NUREG/CR-1938
BNL-NUREG-51341
RD

# Reduction of Structural Degrees of Freedom

Manuscript Completed: December 1980
Date Published: April 1981

Prepared by
C. A. Miller*

Brookhaven National Laboratory
Upton, NY 11973

*Professor, The City University of New York, under contract to BNL

**Prepared for
Division of Reactor Safety Research
Office of Nuclear Regulatory Research
U.S. Nuclear Regulatory Commission
Washington, D.C. 20555
NRC FIN A3216**

8104160448

## Abstract

Guyan reduction is used extensively to reduce the degrees of freedom of a structural model prior to performing dynamic analysis. A modification to Guyan reduction is developed which results in the same eigenvalue problem as Guyan reduction but gives an improved transformation for expanding the reduced mode shapes. The SAPV computer program is modified to include both reduction methods.

Four sample problems are then solved using both of the reduction methods and the results are compared with an exact solution. For each of the problems the number of degrees of freedom retained for the dynamic analysis is varied. The errors in frequencies are found to be small while the errors in member forces can be quite large. The errors resulting from the modified reduction method are much smaller and more uniform throughout the structure than those resulting when Guyan reduction is used. Member force errors are much larger in systems where there are large differences in element stiffnesses (e.g., piping systems) than in systems where the stiffnesses is about the same (e.g., frames).

Table of Contents

Tables of Contents (Cont'd)

## List of Figures

List of Tables

## 1.0 INTRODUCTION

The development of discrete models for structural frameworks requires that compromises be made when deciding on the total number of degrees of freedom to be retained. A precise description of the structure may require many more degrees of freedom than are acceptable from a computational viewpoint. Limitations of the degrees of freedom to be retained arise from both restrictions embedded in the available software and economic considerations of the cost of generating the computer solutions.

Such limitations are usually more serious for dynamic problems than for static problems. Most of the large structural software packages therefore provide the user with the option of describing the structure with more degrees of freedom than are retained for dynamic response computations. Degrees of freedom to be retained in the dynamic analysis are specified by the user and a transformation relating the remaining degrees of freedom to these retained is used to reduce the order of the system (Guyan Reduction, Ref. 1, is used most often).

This dynamic reduction is often (Refs. 1,2,3) performed based upon a transformation which neglects inertial effects for the degrees of freedom to be eliminated. The transformation is then used to generate mass and stiffness matricies for the reduced system that approximate the kinetic and potential energies of the complete system. There are few data available to assess the extent to which this dynamic reduction introduces errors into various aspects of the problem.

The errors introduced in three small problems (cantilever beam, gabled frame, and small piping loop) are discussed in Ref. 4. It is shown that Guyan Reduction may result in very significant errors in the member loads

(20-40 percent for models which appear reasonable) while the modal frequencies appear quite good. A relatively simple modification of Guyan Reduction results in significantly smaller errors.

The purpose of the work reported here is to extend the studies reported in Ref. 4 to include larger structural systems and in particular structural systems found in nuclear power plants. The specific objectives are twofold. First, an assessment is made of the errors in large structural systems when both Guyan and Modified Guyan Reduction are used. Second, the economies that may be achieved by using these reduction methods as compared with obtaining complete solutions (i.e., no reduction) are investigated.

To achieve these objectives both Guyan Reduction and Modified Guyan Reduction are added to the SAPV (Ref. 5) computer program. The current version of SAPV has no reduction capability. SAPV was selected because it has the capabilities (size and element library) to analyze the structures of interest and the program is relatively easy to modify (as compared with other general purpose structural programs).

The analysis upon which the reduction methods are based is described in Section 2.0. The implementation of these methods into SAPV are discussed in Section 3.0 (Appendix A contains a listing of the modification to SAPV and Appendix B contains a users guide indicating how this version of SAPV may be accessed at Brookhaven National Laboratory). The application to sample problems is described in Section 4.0 and the work is summarized in Section 5.0.

## 2.0 ANALYSIS

The normal flow of computations in the SAPV program involves: (i) formation of the mass and stiffness matricies; (ii) calculating the required eigenvalues and eigenvectors; (iii) performing the response spectrum analysis; and (iv) evaluating modal displacements and member loads. When the reduction methods are added, this flow is interrupted after (i). At this stage: reduced mass and stiffness matricies are developed; the eigenvalues and eigenvectors for the reduced system are determined; and the eigenvectors are expanded from the reduced degrees of freedom to all degrees of freedom. The normal flow of SAPV computation at (iii) is then reentered and the solution concluded as a normal SAPV run. The analysis upon which each of these three new computations are based is outlined.

### 2.1 REDUCTION METHODS

The problems of solving for the eigenvectors and forced response of a linear structural system described in terms of its mass and stiffness matricies are considered. In particular it is of interest to consider those problems wherein some reduction in the total number of degrees of freedom is attempted before any solutions are found.

Consider the eigenvalue problem for a discrete structural system where it is desired to eliminate secondary degrees of freedom ($X_s$) and retain only primary degrees of freedom ($X_p$) thereby reducing the order of the equations to be solved. The partitioned equations of motion describing free vibration of the system may be written as:

$$\begin{bmatrix} M_{pp} & M_{ps} \\ M_{sp} & M_{ss} \end{bmatrix} \begin{Bmatrix} \ddot{X}_p \\ \ddot{X}_s \end{Bmatrix} + \begin{bmatrix} K_{pp} & K_{ps} \\ K_{sp} & K_{ss} \end{bmatrix} \begin{Bmatrix} X_p \\ X_s \end{Bmatrix} = 0 \quad (1)$$

where $M$ = mass matrix

$K$ = stiffness matrix

$\ddot{X}$ = acceleration

For each mode of the eigenvalue problem

$$\ddot{X} = \begin{Bmatrix} \ddot{X}_p \\ \ddot{X}_s \end{Bmatrix} = -\lambda^2 \begin{Bmatrix} X_p \\ X_s \end{Bmatrix} \quad (2)$$

where $\lambda$ = eigenvalue

Substituting Eq. (2) into (1) gives,

$$-\lambda^2 \begin{bmatrix} M_{pp} & M_{ps} \\ M_{sp} & M_{ss} \end{bmatrix} \begin{Bmatrix} X_p \\ X_s \end{Bmatrix} + \begin{bmatrix} K_{pp} & K_{ps} \\ K_{sp} & K_{ss} \end{bmatrix} \begin{Bmatrix} X_p \\ X_s \end{Bmatrix} = 0 \quad (3)$$

Elimination of the secondary degrees of freedom may be accomplished from the second of the above equations to give,

$$X_s = -T\ X_p$$

where, $T = (\lambda^2 M_{ss} - K_{ss})^{-1} (\lambda^2 M_{sp} - K_{sp})$ (4)

Substituting Eq. (4) into the first of Eq. (3) yields the eigenvalue problem in the reduced degrees of freedom ($X_p$),

$$\lambda^2 (M_{pp} - M_{ps}T)X_p = (K_{pp} - K_{ps}T)\ X_p \quad (5)$$

Unfortunately the transformation matrix (T) involves the eigenvalue which is not known.

Standard methods of solving the eigenvalue problem may be used to find the solution of Eq. 5 provided a new transformation matrix (T) is computed each time a new estimate of the eigenvalue ($\lambda$) is found. Thus one would begin with the ($\lambda$=0) solution, determine the eigenvalue, calculate a new transformation based upon the updated eigenvalue and repeat the process until the eigenvalue no longer changes. Several methods of carrying out this iterative process have been considered. For some problems an almost linear relationship has been found to exist between the eigenvalue assumed in calculating the transformation matrix and the eigenvalue determined as the solution of Eq. (5). This would happen if ($\lambda^2 M_{ss}$) and ($\lambda^2 M_{sp}$) are small as compared with ($K_{ss}$) and ($K_{sp}$) respectively (see Eq. 4).

