Component Mode Iteration for Frequency Calculations

Jeffrey K. Bennighof*
University of Texas, Austin, Texas

This paper presents component mode iteration, a new method for carrying out a form of subspace iteration on the substructure level, for structures composed of substructures. In component mode iteration, substructure displacement is in terms of "component modes" with the interface displacements as coefficients. Component modes are chosen such that exact compatibility is assured throughout the structure. The reduced structure eigenvalue problem is assembled and solved for lower mode approximations. The resulting estimates of lower mode interface motion are used to refine the component modes by approximating the dynamic response of substructures to interface motion with a vector iteration process. For this step, the substructures are independent and can be processed in parallel. Reassembly and solution of the reduced structure problem yields improved lower mode estimates. Further iterations of refinement and reassembly are performed until convergence takes place. The initial component modes are generated automatically and at negligible expense and numerical examples show that convergence with these is excellent, yielding very accurate results after only one iteration. Indeed, convergence is shown to be significantly faster than in the standard subspace iteration method.

Introduction

In the past two decades, the dynamic analysis of large structures has received considerable attention. A number of methods have been developed for analyzing the vibration of complex structures. Of these, many fall into one of two classifications. In the component mode synthesis methods, a structure is divided into substructures and each substructure is modeled in terms of a few "component modes" or substructure displacement functions. These reduced substructure models are coupled together to form a model for the total structure; the resulting eigenvalue problem is of much lower order than if reduction had not taken place at the substructure level.[1-8] In the subspace iteration method, the structure model is not reduced; however, in a procedure related to vector iteration in the power method, a set of eigenvectors and eigenvalues corresponding to the lower modes of the structure is obtained.[9-11]

In the component mode synthesis methods, the accuracy of the results depends on the quality of the substructure models and on how they are coupled together. In order to ensure that the substructure models will adequately represent the motion of the substructures as part of the total structure, substructure displacement functions of several types have been used. "Constraint modes" are static responses to displacements at the interfaces between substructures and they include the set of substructure rigid-body modes as a subset.[1] "Normal modes" are the natural modes of vibration of the substructures with interfaces either fixed, free, or loaded in some manner in order to simulate the effect of the rest of the structure.[1-3] Of the normal modes used, the loaded-interface type has been shown to give superior results,[3] but the question of what the optimal interface loadings are remains. Other component modes directly related to the solution of some structure response or eigenvalue problem include "attachment modes,"[4] "inertia relief modes,"[5,6] and "residual attachment modes."[4-6] "Admissible vectors" contain coefficients of local finite-element admissible functions and represent substructure admissible functions that usually have no direct physical significance, except that they are often chosen based on the modes of a simpler structure similar to the one being analyzed.[7,8] Admissible vectors appear to require less computational effort to generate, but for complicated structures the selection of a suitable set may present difficulties. Regardless of the method used to generate the component modes, a poor selection can result in a model that is stiffer than the actual structure, so that structure modes can be inaccurately predicted or even skipped altogether.

The object when selecting component modes is to choose the smallest set that will yield the greatest accuracy in the most structure modes. With an ideal set, only as many component modes are needed as the number of structure modes sought. Of course, such a set cannot be selected for a substructure without knowing how it will interact with the rest of the structure. Frequently, in component mode synthesis, the component modes are chosen to give as complete a set of displacement degrees of freedom as possible to the substructures, independent of the properties of the rest of the structure. While substructure independence does have some value, taking an early estimate of the interaction with the structure into account when generating component modes can improve their quality, so that fewer of them will yield better results with a smaller structure eigenvalue problem. The method of this paper has this feature, and the numerical results attest to the merit of such an approach.

Regarding the coupling of substructures, although some may have overlooked this, Meirovitch and Hale[7] have pointed out that the practice of enforcing the geometric compatibility between substructures at only a finite number of points on the substructure interfaces can cause the computed structure eigenvalues to cease to be upper bounds for the actual eigenvalues. This is because geometric compatibility may not be satisfied at some points on the interfaces, particularly if the interfaces are composed of lines or surfaces that contain an infinite number of points. For this reason, they have introduced the concept of the "intermediate structure," which is the structure whose interface compatibility only approximates that of the real structure. Because the intermediate structure model is more flexible than the actual structure, at least at interfaces, its eigenvalues can be lower than those of the actual structure, so that the accuracy of the results can again become questionable.

By comparison with the component mode synthesis methods, the subspace iteration method can be a safer ap-

Presented as Paper 86-1023 at the AIAA/ASME/ASCE/AHS 27th Structures, Structural Dynamics and Materials Conference, San Antonio, TX, May 19-21, 1986; received June 16, 1986; revision submitted Oct. 29, 1986.

*Assistant Professor, Department of Aerospace Engineering and Engineering Mechanics. Member AIAA.

proach to the analysis of large structures. In this method, the structure is treated as a whole and, in the iteration process, a set of initial trial vectors for the entire structure converges to the set of lower modes for the structure. The method has been mechanized so that no intuition is required on the part of the analyst to generate a good set of initial trial vectors. However, large problems may require a great deal of data transfer in and out of core, greatly increasing the computational time required.

