# A Computer Method for Dynamic Structural Analysis Using Stiffness Matrices

John J. Sciarra*
*The Boeing Company, Morton, Pa.*

**A computer program has been written which calculates a stiffness matrix of order 1850 for a complex semimonocoque structure, reduces it to a matrix of order 300, and finds the natural frequencies and mode shapes. In order to speed the programming and simplify the input, a unified structural approach (USA) was taken (in contrast to a substructures method). The stiffness matrix is stored on tape in lower triangular form. It is then reduced in size by eliminating unloading degrees of freedom by employing a swinging tape technique. Since this is the most time-consuming portion of the calculations, zeros are suppressed in tape operations, reducing the running time by half. No inversions are performed and the dynamic matrix $M^{-1}K$ is formed. Using similarity transformations, this matrix is converted to an upper Hessenberg form, from which the eigenvalues are obtained. The eigenvectors are obtained using the Gaussian elimination method. Numerical accuracy is ensured by the selective use of double precision arithmetic in the reduction process, in the upper Hessenberg conversion and in the eigenvector determination. The application of the program to the CH-46A Boeing-Vertol helicopter fuselage is discussed as well as running times.**

## Introduction

THE dynamic analysis of complex semimonocoque structures using stiffness matrices has become more common in recent years. One reason for this is the availability of computers with their speed and storage capacity. Another reason is the need for a more accurate vibratory analysis method for highly complicated assemblages such as fuselages and missiles.

Two methods of matrix structural analysis that have developed over the past ten years are the force (flexibility) method[1,2] and the displacement (stiffness) method.[3] The latter was chosen as the basis for this analysis because it eliminated redundancies of the structure. This simplified matrix manipulations in the computer and subsequent programming.

Helicopter fuselages in flight are subject to natural external exciting forces and moments generated by the oscillatory aerodynamic loading of each rotor blade. The forces add up at the hub in the vertical, longitudinal, and lateral directions. Their frequency is a multiple of the number of blades attached to the hub times the rotor angular velocity. It is necessary to have an accurate vibration picture of the fuselage in order to ascertain the effects of these exciting forces.

This investigation has proceeded in the past from uniform beam theory to variations of Myklestad's method.[4,5] However, for the past five years, a computer program called COSMOS,[6] based on stiffness methods, has been used with considerable success for the dynamic analysis of the CH-46A[7] and the CH-47A helicopter fuselages. The features of the stiffness method are well covered in Refs. 3 and 8. Each structural element in the assemblage has a stiffness matrix defined in

$$\{\bar{F}\} = [\bar{K}]\{\bar{\delta}\} \tag{1}$$

The bar designation over the symbols implies that the matrix is derived in terms of a convenient local coordinate system. In this latter system, the member is oriented in some special way to simplify the analysis. In Refs. 3 and 9 the analyses for some elements are given.

Since the individual stiffness components can have any orientation in space, their matrices are first derived in terms of a local coordinate system. They are then related to the general (space or datum) coordinate system by a transformation matrix defined in

$$\{F\} = [\Lambda]^T[\bar{K}][\Lambda]\{\delta\} = [K]\{\delta\} \tag{2}$$

where $\{F\}$ is a column matrix of forces or moments, $[K]$ is the merged gross stiffness matrix, $\{\delta\}$ is a column matrix of deflections or rotations, and $[\Lambda]$ is the orthogonal transformation matrix. The transpose of $[\Lambda]$ is indicated as $[\Lambda]^T$.

### *K* Matrix

In order to build the mathematical model of the composite structure, it is necessary to add algebraically the stiffness contributions to a nodal degree of freedom from all elements in the structure. This comprises one row in one gross or over-all stiffness matrix. The final stiffness matrix will be symmetric, heavily diagonalized, and highly populated with zeros.

Two methods are available to build $K$: 1) Break the main structure into parts (substructures), and then merge them at their common junction points. 2) Analyze the entire structure at one time. The method 2 is called a unified structural analysis (USA) method. It has not been used much in the past for large configurations because the matrices involved are too large for computer storage, and it seemed more natural and simple to deal with subassemblies, one at a time, rather than with the entire unit.

### Reduction

Both methods then proceed by reducing unloaded nodal degrees of freedom from the gross $K$ matrix by combining their stiffness with those remaining. This mathematical operation is similar to the algorithm for solving a set of simultaneous equations by the Gaussian elimination method; in fact, it is the solution of $m + n$ equations in terms of $m$ unknowns.

Presented at the AIAA/ASME Seventh Structures and Materials Conference, Cocoa Beach, Fla., April 18–20, 1966 (no paper number; published in bound volume of Conference papers); submitted April 29, 1966; revision received November 7, 1966.