This suggests an interesting possibility. If $(\lambda^2 M_{ss} - K_{ss})^{-1}$ is expanded about $(K_{ss})^{-1}$ and terms associated with ($\lambda^4$) are neglected as compared with terms associated with ($\lambda^2$).

$$T = K_{ss}^{-1} K_{sp} + \lambda^2(-K_{ss}^{-1} M_{sp} + K_{ss}^{-1} M_{ss} K_{ss}^{-1} K_{sp}) \qquad (6)$$

This definition of (T) has an advantage over (4) in that an inverse need not be recalculated at each step in the iteration on ($\lambda$).

This requirement for an iterative soltuion is eliminated if Eq. (6) is substituted into the first of (3) and again the ($\lambda^4$) terms neglected as compared to the ($\lambda^2$) terms to give,

$$\begin{aligned} \lambda^2(M_{pp} - M_{ps} K_{ss}^{-1} K_{sp} - K_{ss}^{-1} M_{sp} + K_{ps} K_{ss}^{-1} M_{ss} K_{ss}^{-1} K_{sp}) X_p = \\ (K_{pp} - K_{ps} K_{ss}^{-1} K_{sp}) X_p \end{aligned} \qquad (7)$$

Eigenvalues can be determined directly from Eq. (7) without a requirement for iteration. This is the eigenvalue problem solved using both Guyan and Modified Guyan Reduction. Since the full mass matrix in SAPV is diagonal the reduced mass matrix is,

$$M^* = M_{pp} + K_{ps} K_{ss}^{-1} M_{ss} K_{ss}^{-1} K_{sp} \tag{8}$$

The reduced stiffness matrix is,

$$K^* = K_{pp} - K_{ps} K_{ss}^{-1} K_{sp} \tag{9}$$

## 2.2 EIGENVALUE SOLUTION

The SAPV computer program uses a determinant search algorithm to solve for the eigenvalues of small problems and a iteration algorithm for large problems (Ref. 6). An attempt was made to use the subspace iteration algorithm for the reduced system but difficulties were encountered unless considerable care was taken in selecting the trial vectors. This difficulty occured because the reduced mass matrix is full while the full mass matrix (used in SAPV) is diagonal.

A standard inverse iteration with Gram-Schmidt orthogonalization procedure (Ref. 6) is used. The inverse iteration method may be described as follows. The eigenvalue problem is,

$$\lambda^2 M^* X_p = K^* X_p \tag{10}$$

A trial vector ($X_1$) is selected and the following iteration performed so that ($X_K$) goes to a mode shape as the iteration index (k) increases,

$$K^* \bar{X}_{K+1} = M^* X_K$$
$$\lambda^2_{K+1} = \bar{X}^T_{K+1} K^* \bar{X}_{K+1} / C \tag{11}$$
$$X_{K+1} = \bar{X}^T_{K+1} / C$$
$$C = \bar{X}^T_{K+1} M^* \bar{X}_{K+1}$$

This iteration converges to the mode associated with the smallest eigenvalue. Modes higher than the first are determined by requiring all the vectors ($X_k$) be orthogonal to the lower modes already evaluated, thereby suppressing those modes in the iteration. Gram-Schmidt orthogonalization is used for this purpose and is done as follows when the $i^{th}$ mode is being calculated

$$X = \bar{X} - \sum_{k=1}^{i-1} Z_k^T M^* \bar{X} \tag{12}$$

where, $\bar{x}$ = trial vector

$x$ = trail vector mode orthogonal to all of the lower modes $Z_k$

## 2.3 EXPANSION OF REDUCED EIGENVECTORS

The eigenvectors found in (2.2) are of a length equal to the number of reduced degrees of freedom. If member loads are to be calculated these eigenvectors must be expanded to include all degrees of freedom.

Guyan Reduction calculates the secondary (eliminated) degrees of degrees of freedom from,

$$X_s = -K_{ss}^{-1} K_{sp} X_p \tag{13}$$

The Modified Guyan Reduction calculates the secondary degrees of freedom from Eq. (6) or if the original mass matrix is diagonal,

$$X_s = -\left[K_{ss}^{-1} K_{sp} + \lambda^2 \left(K_{ss}^{-1} M_{ss} K_{ss}^{-1} K_{sp}\right)\right] X_p \qquad (14)$$

## 3.0 SAPV MODIFICATIONS

The changes discussed in Section 2.0 have been incorporated into the SAPV program at Brookhaven National Laboratory (BNL). The program is operational on the BNL CDC-7600 computer system. The main program in SAPV is modified and thirteen new subroutines are added. All of the SAPV subroutines are unaltered. A listing of the modified main program and the thirteen new subroutines is contained in Appendix A. Detailed instructions for using the BNL SAPV program incorporating these changes is given in Appendix B. In this section of the report the general form of the modification, input data requirements, and a description of each of the programs are given.

### 3.1 GENERAL FORM OF MODIFICATION

The SAPV Computer Program is modified so that the total static degrees of freedom may be reduced prior to performing a dynamic analysis. In the previous section of the report two reduction methods are developed:

(i) Guyan Reduction which is the standard reduction used in most structural software packages. (Eq. 7) is the eigenvalue problem and (Eq. 13) is used to expand the modes after solution of the eigenvalue problem.

(ii) Modified Guyan Reduction which uses the same (Eq. 7) for solution of the eigenvalue problem. (Eq. 14) is used to expand the modes rather than (Eq. 13) used in the Guyan Reduction.

These two reduction methods are incorporated into SAPV as indicated on Figure 1. As may be seen the modification is made by interrupting the normal flow of computations in SAPV at two locations:

(i) SAP5.88*- The type of reduction (or no reduction) is read in together with the primary degrees of freedom to be retained. These data are then used to rearrange the equation numbers (defined in the ID matrix) so that the primary degrees of freedom occur first.

(ii) SAP5.296*- The reduction is performed and the eignevalue solution obtained for the reduced system. The modes are expanded to the total degrees of freedom and stored in the appropriate locations so that the normal SAP response analysis may be performed to complete the solution.

Each of the subroutines listed in Figure 1 are discussed in Section 3.3.

## 3.2 INPUT DATA REQUIREMENTS

Only one addition is required to be made to the normal SAPV data input requirements as given in Ref. 5. After the "III Nodal Point Data" and before the "IV Element Data", the following card(s) are inserted:

---

*Refers to line numbers in the BNL SAPV Program

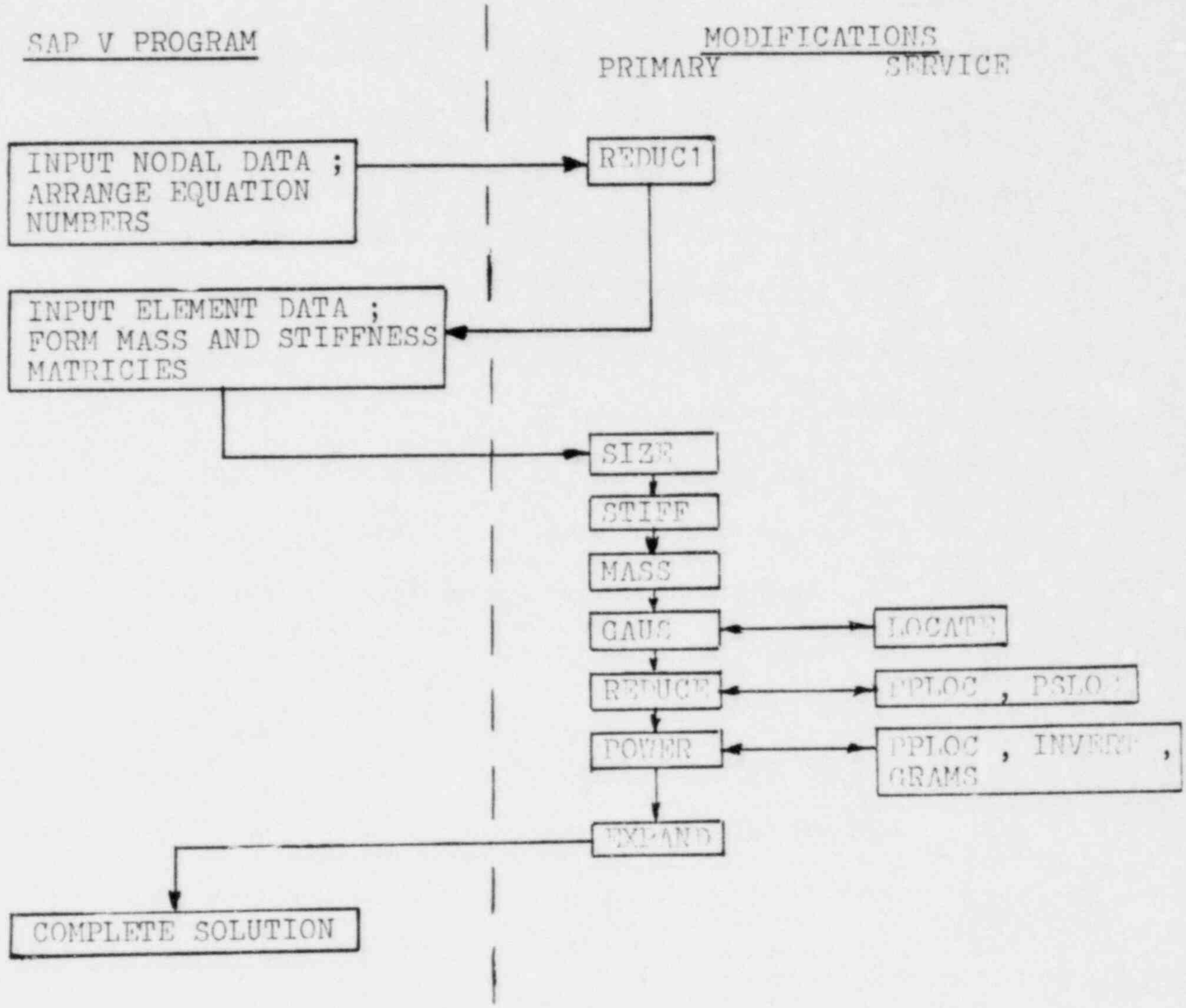


Fig. 1 GENERAL FORM OF SAP V MODIFICATION

| No of Cards | Format | Variables | Definition |
|---|---|---|---|
| 1 | I5 | NTYP | NTYP = 0; No Reduction (Standard SAP run) = 2; Guyan Reduction = 3; Modified Guyan Reduction |
| $\frac{2 \times NRDOF + 1}{14}$ | 14I5 | NRDOF, (MASN(I), MADF(I), I = 1, NRDOF) | NRDOF= Number of Primary Degrees of Freedom<br>MASN = Node Number of Primary Degree of Freedom I<br>MADF = Direction (1-6) of Primary Degree of Freedom I |

Also the Card VIII "Mode Shapes and Frequencies" is omitted if NTYP is not equal to 0.

## 3.3 DESCRIPTION OF SUBROUTINES

Before describing the subroutines common variables are defined and storage allocations are discussed.

### 3.3.1 STORAGE ALLOCATION

Most of the data required in Core is stored in vector (A) in the SAP main program and only put into matrix from within the subroutines so that the size of the matricies may be adjusted to the required sizes.

All data stored in (A) at BNL is put into Large Core Memory (Level 2). The data in (A) at the call to each of the primary subroutines is listed in Table 1. Control parameters are defined in Table 2. Note that the storage in (A) is rearranged between the subroutines REDUCE and POWER. Tape storaged is discussed where applicable in the subroutines.

3.3.2 REDUC1

The additional input data described in Section 3.2 is read into the program through this subroutine. The SAP generated matrix of equation numbers (ID) is read from Tape 8 and reordered so that the primary degrees of freedom are the first numbered equations. The primary and secondary degree of freedom equation numbers are kept in the order that they appeared in the original ID matrix. The new ID matrix is written on Tape 8. The SAP band width minimizer occurs after this location in the program. It should not be used or it will upset the equation number reordering that is performed here. (Therefore MINBND should be read in as 0 on the Master Control Card).

3.3.3 SIZE

The size parameters required for the solution are generated and printed in this routine. The following parameters, which are defined in Table 2, are calculated here: NST, NRDOF, NS, NS1, NS2, NS3, NOT, and MBS. In addition the required size of the vector (A) used in the various phases of the program is determined and printed.

Table 1

DATA STORAGE IN VECTOR A

| Subroutine | Variable | Size | Location in A |
|---|---|---|---|
| REDUC1 | ID = Matrix giving equation number | 6*NUMNP | 1 |
| | IDN = Revised ID | 6*NUMNP | 1+6*NUMNP |
| SIZE | X = One band of stiffness matrix | NEQB*MBAND | 1 |
| | XL = One band of load matrix | NEQB | 1+NEQB*MBAND |
| STIFF | $K_{pp}$ see Sect. 2.0 | NS1 | 1 |
| | $K_{ps}$ see Sect. 2.0 | NS2 | NR2=1+NS1 |
| | $K_{ss}$ see Sect. 2.0 | NSEQ*MBS | NR3=NR2+NS2 |
| | X = Dummy $K_{ss}$ | NSEQ*MBS | NR4=NR3+NSEQ*MBS |
| | XL = Dummy Load | NSEQ | NR5=NR4+NSEQ*MBS |
| MASS | XM = Diagonal Mass | NST | NR3 |
| GAUS | A1 = One block $K_{ss}$ | NSEQ*MBS | NR4=NR3+NST |
| | A2 = Second block | NSEQ*MBS | NR5=NR4+NSEQ*MBS |
| | B = Dummy load | NST | NR6=NR5+NSEQ*MBS |
| | G = Dummy load | NST | NR7=NR6+NST |
| REDUCE | $K_{pp}$ and $K^*$ | NS1 | 1 |
| | $K_{ps}$ | NS2 | NR2 |
| | XM | NST | NR3 |
| | $M^*$ | NS1 | NR4 |
| | XSS = Dummy | NS | NR5=NR4+NS1 |
| | T1 (one row) | NRDOF | NR6=NR5+NS |
| | TO (one block) | NTO*NRDOF | NR7=NR6+NRDOF |

-continued-

Table 1 cont'd

| | | | |
|---|---|---|---|
| POWER | $K^*$ | NS1 | 1 |
| | $M^*$ | NS1 | NR2=1+NS1 |
| | EV = Eigenvectors | NRDOF*MODE | NR3=NR2+NS1 |
| | XD = Factored K | NRDOF*NRDOF | NR4=NR3+NRDOF*MODE |
| | B = Dummy | NRDOF | NR5=NR4+NRDOF*NRDOF |
| | BB = Dummy | NRDOF | NR6=NR5+NRDOF |
| | EDUM = Eigenvalues | MODE | NR7=NR6+NRDOF |
| EXPAND | EV | NRDOF*MODE | NR3 |
| | TO | NTO*NRDOF | NR4=NR3+NRDOF*MODE |
| | T1 | NRDOF | NR5=NR4+NTO*NRDOF |
| | ESAP = Block of expanded modes | NTOB*MODE | NR6=NR5+NRDOF |

Table 2

DEFINITION OF CONTROL PARAMETERS

| Variable | Definition |
|---|---|
| NUMNP | Number of nodes |
| NEQB | Number of equations in one block of stiffness matrix |
| MBAND | Bandwidth of stiffness matrix |
| NS1 | Number of elements in upper half of $K_{pp}$ |
| NS2 | Number of elements in $K_{ps}$ |
| NSEQ | Number of equations in block of $K_{ss}$ |
| MBS | Bandwidth of $K_{ss}$ |
| NST | Number of unrestrained degrees of freedom |
| NS | Number of secondary degrees of freedom |
| NRDOF | Number of primary degrees of freedom |
| NTO | Number of equations in one block of $T_O$ |
| MODE | Number of modes |

### 3.3.4 STIFF

The stiffness matrix is generated in the normal SAP routines and stored banded and blocked on Tape 4. This tape is read and $(K_{pp})$ and $(K_{ps})$ are formed and kept in Large Core Memory (see Table 1). The secondary degree of freedom stiffness $(K_{ss})$ is also determined and stored banded and blocked on a new Tape 16.