In a recent paper,[8] Hale and Meirovitch have developed a substructure synthesis method that allows a form of subspace iteration method to be carried out on the substructure level. An initial set of admissible vectors is chosen for each of the substructures and the reduced structure eigenproblem is assembled and solved for estimates of the lower structure modes. Then, each set of substructure admissible vectors is improved by calculating the dynamic response to the interface forces exerted by the rest of the structure as it vibrates in the computed lower modes. This technique makes the component mode synthesis approach less sensitive to the quality of initial substructure representation, as the component modes are iteratively improved. It also reduces the problems associated with the subspace iteration method for large structures, as the size of the structure eigenproblem is greatly reduced and the substructures can be processed independently and in parallel. However, some problems still remain. In the paper by Hale and Meirovitch,[8] the concept of the "intermediate structure" is still used, as interface compatibility between substructures is not exactly satisfied. Also, the explanation of the method and the computational procedure are quite complicated and cumbersome. The initial admissible vectors are chosen to resemble the modes of a simpler structure similar to the structure being analyzed, but for more complicated structures it may be more difficult to generate a suitable set of initial admissible vectors. For this reason, this technique seems difficult to automate completely, as a considerable amount of intuition on the part of the analyst may still be required.

This paper presents a new method known as component mode iteration (CMI) that carries out a form of subspace iteration on the substructure level. This method dispenses with the concept of the "intermediate structure," as it provides a technique with which interface compatibility can be satisfied as well as is possible when modeling a given structure as a whole using the finite element method. Ordinarily, this can be done exactly, even when the interfaces are lines or surfaces, although in some circumstances it may not prove convenient. The formulation and computational procedure for CMI are much more straightforward than for the method of Hale and Meirovitch and the structure eigenvalue problem is smaller. Again, the substructures can be processed independently and in parallel. Convergence in CMI is significantly faster that in the subspace iteration method, as will be shown. Finally, CMI requires no intuition on the part of the analyst because it generates its own initial trial vectors. In fact, it appears likely that dividing the structure into substructures can eventually be automated. This will allow CMI to be implemented in a form in which an analyst can simply enter data describing the structure as a whole, without being concerned with any of the substructuring details.

Ensuring Exact Interface Compatibility

In the Rayleigh-Ritz method, of which the finite element method can be considered a subset, the computed eigenvalues are guaranteed to be upper bounds for the structure eigenvalues if the admissible functions are continuous in up to the $(p-1)$st derivative wherever the differential stiffness operator is of order $2p$ in space.[12] In the finite element method, when conforming elements are used, the element interpolation functions are selected in such a way that this continuity will be ensured between elements.

The displacement in an element is expressed in terms of interpolation functions multiplied by nodal displacement degrees of freedom. Each of the element interpolation functions has a value of one in its corresponding nodal degree of freedom and a value of zero in all of the element's other nodal degrees of freedom, so that the element nodal degrees of freedom can vary arbitrarily and independently. Further, in the case of conforming elements, the interpolation functions must satisfy the requirement that sharing the nodal degrees of freedom associated with the boundary between two neighboring elements will automatically ensure the continuity of up to the $(p-1)$st displacement derivative everywhere on the boundary between these elements. For a structure modeled in terms of conforming elements, sharing nodal degrees of freedom in this manner ensures exact displacement continuity throughout the model; it also makes assembling the structure mass and stiffness matrices a simple matter of superimposing the element mass and stiffness matrices.

In the component mode iteration method, these properties of the conforming elements used to model a structure are exploited in order to represent substructures as conforming superelements that can be used to model the same structure. For each substructure, substructure interpolation functions are generated in terms of element interpolation functions, with each substructure interpolation function satisfying the requirements above for conforming element interpolation functions. The superelements constructed to represent substructures have the property that, when nodal degrees of freedom are shared between substructures, exact interface compatibility is ensured throughout the structure model. Again, structure mass and stiffness matrices are formed by simply superimposing superelement mass and stiffness matrices, but because these are greatly reduced in size, the structure model is much smaller. In the hierarchical finite element method for one-dimensional elements, elements can be refined by adding hierarchical interpolation functions that go to zero in up to the $(p-1)$st displacement derivatives at element boundaries.[13] These functions clearly do not affect interelement compatibility, but adding displacement degrees of freedom of this type to the elements can improve their performance substantially. Hence, a similar approach will be used here to add displacement degrees of freedom to the substructure models.

It is assumed that a conforming finite-element model of a structure composed of substructures is given. Within a substructure, the finite-element representation of the displacement is given by

$$\boldsymbol{u}(P,t) = \sum_{i=1}^{n_s} \phi_i(P) a_i(t) = \Phi(P)\boldsymbol{a}(t) \tag{1}$$

where $\boldsymbol{u}(P,t)$ is a vector of displacements in the three Cartesian directions as a function of position and time, $\Phi(P)$ is a matrix of interpolation functions, $\boldsymbol{a}(t)$ is a vector of nodal degrees of freedom, and n_s is the total number of nodal degrees of freedom in the substructure. If only m_s of these degrees of freedom are shared with the rest of the structure at substructure interfaces, Eq. (1) can be partitioned as follows:

$$\boldsymbol{u}(P,t) = \sum_{i=1}^{m_s} \phi_i(P) a_i(t) + \sum_{i=m_s+1}^{n_s} \phi_i(P) a_i(t)$$

$$= [\Phi_1(P) | \Phi_2(P)] \begin{bmatrix} \boldsymbol{a}_1(t) \\ \hline \boldsymbol{a}_2(t) \end{bmatrix} \tag{2}$$

in which $\boldsymbol{a}_1(t)$ contains the nodal degrees of freedom shared with the rest of the structure and $\boldsymbol{a}_2(t)$ the nodal degrees of freedom internal to the substructure. If the structure model

were not divided into substructures, displacement continuity at the locations of the substructure interfaces would be ensured by requiring that the elements on either side of the interfaces share the nodal degrees of freedom $a_1(t)$. Therefore, in constructing reduced substructure models, $a_1(t)$ must still be shared with the rest of the structure model so that exact displacement continuity will continue to be ensured everywhere on the interfaces between substructures. Note that the interface compatibility does not depend on the entries in $a_2(t)$.