* Senior Dynamics Engineer, Vertol Division.

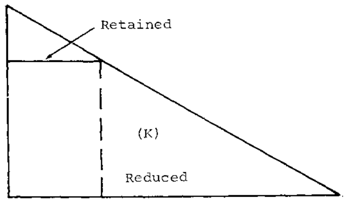

**Fig. 1 Stiffness matrix.**

### Dynamic Matrix

After reduction, the final retained matrix is of a small order. Natural frequencies are then obtained by solving

$$[M^{-1}K - \lambda I] = 0 \tag{3}$$

or

$$[K^{-1}M - (1/\lambda)I] = 0 \tag{4}$$

for the $\lambda$'s.

If Eq. (4) is used, $K^{-1}M$ must be obtained. This dynamic matrix can be formed only if $K$ is nonsingular. (However, this is generally not the case.) Therefore, it is necessary to add equilibrium equations to eliminate the rigid body modes. $\Sigma$ forces $= \Sigma$ moments $= 0$

### Eigenvalues

There are many numerical methods for obtaining eigenvalues, and they are usually unpredictable. Two general methods are available, iteration (such as the power method), and similarity transformations (upper Hessenberg transformation[10]). When the power method is used, the dominant latent root (largest eigenvalue) is obtained first. The dynamic matrix is then reduced in order by one (deflation). Then the next dominant eigenvalue is obtained.

It is thought that the lowest natural frequencies are of most interest; so the dynamic matrix must be defined as $K^{-1}M$. The eigenvalues of this matrix are $1/\lambda$, and the largest (first) latent root gives the lowest frequency. Only a few of the eigenvalues can be obtained using this method for a large matrix because the method progressively loses accuracy. Using similarity transformations, all latent roots are obtained, if desired, with a reasonable degree of ease. The dynamic matrix $M^{-1}K$ may then be used.

### Matrix Operations

The Fortran programming language is very applicable to matrix operations; however, for matrices of a large order, it consumes too much computer core storage. This implies that computer matrix routines are not applicable to a unified approach, and it is necessary to simulate matrix operations in some way to make this approach more feasible. These are described in the succeeding paragraphs.

### Simulation of Matrix Operations

The two major matrix operations in a dynamic stiffness analysis are 1) generating the gross stiffness matrix $K$, and 2) reducing $K$ to a final retained matrix. It has been necessary for computer operations in the past to partition all matrices and then to use a computer matrix interpretive program. However, this routine may be replaced by using a bookkeeping system (map), tape manipulations (string matrices and swinging tapes), or reduction by elementary arithmetic operations (double precision element by element). These are explained as follows: Partition Eq. (2);

$$\begin{Bmatrix} F_1 \\ \hdashline F_2 \end{Bmatrix} = \left[\begin{array}{c:c} K_{11} & K_{12} \\ \hdashline K_{21} & K_{22} \end{array}\right] \begin{Bmatrix} \delta_1 \\ \hdashline \delta_2 \end{Bmatrix} \tag{5}$$

If all the $\delta_2$'s are zero,

$$\{F_1\} = [K_{11}]\{\delta_1\}$$

This is deletion and is the boundary condition of fixity in translation or rotation of the degrees of freedom $\{\delta_2\}$. A method of automatic deletion follows:

1) Examine all degree-of-freedom (DOF) numbers of the node points and generate a map. If the boundary condition allows freedom of motion, then add this degree of freedom to map (reduction or retention). If deletion is desired, ignore the nodal degree of freedom in map.

2) Next, build the $K$ matrix row by row, using map. For example, if there is an entry in map when generating a row for a particular degree of freedom of a node point, add its stiffness contribution to the matrix row; otherwise, ignore it. Deletion will be accomplished automatically. If $\{F_2\} = 0$ in Eq. (5), reduction is implied. Then,

$$\{F_1\} = [K_{11} - K_{12}K_{22}^{-1}K_{21}]\{\delta_1\} \tag{6}$$

and translation or rotation of the $\delta_1$'s and $\delta_2$'s is allowed. In order to complete the preceding operation, some sophisticated computer method must be used, such as partitioning.

An alternative way of reducing a huge matrix would be to use the swinging tape method described below:

1) Generate map as before; however, place the retained degrees of freedom to the left, and the reduced to the right, as shown in Fig. 1.

2) Build the entire matrix on tape A according to map in string matrix form (Appendix A).

3) Bring in the last row of the matrix (called "fix") from tape A into the computer core.

4) Read in as much of the string matrix as possible. Reduce out fix using arithmetic operations rather than matrices. Write the modified matrix segment out on tape B in string form.