### 3.3.5 MASS

The diagonal mass matrix as stored on Tape 9 is read and used to form the vector of all masses. This vector remains in Large Core Memory.

### 3.3.6 GAUS

This subroutine forms $(K_{ss}^{-1})$ using elimination as described in Ref. 6. It operates essentially the same as SESOL in the standard SAPV except that it generates "load vectors" required to compute one column of $(K_{ss}^{-1})$ at a time. The full column of $(K_{ss}^{-1})$ is stored in Tape 17. The elimination operation is set to operate on a banded symmetric matrix stored in block form. In this instance the $K_{ss}$ matrix is read in from Tape 16.

The transfer list to this subroutine contains the parameters NT1, NT2, NT3, and INDEX. if INDEX = 1 the inverse of the matrix stored on Tape NT1 is calculated and stored one column at a time on tape NT3. If INDEX = 2 the matrix stored on Tape NT1 is factored and the result is stored on Tape NT2. If INDEX = 3 the equation Ax = B is solved. (A) is stored on NT1 and B is transferred through the list. The solution (x) is stored in (B). The parameters have the following values in this application.

NT1=16
NT2=18
NT3=17
INDEX=1

### 3.3.7 REDUCE

The transformation matricies and the reduced stiffness and mass matricies are determined. The following sequence of calculations are made:

(i) $T_o = K_{ss}^{-1} * K_{sp}$ ... (first term in Eq. 6)

The transformation matrix $T_o$ is calculated one block at a time (NTO rows) and the result stored on Tape 18. $K_{ss}^{-1}$ is read from Tape 17.

(ii) $T_1 = K_{ss}^{-1} * M_{ss} * T_o$ ... (second term in Eq. 6)

The transformation is calculated one now at a time and stored on Tape 16. It is written over and destroys $K_{ss}$.

(iii) $K^* = K_{pp} - K_{ps} * T_o$

The reduced stiffness matrix (K*) is calculated and written over $K_{pp}$ which is no longer needed.

(v) $M^* = M_{pp} + K_{ps} * T$

The reduced mass matrix (M*) is calculated and the upper half stored in a new location.

### 3.3.8 POWER

The power method using inverse iteration is used to calculate the lowest (MODE) modes for the reduced mass and stiffness matricies. The iterations are continued until the change in eigenvalue ($\lambda^2$) is less than 0.01%. The error at each iteration is printed. After all eigenvalues have been generated, they are written onto Tape 7 as required for the remainder of the SAPV program.

3.3.9 EXPAND

The eigenvectors are expanded to include all unrestrained degrees of freedom according to either Eq. (13) or (14) depending on whether the reduction is Guyan or the Modified Guyan. The full vectors are then stored on Tape 7 after the eigenvalues.

3.3.10 LOCATE

This subroutine locates the (i,j) element in $K_{ss}$ when $K_{ss}$ is stored in block form.

3.3.11 PPLOC

This subroutine locates the (i,j) element in $K_{pp}$.

3.3.12 PSLOC

This subroutine locates the (i,j) element in $K_{ps}$.

3.3.13 INVERT

This subroutine factors the reduced stiffness matrix (K*) as required in the POWER subroutine. It operates the same as the GAUS routine except for the format of the matrix being reduced.

3.3.14 GRAMS

This subroutine performs Gram-Schmidt orthogonalization as described in Section 2.2. At each iteration in POWER the current eigenvector is made orthogonal to all eigenvectors previously calculated.

## 4.0 SAMPLE PROBLEMS

The modified SAPV computer program is used to study four sample problems: a ten story building; the ten story building with a relatively light structure attached to the top story; a nuclear power plant piping system; and a nuclear power plant containment building. For each of the problems solutions are obtained; with no reduction; with Guyan Reduction; and with the Modified Guyan Reduction.

### 4.1 TEN STORY BUILDING

The first problem considered is the ten story by three bay frame as shown on Figure 2. The node numbers used for the SAP data are shown by each of the nodes and the element numbers are shown underlined. All of the columns (elements number 1 through 40) have a cross-sectional area of 60 square inches and a moment of inertia of 1500 inches.$^4$ The beams (elements 41 through 70) have a cross-sectional area of 100 square inches and a moment of inertia of 1000 inches.$^4$ The material of all members has a modulus of elasticity of 30,000 ksi and a density of 490 pounds per cubic foot.

The columns are fixed at the base (nodes 1 through 4) and the response spectrum shown in Figure 3 (solid) is input in the horizontal direction. The frame is solved with no reduction and then with both Guyan and Modified Guyan Reduction. For each reduction method four solutions are obtained retaining 3, 4, 6 and 10 degrees of freedom for the dynamic analysis. In all cases the degrees of freedom retained are in the horizontal direction; with the specific degrees of freedom given as:

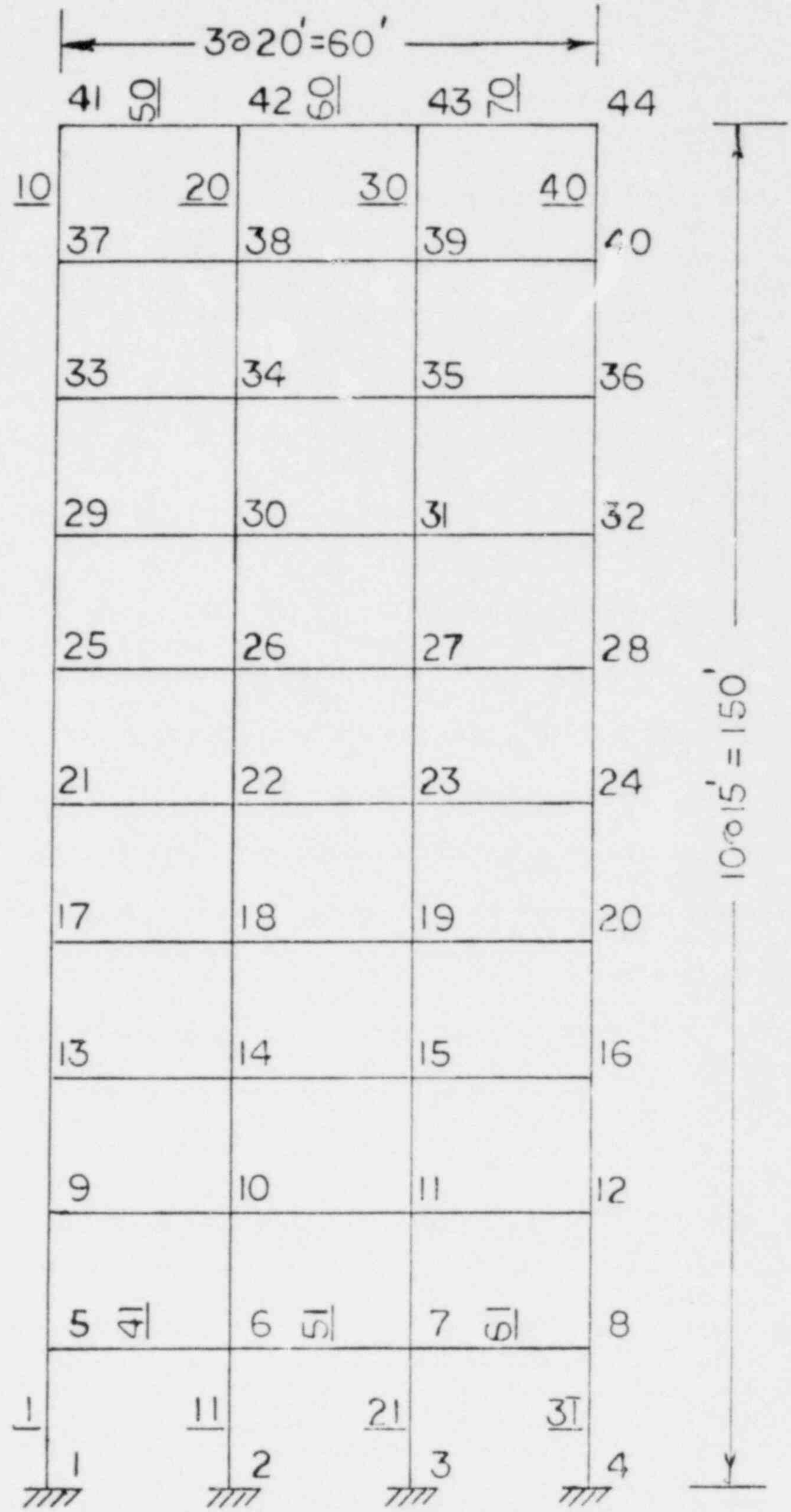


Fig. 2 TEN STORY FRAME

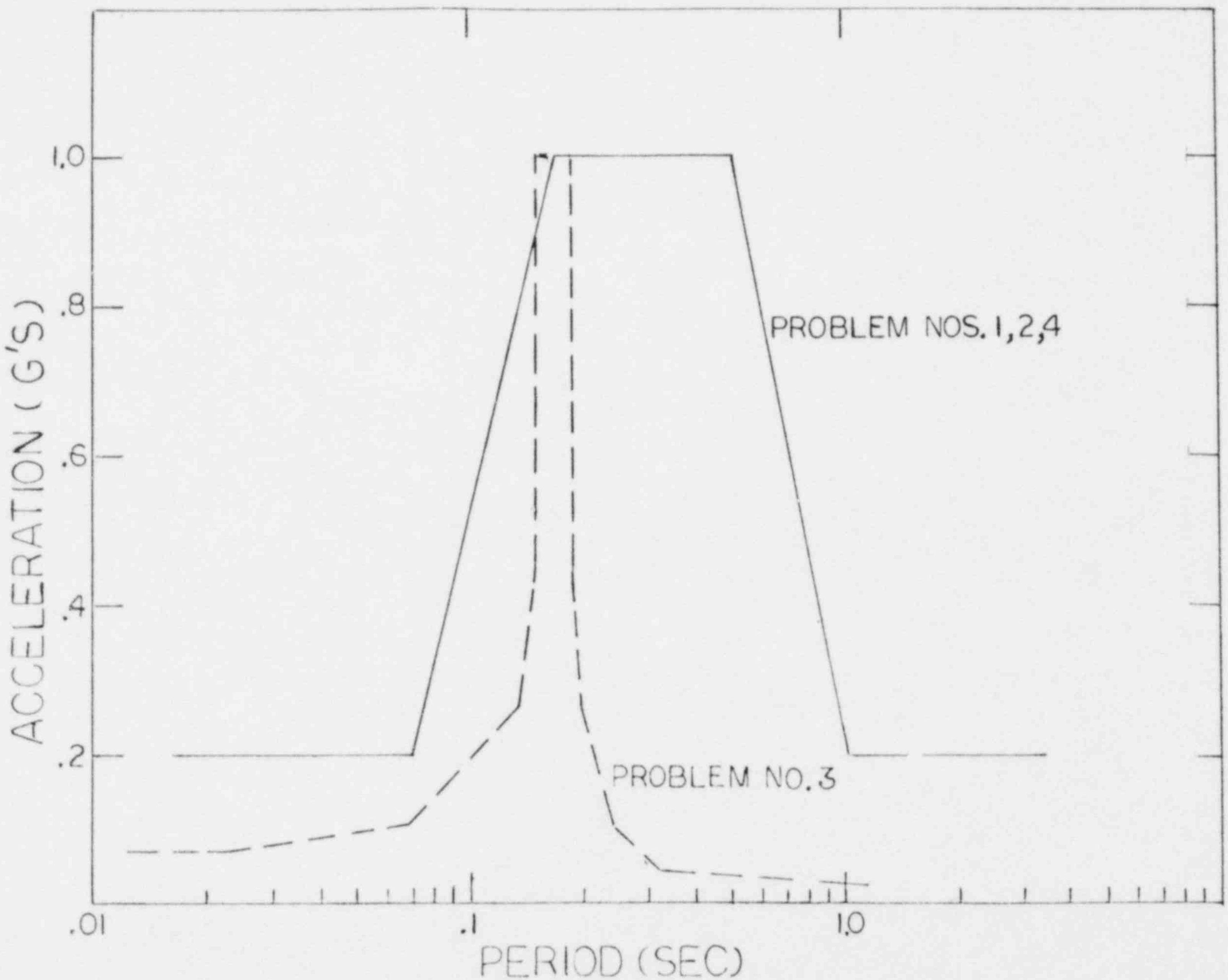

Fig. 3 BASE RESPONSE SPECTRA

| Number of Dynamic Degrees of Freedom | Degrees of Freedom - Horizontal Displacement at Nodes |
|---|---|
| 3 | 11, 26, 41 |
| 4 | 12, 23, 34, 41 |
| 6 | 10, 17, 28, 35, 38, 41 |
| 10 | 6, 9, 16, 19, 22, 25, 32, 35, 38, 41 |

Three modes are included in the solution and the results for frequencies, lower story columns (1 and 11), upper story columns (10 and 20), lower story beams (41 and 51) and upper story beams (50 and 60) are given in Table 3. As may be seen the results for frequencies are quite good regardless of the type of reduction or number of dynamic degrees of freedom retained in the model. Of course this would not hold true if any particular mode could not be reasonably represented in terms of the degrees of freedom retained. This is the reason for the departure in the mode 3 frequency as calculated with Guyan Reduction from the exact solution.

So that the errors in member bending moments may be evaluated, the percentage errors in four of the columns and four of beams are computed from the data in Table 3 and given in Table 4. The Modified Guyan Reduction gives excellent results (less than 2% error) for all reductions, even when only 3 degrees of freedom are retained. Guyan Reduction gives equally good results for the lower story members (11 and 41) but very poor results for the upper story members unless 6 or more degrees of freedom are retained. This variation in accuracy with location in the structure would make it difficult to review data for a Guyan Reduction run to determine the reliability of the results. It also appears that the Modified Guyan Reduction gives errors of one sign (i.e., a lower bound) while Guyan Reduction gives errors which can be positive or negative.