The displacement in the substructure can be expressed in terms of a reduced set of interpolation functions, each of which is a linear combination of the finite-element interpolation functions. Thus, the set of substructure interpolation functions $\Psi(P)$ is of the form

$$\Psi(P) = \Phi(P)C \tag{3}$$

in which C is a rectangular matrix of coefficients having n_s rows. If displacement continuity is to be preserved at the interfaces, there must be a set of functions $\Psi_1(P)$ equal in number to the set of functions $\Phi_1(P)$, which will share nodal degrees of freedom $a_1(t)$ with the rest of the structure as the functions in $\Phi_1(P)$ do in the full finite-element model. The functions $\Psi_1(P)$ must have the same displacement properties at the interfaces that the functions in $\Phi_1(P)$ have. Because the functions in $\Phi_2(P)$ do not affect displacement continuity at the interfaces, they can participate arbitrarily in $\Psi_1(P)$. The remaining functions $\Psi_2(P)$ in $\Psi(P)$ must not affect interface compatibility, so they must consist entirely of the funtions in $\Phi_2(P)$. Hence, if Eq. (3) is partitioned in the form

$$\Psi(P) = [\Psi_1(P)\,|\,\Psi_2(P)] = [\Phi_1(P)\,|\,\Phi_2(P)] \left[\begin{array}{c|c} C_{11} & C_{12} \\ \hline C_{21} & C_{22} \end{array}\right] \tag{4}$$

then C must have the form

$$C = \left[\begin{array}{c|c} C_{11} & C_{12} \\ \hline C_{21} & C_{22} \end{array}\right] = \left[\begin{array}{c|c} I & 0 \\ \hline C_{21} & C_{22} \end{array}\right] \tag{5}$$

where C_{21} and C_{22} are arbitrary. It can be easily verified that the functions in $\Psi_1(P)$ satisfy the requirements for conforming element interpolation functions at the substructure interfaces. Also, the functions in $\Psi_2(P)$ satisfy the requirements for hierarchical functions mentioned above. Hence, with this superelement approach, a given set of displacement degrees of freedom can be assigned to substructures while ensuring exact interface compatibility throughout the structure, so that the "intermediate structure" concept can be discarded.

In terms of the substructure interpolation functions in $\Psi(P)$, the displacement is represented in the substructure in the form

$$u(P,t) = \Psi(P)b(t) = [\Psi_1(P)\,|\,\Psi_2(P)] \left[\begin{array}{c} b_1(t) \\ \hline b_2(t) \end{array}\right] \tag{6}$$

where $b_1(t)$ is shared with the rest of the structure and must be equal to $a_1(t)$ in Eq. (2). The question of what displacement degrees of freedom are spanned by the functions in $\Psi(P)$ remains. If a set of component modes is given for which the matrix of coefficients for the finite elment functions $\Phi(P)$ is A, a matrix C of the form in Eq. (5) can be obtained from A by means of the linear transformation

$$C = \left[\begin{array}{c|c} I & 0 \\ \hline C_{21} & C_{22} \end{array}\right] = AZ = \left[\begin{array}{c|c} A_{11} & A_{12} \\ \hline A_{21} & A_{22} \end{array}\right] \left[\begin{array}{c|c} Z_{11} & Z_{12} \\ \hline Z_{21} & Z_{22} \end{array}\right] \tag{7}$$

in which Z is given by

$$Z = \left[\begin{array}{c|c} A_{11}^{-1} & -A_{11}^{-1}A_{12} \\ \hline 0 & I \end{array}\right] \tag{8}$$

Ordinarily the columns in A can be rearranged so that A_{11} is nonsingular. Hence, this technique allows substructures to be represented as superelements in which interface compatibility is satisfied exactly and nearly any desired set of component modes can be spanned with the set of displacement degrees of freedom. One m_s-dimensional matrix inverse needs to be computed, but its accuracy is not crucial to compatibility if C_{11} and C_{12} are simply set equal to I and 0, respectively. With this approach, only the accuracy with which $\Psi(P)$ spans the desired set of component modes depends on the accuracy of this inverse.

The reduced substructure mass and stiffness matrices M'_{ss} and K'_{ss} are obtained from the finite element mass and stiffness matrices M_{ss} and K_{ss} by means of the familiar transformations

$$M'_{ss} = C^T M_{ss} C, \quad K'_{ss} = C^T K_{ss} C \tag{9a,b}$$

Now M'_{ss} and K'_{ss} can be assembled into the structure mass and stiffness matrices in the usual manner for element matrices. Note that this approach adds a degree of freedom to the structural model for each of the functions in $\Psi_2(P)$, but not for the functions in $\Psi_1(P)$ because these share coefficients with the rest of the structure already.