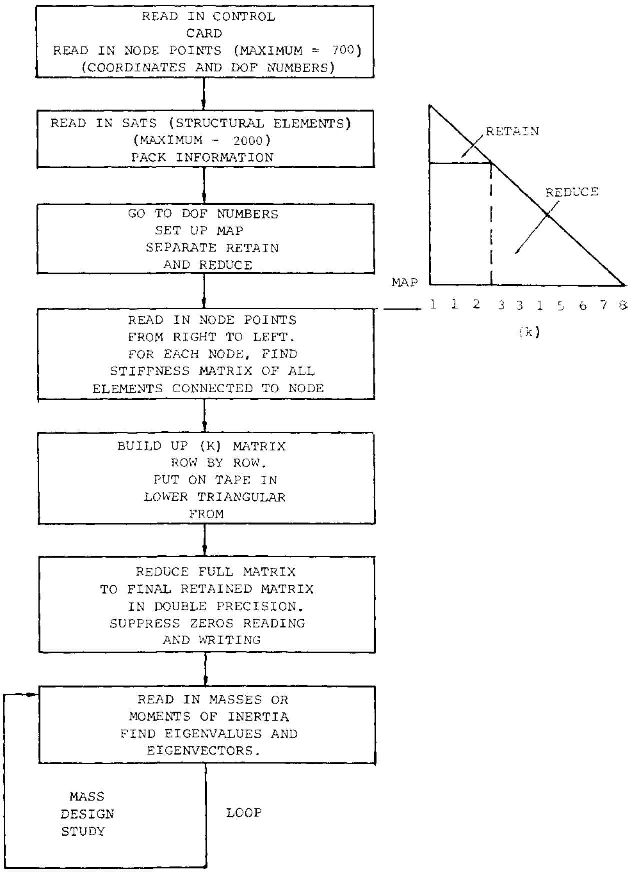
READ IN CONTROL CARD
READ IN NODE POINTS (MAXIMUM = 700)
(COORDINATES AND DOF NUMBERS)
READ IN SATS (STRUCTURAL ELEMENTS)
(MAXIMUM - 2000)
PACK INFORMATION
GO TO DOF NUMBERS
SET UP MAP
SEPARATE RETAIN AND REDUCE
READ IN NODE POINTS FROM RIGHT TO LEFT.
FOR EACH NODE, FIND STIFFNESS MATRIX OF ALL ELEMENTS CONNECTED TO NODE
BUILD UP (K) MATRIX ROW BY ROW.
PUT ON TAPE IN LOWER TRIANGULAR FROM
REDUCE FULL MATRIX TO FINAL RETAINED MATRIX IN DOUBLE PRECISION.
SUPPRESS ZEROS READING AND WRITING
READ IN MASSES OR MOMENTS OF INERTIA
FIND EIGENVALUES AND EIGENVECTORS.
MASS DESIGN STUDY
LOOP
RETAIN
REDUCE
MAP
1 1 2 3 3 1 5 6 7 8
(K)

**Fig. 2. Flow diagram of the program for the unified structural approach.**

5) Continue step 4 by reading in more of the string matrix. Reduce out fix and write on tape B. Eventually the whole matrix will have been read in, operated on and written out.

6) Delete fix. The order of the matrix is reduced by one. Rewind the tapes. Let tape B now be a new tape A, and let old tape A be new tape B.

7) Repeat step 3 by reading in the last row of the new reduced matrix. Do steps 4–6.

8) Continue this process until all the rows to be reduced out have been completed. The final retained matrix of loaded nodal degrees of freedom will then be in string form on tape.

By using map, swinging tapes, and elemental reduction, most of the matrix interpretive routine procedures can be eliminated and a unified approach becomes more feasible.

## Advantages of a Unified Approach

The USA method is defined as a technique in which the configuration is dynamically analyzed in its entirety and the eigenvalues are obtained by similarity transformations. The following advantages of a unified approach as compared to a substructures method have been found.

There is one structure compared to many and no merging of components is needed. There is no matrix interpretive routine. The program is easy to use. The structure need not be reidealized to satisfy storage requirements since the $K$ matrix can be stored on tape. The $K$ matrix is printed as a unit not as separate parts. No preliminary matrix partitioning is needed to create null off-diagonal matrices for program efficiency. Use is made of all zeros in reduction for program speed. No matrix inversion is needed. No equilibrium conditions need be imposed to eliminate rigid body modes. The higher eigenvalues may be obtained. The program of the analysis is easy to write. Figure 2 is a flow diagram of the major portions of the program.

