed when the value of the corresponding element of the vector $\boldsymbol{q}$, which begins at unity, suddenly fell below an adjacent value. Convergence was aborted when this was found to occur, and

Table 4 Results for example case 3

No.	Reference frequencies, Hz	Initial frequencies, Hz	Difference from reference, %	(1)[a] Converged frequencies, Hz	Difference from reference, %	(2)[b] Converged frequencies, Hz	Difference from reference, %
1	5.20377	5.20377	0.00	5.20377	0.00	5.20377	0.00
2	14.3477	14.3477	0.00	14.3477	0.00	14.3477	0.00
3	28.1372	28.1375	0.00	28.1373	0.00	28.1372	0.00
4	46.5345	46.5363	0.00	46.5355	0.00	46.5346	0.00
5	69.5597	69.5635	0.01	69.5604	0.00	69.5598	0.00
6	78.0993	78.1364	0.05	78.1075	0.01	78.1004	0.00
7	97.2227	97.2393	0.02	97.2318	0.01	97.2232	0.00
8	129.561	129.595	0.03	129.567	0.00	129.562	0.00
9	166.574	166.647	0.04	166.609	0.02	166.576	0.00
10	207.528	207.602	0.04	207.541	0.01	207.530	0.00
11	208.320	208.477	0.08	208.372	0.03	208.324	0.00
12	215.002	215.171	0.08	215.160	0.07	215.023	0.01
13	254.812	255.169	0.14	254.903	0.04	254.820	0.00
14	306.075	306.403	0.11	306.271	0.06	306.083	0.00
15	362.208	363.892	0.47	362.474	0.07	362.250	0.01
16	415.162	422.019	1.65	416.433	0.31	415.330	0.04
17	420.689	423.660	0.71	422.041[c]	0.32	420.788	0.02
18	423.139	438.073	3.53	423.623	0.11	423.142	0.00
19	489.074	494.900	1.19	489.813	0.15	489.189	0.02
20	559.892	563.127	0.58	561.029	0.20	559.940	0.01

[a] Two times the cutoff frequency. [b] Four times the cutoff frequency. [c] Iteration aborted.

Table 5 Results for example case 4

No.	Reference frequencies, Hz	Initial frequencies, Hz	Difference from reference, %	(1) Converged frequencies, Hz	Difference from reference, %	(2) Converged frequencies, Hz	Difference from reference, %
1	1.52642	1.52642	0.00	1.52642	0.00	1.52642	0.00
2	12.5076	12.5076	0.00	12.5076	0.00	12.5076	0.00
3	22.7396	22.7407	0.00	22.7399	0.00	22.7397	0.00
4	23.9039	23.9042	0.00	23.9040	0.00	23.9039	0.00
5	39.3884	39.3911	0.01	39.3889	0.00	39.3886	0.00
6	55.9891	55.9982	0.02	55.9904	0.00	55.9894	0.00
7	63.5989	63.6130	0.02	63.6016	0.00	63.5993	0.00
8	88.1008	88.1150	0.02	88.1041	0.00	88.1016	0.00
9	100.265	100.323	0.06	100.273	0.01	100.267	0.00
10	152.045	152.091	0.03	152.055	0.01	152.048	0.00
11	174.586	174.746	0.09	174.612	0.01	174.594	0.00
12	185.010	185.185	0.09	185.120	0.06	185.029	0.01
13	225.879	226.026	0.07	225.903	0.01	225.887	0.00
14	278.941	279.153	0.08	278.974	0.01	278.952	0.00
15	314.067	314.586	0.17	314.125	0.02	314.089	0.01
16	351.198	357.362	1.76	352.202	0.29	351.333	0.04
17	377.366	381.186	1.01	378.309	0.25	377.484	0.03
18	380.677	382.892	0.58	380.841[a]	0.04	380.709[a]	0.01
19	428.739	429.989	1.29	428.853	0.03	428.796	0.01
20	470.861	477.534	1.42	471.668	0.17	471.034	0.04

[a] Iteration aborted.

the eigenparameters at that stage of iteration were taken as the final values. These frequencies are pointed out in Tables 3–5.

From the tables, it can be observed that there is good general agreement between the reference frequencies and the synthesized frequencies, after convergence. For the initial solution, the maximum difference was 4.1%, and for the approximate $x\omega_c$ frequency ratios of 2:1 between the additional and initial modes, the maximum difference was 0.32%. When this ratio was approximately 4:1, the maximum difference was 0.05%.