The nature of interface compatibility when nonconforming elements are used in the structure model remains to be discussed. This question is relevant because in some cases it may be more convenient to use nonconforming elements than conforming elements. Clearly, if nonconfirming elements are used at the interfaces, this method will not guarantee exact displacement continuity at the interfaces. However, this method will preserve whatever compatibility is present in the finite element model of the structure as a whole. Hence, the eigenvalues computed by this method will continue to be upper bounds for the eigenvalues that would be computed for a given finite element model of the structure as a whole.

Eigenanalysis Procedure for the Structure

By the method just described, the substructures making up a given structure can be modeled as superelements with a given set of displacement degrees of freedom. To analyze the structure as a whole using component mode iteration, one begins by selecting a set of trial vectors for each substructure and constructing reduced superelement models possessing these displacement degrees of freedom. These superelements are assembled together to form a reduced-order structure model from which estimates for the lower structure eigenvalues and eigenvectors are obtained. The structure model contains only the interface degrees of freedom shared by various substructures and degrees of freedom corresponding to internal substructure interpolation functions $\Psi_2(P)$ in Eq. (6). Hence, the computed eigensolution for the structure model yields estimates for the interface motion in the lower structure modes. With this information, the dynamic response of each substructure to interface motion in the lower approximate structure modes can be obtained. These response vectors form a new set of substructure displacement degrees of freedom. The superelements are refined so that these new degrees of freedom replace the former ones and the structure model is assembled again and solved. Iteration between structure and substructure levels can occur until the model has converged to the lower structure modes.

In this method, since the trial vectors are automatically improved by the iteration process, the emphasis is on selecting initial trial vectors with a minimum of computational ef-

fort rather than on selecting initial trial vectors of good quality. The method automatically generates and solves the substructure eigenvalue problem that the component modes should satisfy, so one need not solve a substructure eigenvalue problem to obtain initial trial vectors. As it happens, the set of component modes known as constraint modes must be calculated for each substructure for use in the component mode refinement process, so they are available as initial trial vectors at no additional computational expense. In the numerical examples in this paper, these constraint modes were used, sometimes supplemented by trial vectors chosen in a manner similar to that used by Bathe in the subspace iteration method,[11] as these can also be obtained at very little expense.

Determining Substructure Response to Interface Motion

After the reduced structure model is assembled and solved for approximations of the lower structure modes, the estimates thus obtained of the interface motion in the lower modes are used to refine the sets of substructure displacement degrees of freedom. An efficient method for obtaining the dynamic response of substructures to interface motion as prescribed by the structure eigensolution is presented in this section.

In terms of the finite-element model of Eq. (2), the motion of each substructure is governed by the equation

$$M_{ss}\ddot{a}(t)+K_{ss}a(t)=\left[\begin{array}{c|c} M_{11} & M_{12} \\ \hline M_{21} & M_{22} \end{array}\right]\left[\begin{array}{c} \ddot{a}_1(t) \\ \hline \ddot{a}_2(t) \end{array}\right] + \left[\begin{array}{c|c} K_{11} & K_{12} \\ \hline K_{21} & K_{22} \end{array}\right]\left[\begin{array}{c} a_1(t) \\ \hline a_2(t) \end{array}\right]=\left[\begin{array}{c} f(t) \\ \hline \mathbf{0} \end{array}\right] \tag{10}$$

in which M_{ss} and K_{ss} are the substructure mass and stiffness matrices, obtained by the finite-element method, $a_1(t)$ contains interface displacements from the structure eigensolution, $a_2(t)$ contains internal displacements, and $f(t)$ contains interface forces exerted on the substructure by the rest of the structure. The lower partitioned set of equations is sufficient to determine $a_2(t)$, so $f(t)$ is not needed. For the ith structure mode, the interface displacements $a_{1i}(t)$ are a portion of the ith structure eigenvector and the interface accelerations are related to $a_{1i}(t)$ by

$$\ddot{a}_{1i}(t)=-\lambda_i a_{1i}(t) \tag{11}$$

where λ_i is the ith structure eigenvalue, because the substructure is undergoing harmonic motion. Similarly, $\ddot{a}_{2i}(t)=-\lambda_i a_{2i}(t)$, so the lower set of equations in Eq. (10) becomes

$$\{-\lambda_i[M_{21}\,|\,M_{22}]+[K_{21}\,|\,K_{22}]\}\left[\begin{array}{c} a_{1i}(t) \\ \hline a_{2i}(t) \end{array}\right]=\mathbf{0} \tag{12}$$

which can be written

$$(K_{22}-\lambda_i M_{22})a_{2i}=(\lambda_i M_{21}-K_{21})a_{1i} \tag{13}$$

as the time dependence can be neglected. In the spirit of the power method, Eq. (13) can be solved by the iterative approach

$$a_{2i}^{(k+1)}=K_{22}^{-1}\{[\lambda_i M_{21}-K_{21}]a_{1i}-\lambda_i M_{22}a_{2i}^{(k)}\} \tag{14}$$

in which $a_{2i}^{(k)}$ and $a_{2i}^{(k+1)}$ are the kth and $(k+1)$st iterations for a_{2i}. Note that K_{22} cannot be singular if the substructure is properly attached to the structure.