### A Typical Example

The idealization of a complex structure requires many node points; for the CH-46A, 234 are used. Associated with each node point are six degrees of freedom. So, initially the fuselage has 1404 equations although 484 of these are deleted because some nodes have only stringers or skins connected to them and only translate (no rotation), and because most beams are parallel to an axis. Also, for this particular case, the torsional moments of inertia are assumed zero; therefore, some rotational degrees of freedom are nonexistent. This now leaves 920 equations. At 39 node points, masses and moments of inertia are concentrated to give 117 degrees of freedom. When the 920 equations are generated, they are sparsely populated with nonzeros and only these are placed on tape. This is known as zero suppression, and avoids a lot of tape mechanical movements. In Fig. 3, the percent of nonzeros is shown for the fuselage as it is being reduced to order 117.

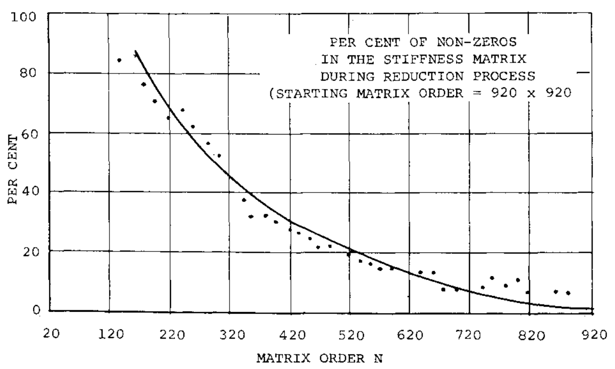

**Fig. 3 Population count.**

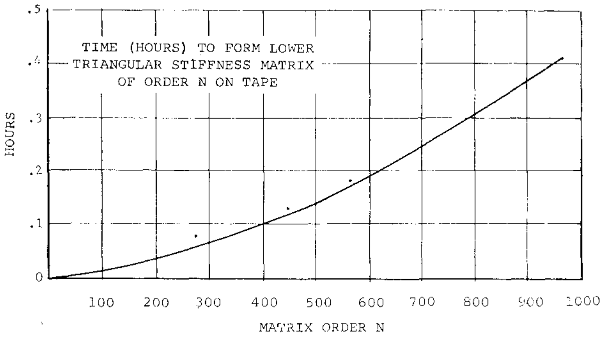

**Fig. 4 Time to generate $K$.**

For the CH-46A helicopter fuselage, 25 min is used to build $K$. The time to build stiffness matrices of any order using the present program is shown in Fig. 4. In order to avoid the accumulation of round-off errors during the generation and reduction of $K$, numbers smaller than $10^{-1}$ are constantly made zero. The fuselage was run both with and without this elimination. Although there was no change, it is felt that a "clean" matrix is preferable because more zeros means more zero suppression and more speed, and matrices are printed with less useless detail.

The reduction of the gross $K$ matrix on tape is done by using the swinging tape technique previously described. In reducing, one degree of freedom is reduced at a time from the entire matrix. If two degrees of freedom are reduced at one time, the running time would again be cut in half. The reason is that, once a portion of the matrix is in core and is being run, the internal calculating time is negligible. However, by using two rows at one time, the number of tape reads, writes, and rewinds, which take the most time, is cut in half. This double reduction procedure is done by using elementary arithmetic operations in double precision, just as in single row reduction; the extension to triple and higher reductions is obvious.

During reduction, all zeros can be tested to skip null operations, which reduces computer running time because elemental arithmetic is used. The time to reduce one row of the gross $K$ matrix in hours per row is shown in Fig. 5 for matrices of various orders. The time to reduce a matrix of order $N$ to order 100 is given in Fig. 6. These times are based on the present program.

## Eigenvalues and Eigenvectors

In the last part of the USA program, the final retained stiffness matrix is converted from its string form on tape to a full dynamic matrix in the computer core in order to find the eigenvalues and eigenvectors. From this matrix, both overall and isolated frequencies are obtained. The latter are uncoupled; the former are of more interest.