Table 3

Ten Story Frame Frequencies and Member Bending Moments

| Type of Reduction | No. of Dynamic Degrees of Freedom | FREQUENCIES (CPS) | | | COLUMN MOMENTS | | | | BEAM MOMENTS | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | MODE 1 | MODE 2 | MODE 3 | 1 | 10 | 11 | 20 | 41 | 51 | 50 | 60 |
| Exact | | 0.86 | 2.64 | 4.61 | 1520. | 150. | 1738. | 315. | 1390. | 1196. | 150.1 | 193.1 |
| G | 3 | 0.87 | 2.85 | 4.85 | 1535. | 344. | 1756. | 630.1 | 1402. | 1206. | 334. | 340.7 |
| U | 4 | 0.86 | 2.71 | 5.07 | 1521. | 198.7 | 1740. | 387.4 | 1393. | 1198. | 198.7 | 225.3 |
| Y A | 6 | 0.86 | 2.66 | 4.72 | 1514. | 151.2 | 1732. | 314.1 | 1391. | 1197. | 151.2 | 192.7 |
| N | 10 | 0.86 | 2.64 | 4.61 | 1520. | 149.9 | 1738. | 312.5 | 1390. | 1196. | 149.9 | 192.1 |
| M. G | 3 | 0.87 | 2.85 | 4.85 | 1496. | 148.7 | 1710. | 311.0 | 1367. | 1177. | 148.7 | 190.6 |
| U | 4 | 0.86 | 2.71 | 5.07 | 1513. | 149.6 | 1730. | 313.6 | 1383. | 1190. | 149.6 | 192.6 |
| Y A | 6 | 0.86 | 2.66 | 4.72 | 1519. | 149.7 | 1736. | 314.0 | 1388. | 1195. | 149.7 | 192.9 |
| N | 10 | 0.86 | 2.64 | 4.61 | 1521. | 149.8 | 1738. | 314.2 | 1390. | 1196. | 149.8 | 193 |

Table 4

%Errors in Member Moments

| No. of Deg. of Freedom | Guyan Reduction Member No. | | | | Modified Guyan Reduction Member No. | | | |
|---|---|---|---|---|---|---|---|---|
| | 11 | 41 | 20 | 60 | 11 | 41 | 20 | 60 |
| 3 | 1.0 | 0.9 | 100.0 | 76.4 | -1.6 | -1.6 | -1.3 | -1.3 |
| 4 | 0.1 | 0.2 | 23.0 | 16.7 | -0.5 | -0.5 | -0.4 | -0.3 |
| 6 | -0.3 | 0.1 | -0.3 | -0.2 | -0.1 | -0.1 | -0.3 | -0.1 |
| 10 | 0 | 0 | -0.8 | -0.5 | 0 | 0 | -0.2 | 0 |

The running time for both Guyan and Modified Guyan Reduction solution are about the same (Modified Guyan is about 0.5% slower). The solution times on the BNL CDC 7600 computer are 6.95, 7.28, 7.30 and 7.90 seconds respectively for the 3, 4, 6, and 10 degree of freedom problem. The computer implications of the reduction schemes are discussed in Section 4.5.

## 4.2 SMALL STRUCTURE ADDED TO TEN STORY BUILDING

The small structure shown in Figure 4 is added to the central bay of the top floor of the ten story building. The dashed lines represent the main structural frame as described on Figure 2 while the solid lines are the added structure. The new structure is attached to nodes 38 and 39 of the main frame. Member numbers in the new frame are underlined and node numbers shown next to each node.

All members of the added frame have a cross-sectional area of 60 square inches and a moment of inertia of 3 $inches.^4$ The relatively large cross-sectional area is used to simulate added mass to the small structure. Material properties of the small structure are the same as the main frame. In a separate computer run the small structure, is found to have frequencies of 1.78 cps and 5.86 cps when isolated from the main frame. Therefore its modes of vibration would be expected to interact with the modes of the ten story building.

The response spectrum of Figure 3 is again used as input at the base of the ten story frame and solutions obtained with no reduction and with both Guyan and Modified Guyan Reduction.

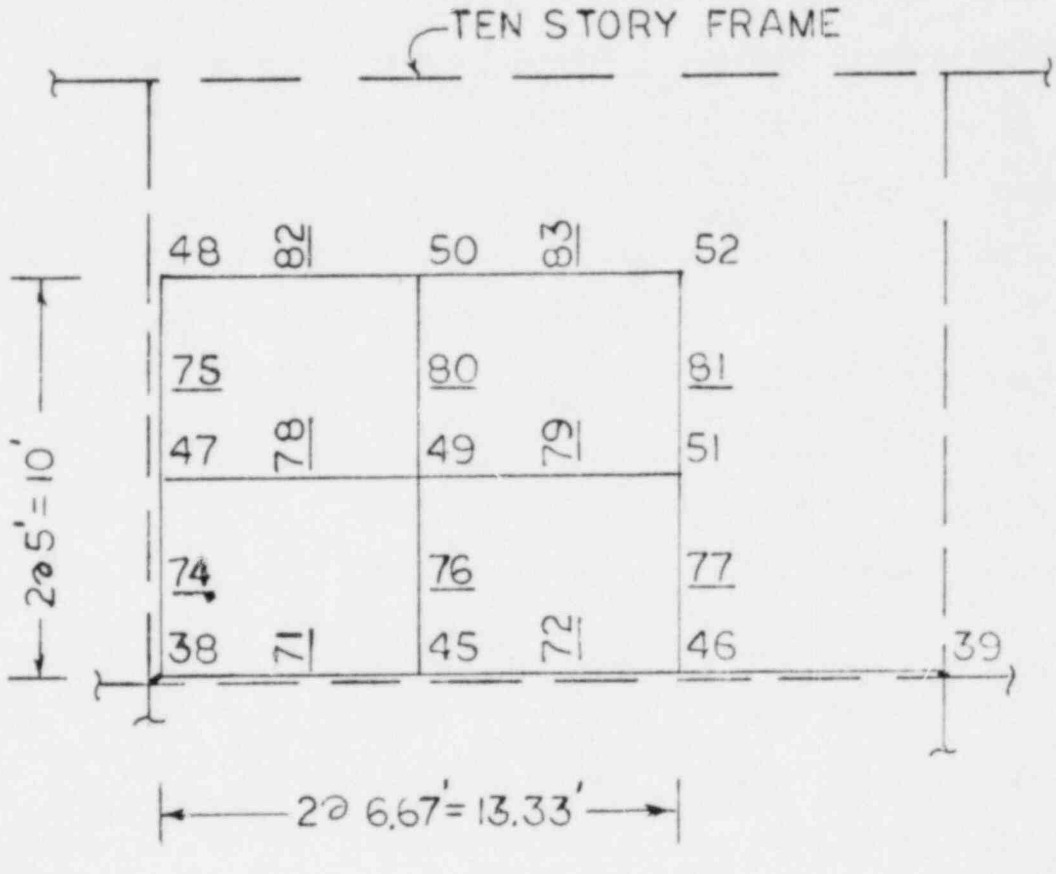


Fig. 4. SMALL STRUCTURE

Six dynamic degrees of freedom in the main structure are retained for all reduction runs since these are shown (Table 4) to give negligible errors in the main frame. Four different reductions are used for the degrees of freedom in the small structure as defined in Table 5.

Table 5

Reduction in Small Structure

| Case | Degrees of Freedom Retained* |
|---|---|
| 1 | 50 (horizontal) |
| 2 | 50 (horizontal), 49 (horizontal) |
| 3 | 48(horizontal and vertical), 52 (horizontal and vertical) |
| 4 | All horizontal and vertical |

*The horizontal degrees of freedom at the following nodes in the main structure are retained: 10, 17, 28, 35, 38, 41

The results of the computer solution are shown on Tables 6 and 7. Five modes are considered and the calculated frequencies are shown in Table 6. The results for the first four modes are good for all of the cases while there is considerable variability is the fifth mode.

The member moments are shown on Table 7 and percentage errors for each of the reduction runs is computed and tabulated in Table 8. As may be seen the Modified Guyan Reduction gives significantly better results than the Guyan Reduction. It may again be noted that the errors for Guyan Reduction vary significantly from member to member for the same reduction scheme while the errors are much closer to the same when the Modified Reduction is used.