Component Mode Convergence

The convergence rate for the iterative procedure of Eq. (14) is worth examining. To this end, let

$$a_{2i}=Uq \tag{15}$$

where U is the square matrix whose columns are eigenvectors of the constrained-interface substructure eigenvalue problem $K_{22}U=M_{22}U\Gamma$, normalized so that $U^TM_{22}U=I$, and q is a vector of modal coordinates for the constrained-interface substructure. Γ is the diagonal matrix of eigenvalues for the constrained-interface substructure. Premultiplying Eq. (13) by U^T and using Eq. (15) yields

$$(\Gamma-\lambda_i I)q=U^T[\lambda_i M_{21}-K_{21}]a_{1i}\equiv Q \tag{16}$$

where Q is a vector of modal forces. Letting γ_j be the jth eigenvalue in Γ, the solution for q_j in Eq. (16) is simply

$$q_j=Q_j/(\gamma_j-\lambda_i) \tag{17}$$

The iterative procedure of Eq. (14) in modal coordinates becomes, from Eq. (16),

$$q^{(k+1)}=\Gamma^{-1}[Q+\lambda_i q^{(k)}] \tag{18}$$

so that

$$q_j^{(k+1)}=1/\gamma_j(Q_j+\lambda_i q_j^{(k)}) \tag{19}$$

If

$$q_j^{(k)}=Q_j/(\gamma_j-\lambda_i)+e_j \tag{20}$$

where e_j is the error in the jth mode, the $(k+1)$st iteration yields

$$\begin{aligned} q_j^{(k+1)} &= 1/\gamma_j\left[Q_j+\lambda_i\left(\frac{Q_i}{\gamma_j-\lambda_i}+e_j\right)\right] \\ &= \frac{Q_j}{\gamma_j}\left[1+\frac{\lambda_i}{\gamma_j-\lambda_i}\right]+\frac{\lambda_i}{\gamma_j}e_j \\ &= \frac{Q_j}{\gamma_j-\lambda_i}+\frac{\lambda_i}{\gamma_j}e_j \end{aligned} \tag{21}$$

Hence, the error in the jth mode is multiplied by λ_i/γ_j in each iteration. Because the constrained-interface substructure is much stiffer than the total structure, γ_j is ordinarily much larger than λ_i, so that convergence is extremely rapid. Indeed, in the numerical examples in this paper *one* iteration was sufficient.

Computational Procedure

The computational procedure for this method is described in the outline below, where it is assumed that the m lower modes are sought. The simplicity of the method is readily apparent.

1) For each substructure:

a) Assemble and store M_{ss} and K_{ss}, which can be partitioned as in Eq. (10).

b) Decompose K_{22} and calculate and store $D_1=K_{22}^{-1}M_{21}$, $D_2=K_{22}^{-1}K_{21}$, and $D_3=K_{22}^{-1}M_{22}$ for use in step 3c.

c) For starting vectors, let

$$C=\left[\begin{array}{c|c} I & 0 \\ \hline C_{21} & C_{22} \end{array}\right]=\left[\begin{array}{c|c} I & 0 \\ \hline -D_2 & V \end{array}\right]$$

where V contains vectors chosen as in the subspace iteration method (see discussion below). The left part of C contains "constraint modes." Store C_{21} and C_{22} for use in step 3b.

d) Using Eqs. (9), calculate superelement matrices M'_{ss} and K'_{ss} and assemble them into the reduced structure matrices M and K.

2) On the structure level, find the lower m eigenvalues and eigenvectors to solve $KX=MX\Lambda$, where Λ is square and diagonal and X is rectangular.

3) On the substructure level:

a) Extract the substructure degrees of freedom X_{ss} from X. Each of the m columns of X_{ss} corresponds to $\boldsymbol{b}$ in Eq. (6).

b) Obtain

$$A=\begin{bmatrix} A_1 \\ \hline A_2 \end{bmatrix}=\left[\begin{array}{c|c} I & 0 \\ \hline C_{21} & C_{22} \end{array}\right] X_{ss}$$

Each column of A corresponds to $\boldsymbol{a}$ in Eq. (1). Note that A_1 can simply be partitioned from X_{ss}.

c) As in Eq. (14), carry out the superelement refinement

$$A_2' = D_1 A_1 \Lambda - D_2 A_1 - D_3 A_2 \Lambda$$

d) Carry out the linear transformation

$$C=\left[\begin{array}{c|c} I & 0 \\ \hline C_{21} & C_{22} \end{array}\right]=\begin{bmatrix} A_1 \\ \hline A_2' \end{bmatrix} Z$$

of Eq. (7) to ensure interface compatibility, interchanging columns of A if necessary (see discussion below). Store C_{21} and C_{22}.

e) Again, use Eqs. (9) to calculate superelement matrices and assemble them into structure matrices.

Alternate between steps 2 and 3 until the desired level of accuracy is obtained.