It is known from matrix algebra that the eigenvalues of the nonsymmetric dynamic matrix $[DM]$ of order $N$ are the same as those of the matrix $[L]^{-1}[DM][L]$, where $L$ is any nonsingular matrix. This is known as a similarity transformation. By the proper choice of $L$, it is possible to convert $[DM]$ to a simpler form from which the latent roots may be obtained. One of these transformations converts the dynamic

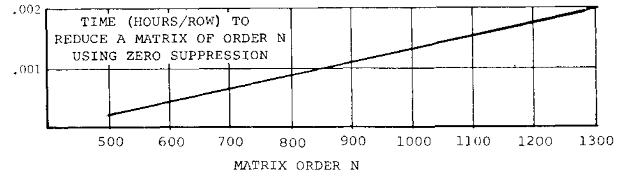

**Fig. 5 Time to reduce one row of $K$.**

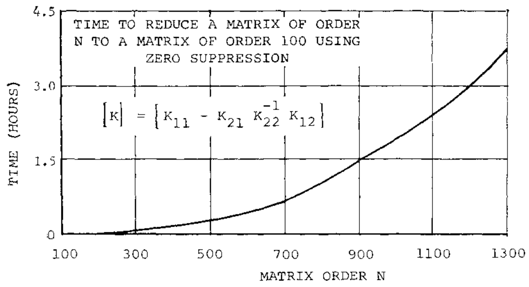

**Fig. 6 Time to reduce *K*.**

matrix to a convenient upper Hessenberg form, where the matrix is upper triangular with an additional subdiagonal. For example, consider a matrix of order 4

$$[DM] = \begin{bmatrix} a_{11} & a_{12} & a_{13} & a_{14} \\ a_{21} & a_{22} & a_{23} & a_{24} \\ a_{31} & a_{32} & a_{33} & a_{34} \\ a_{41} & a_{42} & a_{43} & a_{44} \end{bmatrix}$$

then the upper Hessenberg form would be

$$[\overline{DM}] = \begin{bmatrix} b_{11} & b_{12} & b_{13} & b_{14} \\ b_{21} & b_{22} & b_{23} & b_{24} \\ 0 & b_{32} & b_{33} & b_{34} \\ 0 & 0 & b_{43} & b_{44} \end{bmatrix}$$

The equations determining the eigenvalues and eigenvectors would then be

$$\begin{aligned} (b_{11} - \lambda)x_1 + b_{12}x_2 + b_{13}x_3 + b_{14}x_4 &= 0 \\ b_{21}x_1 + (b_{22} - \lambda)x_2 + b_{23}x_3 + b_{24}x_4 &= 0 \\ b_{32}x_2 + (b_{33} - \lambda)x_3 + b_{34}x_4 &= 0 \\ b_{43}x_3 + (b_{44} - \lambda)x_4 &= 0, \end{aligned}$$

where

$$\begin{Bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{Bmatrix}$$

is the eigenvector. Since the basic determining equation for eigenvalues is

$$[\overline{DM} - \lambda I]\{x\} = 0$$

the eigenvector is arbitrary up to a multiplicative factor. It can be normalized by dividing all the elements by $x_4$. In order to solve the preceding equations, the eigenvalue $\lambda$ is first given a trial value $\lambda_1$. The equations are then solved, starting with the last row, finding $x_3$, $x_2$, and $x_1$ in that order. Only the last three equations are used. The first equation is then used as a check; it will produce an error $E_1$ in general. A second trial value $\lambda_2$ is then taken for $\lambda$, and a second error $E_2$ is found. Since the $\lambda$'s are the roots of the characteristic equation, the error $E$ is also this algebraic polynominal in $\lambda$. A zero error $E$ indicates a solution. Since all the roots are real for a dynamic system, a reguli falsi (straight line interpolation) method would give a better estimate for the latent root by using the first two assumptions. Only real arithmetic is needed. This reguli falsi method is

$$\lambda_{\text{new}} = \lambda_1 - [(\lambda_2 - \lambda_1)/(E_2 - E_1)]E_1$$

$\lambda_{\text{new}}$ is then tried. The process continues using the two smallest errors $E$. When two successive $\lambda$'s agree within some tolerance, the iteration ceases.

The range of the eigenvalues desired for the structure could be covered, or isolated values may be obtained which are near an external exciting forcing frequency. The eigenvectors occur as a by-product of the method. The conversion of a dynamic matrix of the order $N$ to upper Hessenberg form is achieved by letting

$$[L] = \begin{bmatrix} 1 & 0 & 0 & 0 & \ldots \\ 0 & 1 & 0 & 0 & \ldots \\ 0 & l_{32} & 1 & 0 & \ldots \\ 0 & l_{42} & 0 & 1 & \\ . & . & & & \\ . & . & & & \\ . & . & & & \end{bmatrix}$$

where

$$l_{i,2} = a_{i,1}/a_{21} \qquad (i = 3,4,5\ldots,N)$$

Forming $[DM]_1 = L^{-1}[DM]L$ provides the necessary zeros in column 1. The process continues using a new $L$ matrix