Table 6

Frequencies in Frame with Small Structure

| Case (Table 5) | Frequencies (CPS) | | | | |
|---|---|---|---|---|---|
| | Mode 1 | Mode 2 | Mode 3 | Mode 4 | Mode 5 |
| Exact | .81 | 1.98 | 2.67 | 4.58 | 4.88 |
| 1 | .82 | 2.02 | 2.69 | 4.71 | 7.14 |
| 2 | .82 | 1.90 | 2.69 | 4.71 | 6.58 |
| 3 | .82 | 1.82 | 2.69 | 4.71 | 4.99 |
| 4 | .82 | 1.98 | 2.68 | 4.70 | 4.88 |

Table 7

Bending Moments in Small Structure

| Reduction | Case 5 (Tables) | Moments in Member No. | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 71 | 72 | 73 | 74 | 75 | 76 | 77 | 78 | 79 | 80 | 81 | 82 | 83 |
| Exact | | 23.2 | 9.1 | 20.7 | 75.0 | 21.1 | 32.2 | 28.2 | 52.7 | 74.4 | 30.7 | 11.7 | 22.8 | 11.7 |
| GUYAN | 1 | 19.2 | 6.5 | 16.9 | 60.1 | 28.1 | 25.7 | 22.1 | 51.1 | 27.8 | 42.2 | 19.9 | 28.1 | 19.9 |
| | 2 | 21.8 | 10.8 | 19.4 | 74.8 | 20.3 | 32.7 | 28.8 | 51.7 | 26.1 | 31.1 | 12.9 | 21.6 | 12.9 |
| | 3 | 20.3 | 5.3 | 18.0 | 60.1 | 28.7 | 25.2 | 21.5 | 51.8 | 26.3 | 41.8 | 18.8 | 28.7 | 18.8 |
| | 4 | 23.1 | 9.0 | 20.6 | 74.9 | 21.1 | 32.1 | 28.2 | 52.6 | 24.4 | 30.7 | 11.6 | 22.8 | 11.5 |
| M. GUYAN | 1 | 22.4 | 8.7 | 20.0 | 72.4 | 20.6 | 31.1 | 27.3 | 51.1 | 23.9 | 30.1 | 11.6 | 22.4 | 11.6 |
| | 2 | 23.1 | 9.2 | 20.6 | 75.1 | 21.1 | 32.3 | 28.3 | 52.7 | 24.6 | 30.8 | 11.8 | 22.8 | 11.8 |
| | 3 | 22.5 | 8.8 | 20.1 | 72.4 | 20.7 | 31.1 | 27.3 | 51.2 | 23.8 | 30.1 | 11.5 | 22.4 | 11.5 |
| | 4 | 23.2 | 9.0 | 20.7 | 75.1 | 21.1 | 32.2 | 28.3 | 52.8 | 24.5 | 30.8 | 11.7 | 22.9 | 11.7 |

Table 8

%Errors in Small Structure Moments

| Case (Tables) | Guyan Reduction Member No. | | | | | Modified Guyan Reduction Member No. | | | | |
|---|---|---|---|---|---|---|---|---|---|---|
| | 72 | 76 | 79 | 80 | 83 | 72 | 76 | 79 | 80 | 83 |
| 1 | -28.6 | -20.2 | +13.9 | +37.5 | +70.1 | -4.4 | -3.4 | -2.0 | -2.0 | -0.9 |
| 2 | +18.7 | +1.6 | +7.0 | +1.3 | +10.3 | +1.1 | +0.3 | +0.8 | +0.3 | +0.9 |
| 3 | -41.8 | -21.7 | +7.8 | -14.3 | +60.7 | -3.3 | -3.4 | -2.5 | -2.0 | -1.7 |
| 4 | -1.1 | -0.3 | 0 | 0 | -0.9 | -1.1 | 0 | +0.4 | +0.3 | 0 |

## 4.3 PIPING SYSTEM

The piping system shown in Figure 5 is considered. The details of this system are taken from Reference 7 (Piping Benchmark Problem No. 5). Elements 1 through 19 are 14 inch diameter pipes with a wall thickness of 0.438 inches. Element 20 through 32 are 12 inch diameter pipes with a wall thickness of 1.312 inches. Weights of 3864 pounds are placed at nodes 19, 22, and 27 to simulate valve weights and to test the effect of lumped masses. The pipe supports are modeled with boundary elements having stiffnesses of $10^7$, $10^7$, $10^7$, 450, 800 and 600 pounds per inch for elements between nodes 7-37, 18-38, 18-39, 13-34, 25-35, and 31-36 respectively.

The piping system is subjected to the response spectrum shown on Figure 3 (dashed). The response spectrum is input in the two horizontal directions (X,Z) and 0.67 of the spectrum is input in the vertical direction (Y). An exact solution is obtained and solutions are found using 7 sets of reduced degrees of freedom with both Guyan and the Modified Guyan Reductions. The primary degrees of freedom for each of the reduction schemes is shown in Table 9. Note that there are 186 unrestrained degrees of freedom but only the 93 translational degrees of freedom have mass attached.

Five modes are included in the calculation with the frequencies shown in Table 10. The same frequencies are found with Guyan and Modified Guyan Reduction.