The first vector in V of step 1c consists of the quotients m_{ii}/k_{ii}, where m_{ii} and k_{ii} are the diagonal entries in M_{22} and K_{22}. These are scaled so that superelement mass matrices are well conditioned. The other vectors are simply unit vectors with entries of one in locations where the ratio m_{ii}/k_{ii} is largest. This is analogous to the procedure used to establish starting vectors in the subspace iteration method.[11]

For the linear transformation in step 3d, a maximum-pivot-strategy matrix inversion subroutine was modified to generate a matrix Z that performs the necessary elementary operations on the columns of A so that C will be of the proper form. With the maximum-pivot feature, the columns of A are reordered, in effect, so that the square submatrix of A_1 yielding the best inverse becomes A_{11} in Eq. (7). We have not yet formally addressed the case in which fewer structure modes are sought than the number of interface degrees of freedom for a given substructure, i.e., $m<m_s$. In this case, A has m columns and C must have m_s columns, with the upper m_s rows being the identity matrix. Here the $m\times m_s$ transformation matrix Z consists of the inverse of the "best" $m\times m$ submatrix of the upper m_s rows of A, with zero columns inserted in locations corresponding to rows of A_1 not containing pivots. Hence, if the upper square portion of C is simply set equal to the identity matrix and the lower portion is A_2Z, C spans the subspace spanned by A.

The operations in step 3 of the computational procedure are performed independently for each of the substructures, so that these tasks can be carried out simultaneously for all of the substructures on machines capable of parallel processing. This feature and the rapid convergence of the response vectors mentioned in the previous section are two major reasons for using this method rather than the subspace iteration method. A third is the decrease in storage requirements at any time during the process.

Numerical Examples

For the first numerical example, we consider the positive semidefinite structure shown in Fig. 1. The structure is composed of two rectangular frames which are diagonally braced (substructures I and III), spaced apart by a three-longeron truss (substructure II). All of the joints in substructures I and III are assumed to be rigid and the members are modeled as thin-walled tubular grid elements with the properties $EI=20$, $EA=4000$, $GJ=15.385$, $m=4$, and $I_m=0.04$. Here, EI, EA, and GJ are bending, axial, and torsional stiffnesses, respectively, m is the mass per unit length, and I_m is the mass moment of inertia about each member's axis, per unit length. On the other hand, substructure II is constructed entirely with pin joints and is attached to each of the other substructures by means of three pin joints. The members of substructure II are modeled as thin-walled tubular spar

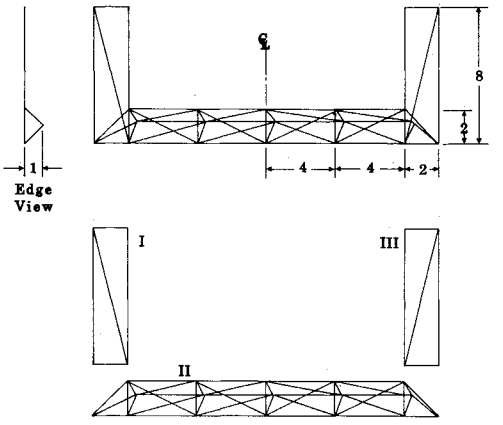

Fig. 1 Assembled and disassembled views of the structure used in the first numerical example.

Table 1 Computed nonzero eigenvalues for structure in Fig. 1

	Eigenvalues computed from reduced structure eigenproblems			
Exact eigenvalues	First eigenproblem	After one iteration	After two iterations	After three iterations
0.38962354E−02	0.45979363E−02	0.38962377E−02	0.38962354E−02	0.38962354E−02
0.63739073E−01	0.73921460E−01	0.63739341E−01	0.63739101E−01	0.63739088E−01
0.88234673E−01	0.14550932E+00	0.88259295E−01	0.88236477E−01	0.88235366E−01
0.14414122E+00	0.27638192E+00	0.14433102E+00	0.14414819E+00	0.14414441E+00
0.18342189E+00	0.43954679E+00	0.18438150E+00	0.18352517E+00	0.18346489E+00
0.20869841E+00	0.65449295E+00	0.20876625E+00	0.20870027E+00	0.20869968E+00
0.23543992E+00	0.97453193E+00	0.23749514E+00	0.23548173E+00	0.23546245E+00
0.38690624E+00	0.12684831E+01	0.40531417E+00	0.38755807E+00	0.38727872E+00
0.50425149E+00	0.16324224E+01	0.51298943E+00	0.50468874E+00	0.50446924E+00
0.65441061E+00	0.17445160E+01	0.66171124E+00	0.65557118E+00	0.65486285E+00

Table 2 Computed eigenvalues for a helicopter tail-boom

Exact eigenvalues	Eigenvalues computed from reduced structure eigenproblems		
	First eigenproblem	After one iteration	After two iterations
0.18802306E+05	0.19027203E+05	0.18802306E+05	0.18802306E+05
0.21126307E+05	0.21372486E+05	0.21126308E+05	0.21126307E+05
0.40750934E+06	0.46308348E+06	0.40751066E+06	0.40750969E+06
0.43704323E+06	0.47270108E+06	0.43704602E+06	0.43704330E+06
0.45531624E+06	0.52607062E+06	0.45531814E+06	0.45531646E+06
0.15930000E+07	0.16510430E+07	0.15930035E+07	0.15930013E+07
0.20446687E+07	0.29945353E+07	0.20458624E+07	0.20452590E+07
0.22619709E+07	0.32148129E+07	0.22633355E+07	0.22621885E+07
0.23117166E+07	0.35742933E+07	0.23202245E+07	0.23118595E+07
0.56245722E+07	0.20010764E+08	0.59675289E+07	0.56462276E+07
0.57624179E+07	0.23525317E+08	0.65486958E+07	0.58433623E+07
0.63261552E+07	0.24604060E+08	0.70571664E+07	0.63884892E+07