$$[\bar{L}] = \begin{bmatrix} 1 & 0 & 0 & 0 & \ldots \\ 0 & 1 & 0 & 0 & \ldots \\ 0 & 0 & 1 & 0 & \ldots \\ 0 & 0 & l_{43} & 1 & \ldots \\ 0 & 0 & l_{53} & 0 & \ldots \\ . & . & & & \\ . & . & & & \\ . & . & & & \end{bmatrix}$$

where

$$l_{i,3} = \bar{a}_{i,2}/\bar{a}_{32} \qquad (i = 4,5,\ldots N).$$

The $\bar{a}_{ij}$ represents the current value of the elements of the transformed matrix. The operation continues in a similar form until the Hessenberg form is obtained. The inverse of $L$ is

$$[L]^{-1} = \begin{bmatrix} 1 & 0 & 0 & 0 & \ldots \\ 0 & 1 & 0 & 0 & \ldots \\ 0 & -l_{32} & 1 & 0 & \ldots \\ 0 & -l_{42} & 0 & 1 & \ldots \\ . & . & & & \\ . & . & & & \\ . & . & & & \end{bmatrix}$$

At times this conversion gives rise to the difference of two nearly equal numbers. Double precision arithmetic is advised. It is also necessary to select the maximum pivotal element $a_{j+1,j}$ in

$$l_{i,j+1} = a_{ij}/a_{j+1,j} \qquad (i = j + 2,\ldots,N)$$

for $j = 1, 2, \ldots, N - 2$. This is accomplished by the interchange of rows and columns. It avoids dividing by zero and also minimizes round-off errors.

A faster way to find eigenvalues is to transform the dynamic matrix to upper Hessenberg form first; then convert the resulting matrix using the *Q-R* method of J. G. F. Francis[11] to a pure upper triangular. This method is described in Appendix B. By similarity transformations, the subdiagonal of the Hessenberg form is obliterated, and the eigenvalues result on the diagonal.

Both processes, the Hessenberg and the *Q-R* transformations, are noted for their stability, particularly where all the latent roots are real. All eigenvalues may also be obtained. If the *Q-R* method is used, selective eigenvectors may be obtained, using any of the eigenvalues, by the Gaussian elimination method. Double precision arithmetic and a maximum pivotal element should be used.

If the column eigenvector is $\{x_i\}$ where the subscript $i$ is the element row number, the orthonormal column eigenvector is $\{\phi_i\}$ where

$$\phi_i = \left[ x_i \Big/ \left( \sum_{j=1}^{n} x_j{}^2 \right)^{1/2} \right.$$

The orthonormal eigenvectors satisfy the orthogonality relations,

$$\{\phi_i\}^T[M]\{\phi_i\} = [m_{\text{eff}}]$$

$$\{\bar{\phi}_i\}^T[M]\{\phi_i\} = 0$$

where $[M]$ is the diagonal mass matrix, and $[m_{\text{eff}}]$ is the diagonal effective mass. The bar indicates a different eigenvalue from that in the unbarred designation. Let $[\phi_{ij}]$ be the modal matrix whose columns are the orthonormal eigenvectors. For the CH-46A, the numerical ratio of diagonal elements to off-diagonal elements for;

$$[\phi_{ij}]^T[M][\phi_{ij}] = [m_{\text{eff}}]$$

is better than $10^6$ to 1.

The computer running time for the determination of 117 eigenvalues of a dynamic matrix of the order of 117 for the CH-46A fuselage is 2 min on the IBM 360. The calculation of each eigenvector requires 2 min, which involves the solution of 116 equations in 116 unknowns.

## Stability and Ill-Conditioning

Stability and ill-conditioning are not usually considered in stiffness-method analyses. To demonstrate one aspect of the conditioning of a stiffness matrix, consider simultaneous equations. As a measure of ill-conditioning, it is well known that if two equations of a set of $N$ simultaneous linear equations are almost equal, the $N$ equations are ill-conditioned. As a measure of this we have[12]

$$\cos\theta_{ij} = \left[\sum_{K=1}^{N} a_{iK}a_{jK} \Big/ \left(\sum_{K=1}^{N} a_{iK}^2\right)^{1/2}\left(\sum_{K=1}^{N} a_{jK}^2\right)^{1/2}\right]$$

for $i = 1,2,3,\ldots, N - 1$ and $j = i + 1, i + 2,\ldots, N$. This gives

$$m = N!/2(N - 2)!$$

conditioning numbers which are functions of the $m$ hyperangles between the $N$ hyperplanes which represent the $N$ linear equations. If the $\cos\theta_{ij} < 0.9$, the equations are well-conditioned; if the cosines are near unity, they are ill-conditioned. In order to see more clearly what is meant, consider the two equations

$$a_{11}x_1 + a_{12}x_2 = b_1 \qquad a_{21}x_1 + a_{22}x_2 = b_2$$

Assume that $a_{11}$ and $a_{22}$ are much greater than $a_{12}$ and $a_{21}$, which would exist in a stiffness matrix. Since $m$ equals 1, we get the single cosine check for conditioning

$$\cos\theta_{12} = (a_{11}a_{21} + a_{22}a_{12})/(a_{11}^2 + a_{12}^2)^{1/2}(a_{21}^2 + a_{22}^2)^{1/2}$$