Some initial and converged mode shapes (with about 4:1 ratio of additional to initial mode frequency limits) are shown in Figs. 2–5. For example, Fig. 2 shows the initial and converged mode shapes for mode number 19 for case 1, and Figs. 3–5 show sample mode shapes for the other three cases.

A comparison of solution times for two-step component-mode synthesis and conventional component-mode synthesis using the Jacobi method and the subspace iteration method are reported in Table 6 for the 4:1 frequency ratio ($x \cong 4$) between the additional and initial modes. For cases 1 and 2, the two-step method performs in about 50–60% of the time required for conventional component-mode synthesis using a Jacobi solution and in 20–25% of the time required using a subspace iteration solution. For cases 3 and 4, solution times are 74–98% of those for Jacobi solution and subspace iteration solution. This is because of the large number of interface coordinates in the shell-element models requiring considerably more solution time for the complete eigensolution of the initial problem compared to the beam-element models.

Table 6 Solution times for different methods

	Solution time in 0.01 s		
Case	Two-step-method	Jacobi method	Subspace iteration method
1	295	500	1145
2	295	580	1500
3	7050	7142	8330
4	6905	9302	8588

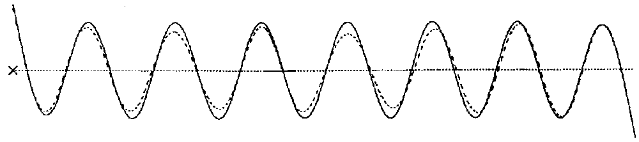

Fig. 2 Initial and final mode shapes for case 1, mode number 19: - - -, initial mode shape and ——, converged mode shape.

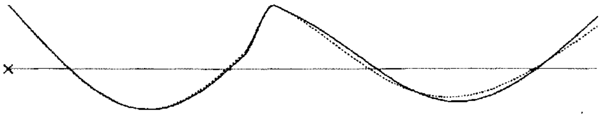

Fig. 3 Initial and final mode shapes for case 2, mode number 16: - - -, initial mode shape and ——, converged mode shape.

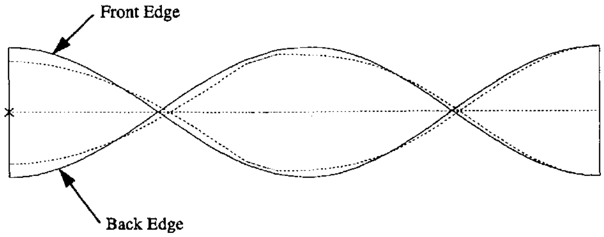

Fig. 4 Initial and final mode shapes for case 3, mode number 16: - - -, initial mode shape and ——, converged mode shape.

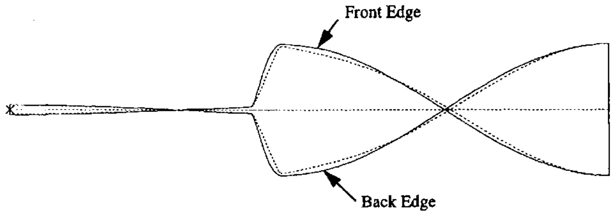

Fig. 5 Initial and final mode shapes for case 4, mode number 17: - - -, initial mode shape and ——, converged mode shape.

Conclusions

A two-step component-mode synthesis method has been developed and tested on some example problems. The method possesses all of the advantages mentioned earlier, viz., minimal order of system matrices, easy computer implementation, and savings in computational time. The method is an effective component-mode synthesis tool and gives the analyst a knowledge of the convergence of the initial system eigenparameters and the freedom to select the number of modes for convergence. On occasion, there was a problem with convergence, when an eigenvalue–eigenvector pair would wander from their true values and instead converge to an adjacent eigenvalue and eigenvector. This was easily detected in the algorithm by monitoring the ith element value of the $\boldsymbol{q}$ vector in Eq. (43). When the value fell below an adjacent value, iteration was aborted and the current values of the eigenvalue and eigenvector were considered to be the converged values. This resulted in a small loss of accuracy in the test cases, but was considered tolerable. Regarding solution time in general, the method is effective for component-mode synthesis and is comparable to other conventional methods.

Savings in computational time is less significant for systems with a large number of connection coordinates compared to normal-mode coordinates, because, in the current approach, the initial eigenproblem has to be solved completely. Approximately converged frequencies could be obtained in less solution time if the initial eigenproblem is only partially solved, resulting in a smaller order of the nonlinear eigenvalue problem, and this forms the scope for future research.

References

[1] Hurty, W. C., "Dynamic Analysis of Structural Systems Using Component Modes," *AIAA Journal*, Vol. 3, No. 4, 1965, pp. 678–685.