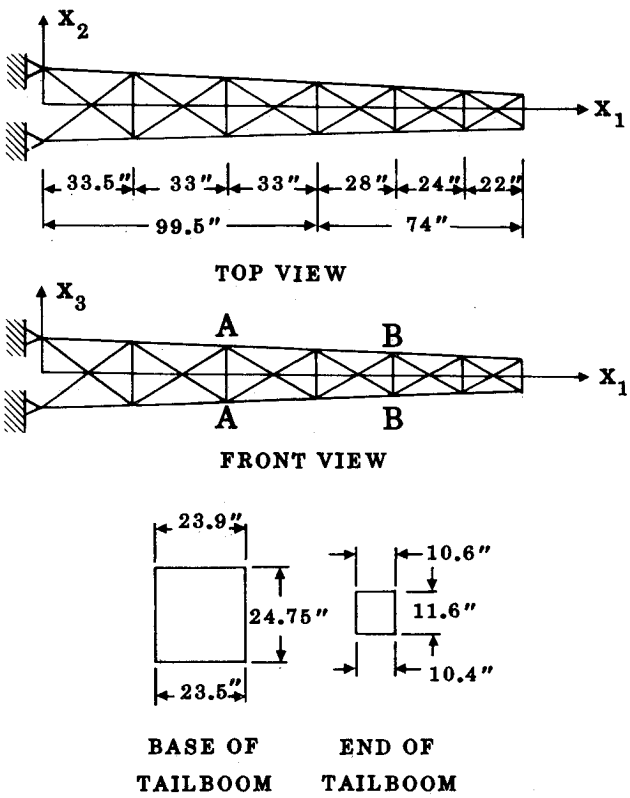

Fig. 2 Helicopter tail-boom used in the second numerical example.

elements having one-half the diameter and the wall thickness of the members comprising substructures I and III. Material properties for the spar elements in substructure II are $EA = 1000$, $m = 1$. The structure has a total of 93 degrees of freedom.

Substructures I and III each share 9 displacement degrees of freedom with substructure II, for a total of 18 interface degrees of freedom. The structure has six rigid body modes, so that if we seek estimates for the lower ten elastic modes, we must have $m = 16$ for each of the substructures. This requires 7 internal degrees of freedom for each of substructures I and III, for a total of 32 degrees of freedom in the structure eigenproblem. For comparison, the method of Ref. 8 would require a structure eigenproblem of order 48 to obtain the 10 lower flexible modes.

The first column of Table 1 contains the exact nonzero eigenvalues obtained for the structure as a whole. The following columns contain nonzero eigenvalues obtained in the solution of successive reduced structure eigenproblems. Clearly, the eigenvalues obtained before substructure trial vectors are improved from the very simple initial set are not very accurate. But the eigenvalues obtained from the second, third, and fourth structure eigenproblems demonstrate the rapid convergence of this method. Indeed, the lower seven nonzero eigenvalues from the fourth structure eigenproblem are accurate to within three places. After only *one* improvement of the substructure vectors, the lower two nonzero eigenvalues are correct to five places and the lower seven nonzero eigenvalues have less than 1% error. It is important to note that this is without any effort on the part of the analyst to select "good" initial component modes.

For a second numerical example, we consider the helicopter tail-boom structure analyzed in Ref. 14 and shown in Fig. 2. This is an open truss positive definite structure with 108 truss members, 28 joints, and 72 degrees of freedom. All members have the same material properties of $EA = 10.5 \times 10^6$ and $m = 2.588 \times 10^{-4}$.

The structure is conveniently divided into three substructures at locations A-A and B-B (see Fig. 2). There are 12 interface degrees of freedom at each division, for a total of 24 interface degrees of freedom in the structure. Without adding any internal degrees of freedom to substructures, we can find the lowest 12 structure modes with a structure eigenproblem of order 24. Here, the method of Ref. 8 would require a structure eigenproblem of order 60. Table 2 contains the lowest 12 exact eigenvalues for the structure as a whole and the estimates for the lowest 12 eigenvalues obtained in the first 3 reduced structure eigenproblems. After the first improvement of trial vectors, the first five eigenvalues are correct to five places and the next three have less than 0.4% error. After another iteration the first six eigenvalues are correct to six places. Again, these results were obtained without any particular effort to select initial trial vectors of good quality.

These results permit a direct comparison between CMI and subspace iteration, because Ref. 14 presents an algorithm for carrying out the standard subspace iteration method on structures composed of substructures. In Ref. 14, 18 trial vectors are used to obtain accurate estimates of the lowest 10 modes of the helicopter tail-boom structure analyzed here. Convergence was assumed to have taken place when successive eigenvalue estimates agreed to six decimal places. In Ref. 14, 9 or 13 iterations were required for convergence on the lowest 10 modes, depending on the starting subspace. In this paper, 12 trial vectors were used, which is the number that would ordinarily be used to obtain the lowest 6 modes accurately in the subspace iteration method.[11] Here, only *two* iterations were required for agreement with the lowest 6 exact eigenvalues to 6 places.