The numerator is small, and the denominator is relatively large. This indicates stability since the cosine is much less than unity. This test can be used as a check on the reduction process, because the method is similar to solving simultaneous equations. The diagonality of $K$ guarantees well-conditioned equations.

## CH-46A Helicopter Fuselage

The CH-46A fuselage structure is a semimonoque construction consisting of bulkheads, frames, beams, and stringers. The structure comprises three sections, a cockpit section, a, cabin section, and an aft section. For the USA program, the structure was decomposed into 234 node points connected by 1241 structural elements. Of these structural elements, there were 504 axial members, 538 triangular skins, and 199 beams. The results were compared to test data from Ref. 7. The first few frequencies in cycles per second from the tests were 7.6, 8.1, and 9.2; the USA program gave 7.5, 8.8, and 9.3.

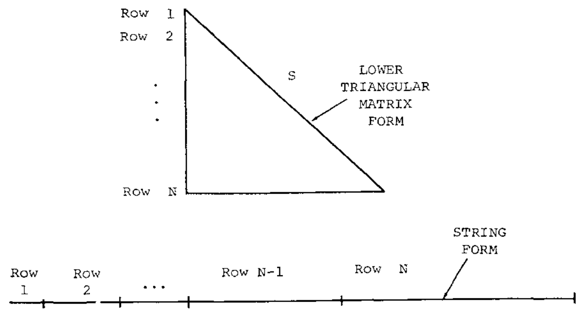

**Fig. 7 String matrices.**

## Conclusions

From the experience gained through the development of the unified structural analysis program, the following conclusions are drawn.

1) More attention should be given to this approach for the dynamic investigation of a complex structure. 2) Matrix methods can be avoided in certain computer procedures by using elemental arithmetic operations (with double precision where necessary) to make a simpler and more efficient program. 3) Similarity transformation methods are superior to iterative methods for obtaining eigenvalues, particularly where the higher latent roots are needed accurately and expediously. 4) The stability in the manipulations of large stiffness and dynamic matrices is adequate. 5) The running time of a unified structure approach is reasonable.

## Appendix A: String matrices

String matrices (see Fig. 7) are used to store symmetric matrices efficiently. If a program is written in Fortran, the matrix would have to be coded usually as a double subscripted array, and the full matrix would have to be stored; however, if it is strung out, it can be coded as a single subscripted array. Only the row position along the string is needed.

The string occupies $N(N + 1)/2$ locations. Considering computer storage available to be 22,500 location, the maximum symmetric matrix storable using Fortran double subscripts, would be of order 150. Using string matrices, the maximum becomes 212. The coding for the latter is more difficult.

## Appendix B: The *Q-R* transformation

Although the converting of a nonsymmetric dynamic matrix to the upper Hessenberg form could give the real eigenvalues by a trial-and-error procedure, another similarity transformation would accelerate this determination. The Hessenberg form is decomposed further by letting

$$[DM] = [DM]_1 = [Q_1][R_1]$$

where $[Q_1]$ is an orthogonal matrix which annihilates the subdiagonal elements and $[R_1]$ is a resulting upper triangular. Now form

$$[DM]_2 = [R_1][Q_1] \qquad \text{(the reverse product)}$$

Continue this process by finding

$$[DM]_2 = [Q_2][R_2]$$

where $[Q_2]$ is another orthogonal matrix which annihilates the subdiagonal, and $[R_2]$ is upper triangular.

So

$$[DM]_2 = [Q_2][R_2] = [R_1][Q_1]$$

but

$$[Q_1] = [DM]_1[R_1]^{-1} = [DM][R_1]^{-1}$$

so

$$[DM]_2 = [R_1][DM][R_1]^{-1}$$

This is a similarity transformation. Continuing the process, we get

$$[DM]_n = [R_{n-1}R_{n-2}\ldots R_2R_1][DM]\ [R_{n-1}\ldots R_{n-2}\ldots R_2R_1]^{-1}$$

where $n$ is the iteration number. If this is continued, convergence occurs, and an upper triangular matrix is obtained where the eigenvalues are on the diagonal. One property of this similarity transformation is that it is stable just as the reduction to the upper Hessenberg form is. Since the matrices involved are of large order, this is a prerequisite.