[2] Craig, R. R., Jr., and Chang, C-J., "On the Use of Attachment Modes in Substructure Coupling for Dynamic Analysis," *Proceedings of the AIAA/ASME 18th Structures, Structural Dynamics, and Materials Conference*, Vol. B, AIAA, New York, 1977, pp. 89–99.

[3] Hintz, R. M., "Analytical Methods in Component Mode Synthesis," *AIAA Journal*, Vol. 13, No. 8, 1975, pp. 1007–1016.

[4] MacNeal, R. H., "A Hybrid Method of Component Mode Synthesis," *Computers and Structures*, Vol. 1, No. 4, 1971, pp. 581–601.

[5] Rubin, S., "Improved Component-Mode Representation for Structural Dynamic Analysis," *AIAA Journal*, Vol. 13, No. 8, 1975, pp. 995–1006.

[6] Craig, R. R., Jr., and Bampton, M. C. C., "Coupling of Substructures for Dynamic Analysis," *AIAA Journal*, Vol. 6, No. 7, 1968, pp. 1313–1319.

[7] Craig, R. R., "A Review of Time Domain and Frequency Domain Component Mode Synthesis Method," *Combined Experimental and Analytical Modeling of Dynamic Structural Systems* (Albuquerque, NM), American Society of Mechanical Engineers, New York, June 1985, pp. 1–30.

[8] Herting, D. N., "A General Purpose Multi-Stage, Component Modal Synthesis Method," *Finite Elements in Analysis and Design*, Vol. 1, No. 2, 1985, pp. 153–164.

[9] Craig, R. R., Jr., and Chang, C-J., "A Review of Substructure Coupling Methods for Dynamic Analysis," Society for Engineering Science 13th Annual Meeting, *Advances in Engineering Science*, Vol. 2, NASA CP-2001, 1976, pp. 393–408.

[10] Engels, R. C., "Convergence Improvement for Component Mode Synthesis," *AIAA Journal*, Vol. 30, No. 2, 1992, pp. 490–495.

[11] Hasselman, T. K., and Hart, G. C., "A Minimization Method for Treating Convergence in Modal Synthesis," *AIAA Journal*, Vol. 12, No. 3, 1974, pp. 316–322.

[12] Kublanovskaya, V. N., "On an Approach to the Solution of the Generalized Latent Value Problem for λ-Matrices," *SIAM Journal on Numerical Analysis*, Vol. 7, No. 4, 1970, pp. 532–537.

[13] Leung, A. Y. T., "An Algorithm for Matrix Polynomial Eigenvalue Problems," *Journal of Sound and Vibration*, Vol. 158, No. 2, 1992, pp. 363–368.

[14] Thurston, G. A., "Roots of Lambda Matrices," *Journal of Applied Mechanics*, Vol. 45, No. 4, 1978, pp. 859–863.

[15] Ruhe, A., "Algorithms for the Nonlinear Eigenvalue Problem," *SIAM Journal on Numerical Analysis*, Vol. 10, No. 4, 1973, pp. 674–689.

[16] Rajakumar, C., "Lanczos Algorithm for the Quadratic Eigenvalue Problem in Engineering Applications," *Computer Methods in Applied Mechanics and Engineering*, Vol. 105, No. 1, 1993, pp. 1–22.

[17] Moler, C. B., and Stewart, G. W., "An Algorithm for Generalized Matrix Eigenvalue Problems," *SIAM Journal on Numerical Analysis*, Vol. 10, No. 2, 1973, pp. 241–256.

[18] Fricker, A. J., "A New Approach to the Dynamic Analysis of Structures Using Fixed Frequency Dynamic Stiffness Matrices," *International Journal for Numerical Methods in Engineering*, Vol. 19, No. 8, 1983, pp. 1111–1129.

[19] Jain, N. K., Singhal, K., and Huseyin, K., "On Roots of Functional Lambda Matrices," *Computer Methods in Applied Mechanics and Engineering*, Vol. 40, No. 3, 1983, pp. 277–292.

[20] Yang, W. H., "A Method for Eigenvalues of Sparse λ Matrices," *International Journal for Numerical Methods in Engineering*, Vol. 19, No. 6, 1983, pp. 943–948.

[21] Osborne, M. R., and Michaelson, S., "The Numerical Solution of Eigenvalue Problems in Which the Eigenvalue Parameter Appears Nonlinearly, with an Application to Differential Equations," *Computer Journal*, Vol. 7, April 1964, pp. 66–71.