Conclusions

In this paper, a new method is presented for carrying out an improved form of subspace iteration on the substructure

level. It is not afflicted with problems associated with similar methods developed in the past, as the compatibility present in the finite-element model of the structure as a whole is preserved and the computational procedure for the method is much more streamlined. The convergence rate for this method is excellent, as demonstrated by the numerical examples. This is in spite of the fact that initial trial vectors were generated automatically at virtually no computational expense above that necessary for setting up the capability to improve the component modes.

Component mode iteration appears to lend itself to being totally automated. Indeed, the only part of the method that was not automated in the examples for this paper was the choice of where to divide the structures into substructures. With skyline information available for the structure as a whole, this should also be easy to automate. Hence, it appears that CMI can eventually be implemented in a form in which an analyst can simply prepare the data describing the structure as a whole wtihout concern for the substructuring details. In this form, this method offers three significant advantages over the standard subspace iteration method: the ability to process substructures in parallel, a faster convergence rate when the iteration is carried out on the substructure level, and reduced core storage requirements.

Although the performance of CMI on the example problems here was impressive, it is expected that the advantages offered by CMI will become increasingly significant as the number of degrees of freedom in the structure model increases. This is because the benefit of faster convergence in CMI will exceed the drawback of greater complexity in the algorithm by a wider margin as the computational task becomes larger. Also, the size of the reduced structure eigenvalue problem will become a smaller fraction of the size of the overall structure problem. In these examples, the reduced eigenvalue problems were about one-third the size of the overall problems, which had less than 100 degrees of freedom. In problems with thousands or tens of thousands of degrees of freedom, the structure models may have more than a hundred times as many degrees of freedom as the reduced eigenvalue problems, as these frequently will need to include only the interface degrees of freedom.

The component mode synthesis methods offer the capability of designing various components of a structure independently of one another. This method preserves this capability and even enhances it. Based on a preliminary design for each of the components, the method can be used to assemble the reduced structure model and efficiently obtain a good set of component modes and a good approximation of the interface motion in each of the structure's lower modes with only one improvement of the initial trial vectors for the components. With these data, the component modes can be improved independently for each component as design changes are made. If the changes are substantial, the estimates for interface motion can be improved by assembling the altered superelement matrices together again and resolving the reduced structure eigenproblem.

In summary, the component mode iteration method offers a number of advantages over existing component mode synthesis and subspace iteration methods in allowing for exact compatibility throughout the structure model, iterative improvement of substructure trial vectors so that an initial set can be generated automatically and economically, faster convergence when subspace iteration is carried out on the substructure level, and parallel processing of substructures.

References

[1]Craig, R. R. Jr. and Bampton, M. C. C., "Coupling of Substructures for Dynamic Analysis," *AIAA Journal*, Vol. 6, 1968, pp. 1313–1319.

[2]Benfield, W. A. and Hruda, R. F., "Vibration Analysis of Structures by Component Mode Substitution," *AIAA Journal*, Vol. 9, 1971, pp. 1255–1261.

[3]Curnier, A., "On Three Modal Synthesis Variants," *Journal of Sound and Vibration*, Vol. 90, Oct. 22, 1983, pp. 527–540.

[4]Craig, R. R. Jr. and Chang, C-J., "On the Use of Attachment Modes in Substructure Coupling for Dynamic Analysis," AIAA Paper 77-405, 1977.

[5]MacNeal, R. H., "A Hybrid Method of Compnent Mode Synthesis," *Computers and Structures*, Vol. 1, 1971, pp. 581–601.

[6]Rubin, S., "Improved Component-Mode Representation for Structural Dynamics Analysis," *AIAA Journal*, Vol. 13, 1975, pp. 995–1006.

[7]Meirovitch, L. and Hale, A. L., "A General Dynamic Synthesis for Structures with Discrete Substructures," *Journal of Sound and Vibration*, Vol. 86, Dec. 22, 1982, pp. 445–457.

[8]Hale, A. L. and Meirovitch, L., "A General Procedure for Improving Substructure Representation in Dynamic Synthesis," *Journal of Sound and Vibration*, Vol. 84, Sept. 22, 1982, pp. 269–287.

[9]Bathe, K. J., "Solution Methods of Large Generalized Eigenvalue Problems in Structural Engineering," Civil Engineering Dept., University of California, Berkeley, Rept. UC SESM 71-20, 1971.

[10]Bathe, K. J. and Ramaswamy, S., "An Accelerated Subspace Iteration Method," *Computer Methods in Applied Mechanics and Engineering*, Vol. 23, 1980, pp. 313–331.

[11]Bathe, K. J., *Finite Element Procedures in Engineering Analysis*, Prentice-Hall, Englewood Cliffs, NJ, 1982.

[12]Meirovitch, L., *Computational Methods in Structural Dynamics*, Sijthoff & Noordhoff, Groningen, the Netherlands, 1980.

[13]Bennighof, J. K. and Meirovitch, L., "Eigenvalue Convergence in the Finite Element Method," *International Journal for Numerical Methods in Engineering*, Vol. 23, No. 11, 1986, pp. 2153–2165.

[14]Arora, J. S. and Nguyen, D. T., "Eigensolution for Large Structural Systems with Substructures," *International Journal for Numerical Methods in Engineering*, Vol. 15, No. 3, 1980, pp. 333–341.