The orthogonal matrix $[Q]$ is a product of single element annihilator matrices similar to those used in the Jacobi method for determining eigenvalues. Its formulation is exemplified in the following example. Consider a matrix of order 3 already converted to the upper Hessenberg form

$$[DM] = \begin{bmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ 0 & a_{32} & a_{33} \end{bmatrix}$$

Form an auxiliary orthogonal matrix

$$[P_1]^T = \begin{bmatrix} C & S & 0 \\ -S & C & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

where

$$S = a_{21}/(a_{11}^2 + a_{21}^2)^{1/2} \qquad C = a_{11}/(a_{11}^2 + a_{21}^2)^{1/2}$$

Then

$$[DM]_1 = [P_1]^T[DM] = \begin{bmatrix} \bar{a}_{11} & \bar{a}_{12} & \bar{a}_{13} \\ 0 & \bar{a}_{22} & \bar{a}_{23} \\ 0 & a_{32} & a_{33} \end{bmatrix}$$

and $a_{21}$ has been annihilated. The last row remains the same. Now form

$$[P_2]^T = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \bar{C} & \bar{S} \\ 0 & -\bar{S} & \bar{C} \end{bmatrix}$$

where

$$\bar{S} = a_{32}/(\bar{a}_{22}^2 + a_{32}^2)^{1/2} \qquad \bar{C} = \bar{a}_{22}/(\bar{a}_{22}^2 + a_{32}^2)^{1/2}$$

So

$$[DM]_2 = [P_2]^T[DM]_1 = \begin{bmatrix} \bar{a}_{11} & \bar{a}_{12} & \bar{a}_{13} \\ 0 & \bar{\bar{a}}_{22} & \bar{\bar{a}}_{23} \\ 0 & 0 & \bar{\bar{a}}_{33} \end{bmatrix}$$

Therefore,

$$[P_2]^T[P_1]^T[DM] = [R_1] \qquad \text{(an upper triangular)}$$

So

$$[DM] = [DM]_1 = [P_1P_2R_1] \text{ and } [R_1Q_1] = [R_1P_1P_2]$$

where

$$[Q_1] = [P_1P_2]$$

Now $[DM]$ has been decomposed, and the reverse $[R_1Q_1]$ product has been formed.

The process of similarity transformations continues until $[DM]_n$ converges within some limit to an upper triangular form between two successive iterations. The preliminary conversion to the upper Hessenberg form eliminates the need to annihilate the sub-subdiagonal elements.

## References

[1] Argyris, J. H. and Kelsey, S., *Energy Theorems and Structural Analysis,* Butterworth, London, 1960.

[2] Denke, P. H., "A General Digital Computer Analysis of Statically Indeterminant Structures," TN D-1666, Dec. 1960, NASA.

[3] Turner, M. N. et al., "Stiffness and Deflection Analysis of Complex Structures," *Journal of the Aeronautical Sciences,* Vol. 23, No. 9, Sept. 1956.

[4] Leone, P. F., "Theory of Elastic Vibrations of Helicopter Fuselages," *Proceedings of IBM Computational Seminar,* Aug. 1951.

[5] Ricks, R. G., "A Study of Tandem Helicopter Fuselage Vibration," Tech. Doc. Rept. ASD-TDR-62-284, March 1962.

[6] Kiersky, L. B., "COSMOS, A Computer Program for Structural Analysis," Doc. D2-4513, 1962, The Boeing Co.

[7] Gabel, R., "AVID Program, Advanced Vibration Development," Rept. 107M-D-09, April 1965, The Boeing Co., Vertol Div.

[8] Archer, J. S., "Stiffness Matrix Method of Natural Mode Analysis," *Proceedings of the National Specialist Meeting on Dynamics and Aeroelasticity,* IAS, New York, Nov. 1958.

[9] Przemieniecki, J. S. and Berke, L., "Digital Computer Program for the Analysis of Aerospace Structures by the Matrix Displacement Method," Tech. Doc. Rept. FDL TDR 64-18, April 1964, Flight Dynamics Lab.

[10] Fox, L., *An Introduction to Numerical Linear Algebra,* Oxford University Press, New York, 1964, Flight Dynamics Lab.

[11] Francis, J. G. F., "The Q-R Transformation," *The Computer Journal,* Pt. I, Oct. 1961, and Pt. II, Jan. 1962.

[12] Stanton, R. G., *Numerical Methods for Science and Engineering,* Prentice-Hall, 1961.