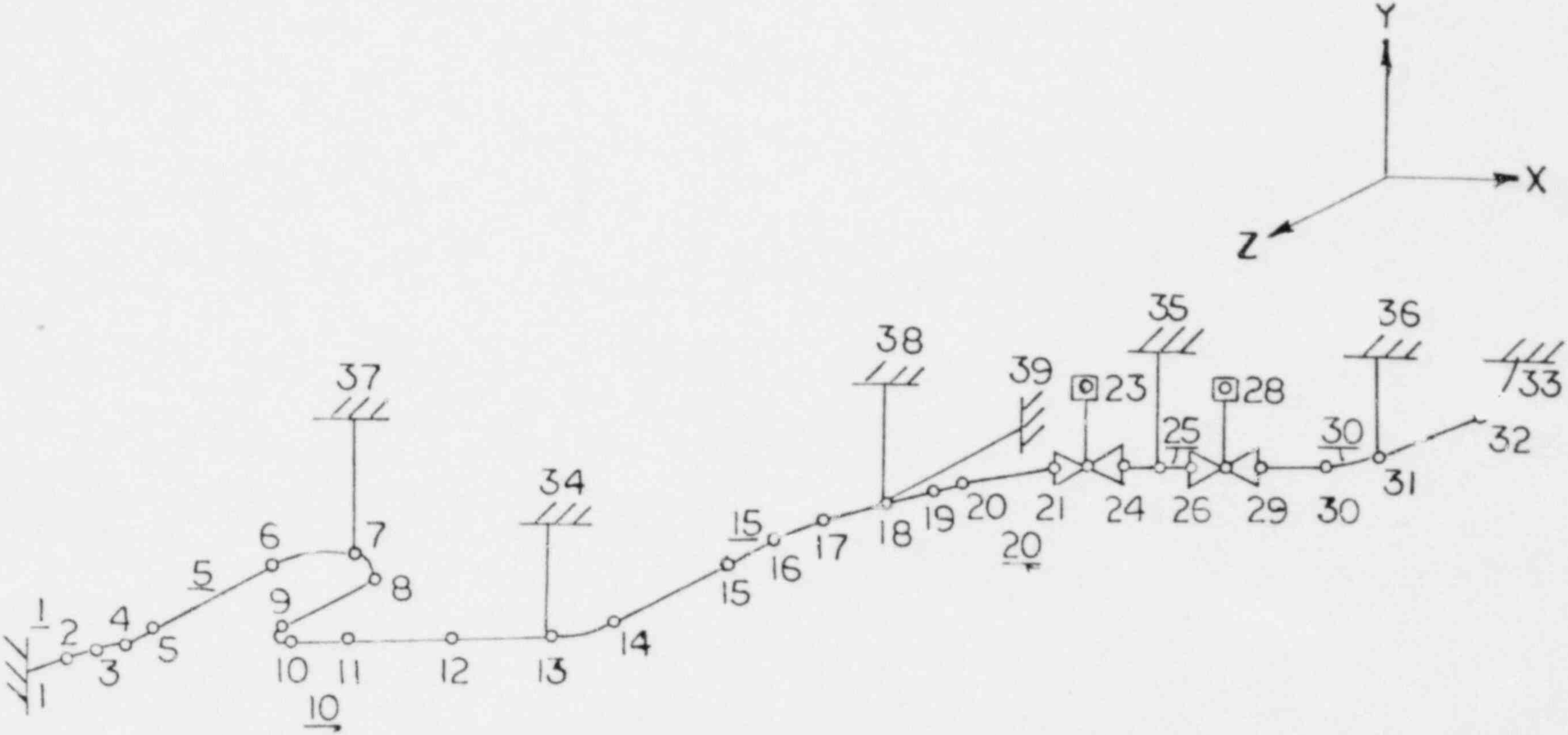


Fig. 5 PIPING SYSTEM

Table 9

Reduction Schemes for Piping System

| Case | No. of Primary Degrees of Freedom | Degrees of Freedom Retained |
| --- | --- | --- |
| A | 3 | 9-X;9-Y;9-Z |
| B | 3 | 9-X;15-Y;22-Z |
| C | 5 | 9-X;15-Y;18-X;22-X;27-Z |
| D | 10 | 5-Y;9-X;9-Z;13-Z;15-Y;18-X;22-X;22-Y;27-Z;31-Z |
| E | 15 | 5-X;5-Y;7-Z;9-X;9-Z;13-Z;15-X;15-Y;18-X;22-X;22-Y;27-X;27-Z;31-X;31-Z |
| F | 21 | 5-X;5-Y;7-X;7-Z;9-X;9-Y;9-Z;13-X;13-Z;15-X;15-Y;15-Z;18-X;22-X;22-Y;22-Z;27-X;27-Y;27-Z;31-X;31-Z |
| G | 26 | 5-X;5-Y;5-Z;7-X;7-Z;9-X;9-Y:9-Z;11-X;11-Y;11-Z;13-X;13-Z;15-X;15-Y;15-Z;18-X;22-X;22-Y;22-Z;27-X;27-Y;27;Z;31-X;31-Z;31-Y |

Table 10

Piping System Frequencies

| Reduction Type (Table 9) | Mode 1 | Mode 2 | Frequencies (CPS) Mode 3 | Mode 4 | Mode 5 |
|---|---|---|---|---|---|
| EXACT | 5.71 | 6.14 | 12.85 | 13.32 | 14.90 |
| A | 6.21 | 7.11 | 14.70 | - | - |
| B | 6.30 | 6.32 | 16.40 | - | - |
| C | 6.28 | 6.32 | 14.30 | 20.2 | 42.1 |
| D | 6.18 | 6.22 | 13.90 | 14.20 | 15.70 |
| E | 6.15 | 6.20 | 13.70 | 14.00 | 15.60 |
| F | 5.86 | 6.14 | 13.00 | 13.30 | 15.00 |
| G | 5.75 | 6.14 | 12.90 | 13.30 | 14.90 |

The frequency computations for the piping system may be seen to behave quite differently than for the frame structures. Ten percent errors occur in the first mode frequency until 21 dynamic degrees of freedom are retained.

The forces in all of the piping members for the cases run are listed in Table 11. The P column is the maximum thrust force in the pipe element while the M column is the maximum bending moment in the pipe. There is a great deal of variability in the results but it may be noted that large errors result when Guyan Reduction is used while Modified Guyan Reduction significantly reduces the errors. To illustrate some of these effects percentage errors for "typical" elements are shown in Table 12.

Table 11

PIPING FORCES

| Model | | MEMBER NUMBER 1 P | 1 M | 2 P | 2 M | 3 P | 3 M | 4 P | 4 M | 5 P | 5 M | 6 P | 6 M |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Exact | | 1558 | 171500 | 1560 | 143600 | 1847 | 137800 | 1847 | 134300 | 1016 | 73300 | 965 | 52350 |
| GUYAN | MODEL A | 1277 | 193200 | 1278 | 156600 | 1676 | 148200 | 1676 | 143500 | 1577 | 75400 | 1577 | 496690 |
| | MODEL B | 1505 | 127800 | 1506 | 110800 | 1667 | 107400 | 1667 | 105900 | 520 | 59990 | 520 | 58020 |
| | MODEL C | 1494 | 126900 | 1494 | 110000 | 1655 | 106700 | 1655 | 105100 | 516 | 59550 | 516 | 57600 |
| | MODEL D | 1069 | 158400 | 1070 | 127900 | 1381 | 117000 | 1381 | 117000 | 1346 | 60940 | 1346 | 42610 |
| | MODEL E | 1966 | 158300 | 1967 | 129900 | 2251 | 124000 | 2251 | 192560 | 1005 | 53260 | 1005 | 39060 |
| | MODEL F | 1775 | 171300 | 1776 | 141400 | 2046 | 138800 | 2046 | 136000 | 960 | 73040 | 960 | 53170 |
| | MODEL G | 1612 | 174300 | 1614 | 145900 | 1905 | 139700 | 1905 | 136500 | 857 | 74070 | 857 | 53690 |
| MOD. GUYAN | MODEL A | 1551 | 171800 | 1522 | 143700 | 1843 | 137600 | 1843 | 134400 | 1020 | 73300 | 970 | 51940 |
| | MODEL B | 1487 | 158000 | 1488 | 132900 | 1747 | 124700 | 1746 | 124700 | 894 | 68380 | 853 | 50580 |
| | MODEL C | 1455 | 154900 | 1456 | 130300 | 1710 | 124900 | 1710 | 122200 | 877 | 67050 | 837 | 49470 |
| | MODEL D | 1296 | 140700 | 1297 | 117700 | 1531 | 112700 | 1531 | 110100 | 836 | 60910 | 793 | 60910 |
| | MODEL E | 1296 | 141500 | 1297 | 118500 | 1534 | 113500 | 1532 | 110800 | 839 | 60560 | 795 | 60560 |
| | MODEL F | 1574 | 173500 | 1575 | 145200 | 1867 | 139100 | 1866 | 135800 | 1031 | 74180 | 980 | 52820 |
| | MODEL G | 1577 | 174100 | 1579 | 145700 | 1871 | 139500 | 1871 | 136300 | 1034 | 74410 | 983 | 52930 |

Table 11 (cont'd)

PIPING FORCES

| Model | | MEMBER NUMBER | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 7 | | 8 | | 9 | | 10 | | 11 | | 12 | |
| | | P | M | P | M | P | M | P | M | P | M | P | M |
| Exact | | 1302 | 80730 | 729 | 66880 | 585 | 17660 | 501 | 29440 | 192 | 45870 | 166 | 52230 |
| GUYAN | MODEL A | 1676 | 96080 | 1577 | 80910 | 324 | 11760 | 1071 | 31220 | 1071 | 50290 | 1071 | 62820 |
| | MODEL B | 1667 | 75960 | 520 | 51830 | 520 | 38700 | 1005 | 12580 | 1005 | 43520 | 1005 | 63740 |
| | MODEL C | 1655 | 75420 | 516 | 51480 | 516 | 37890 | 962 | 11850 | 962 | 42580 | 962 | 62680 |
| | MODEL D | 1381 | 79850 | 1346 | 67720 | 340 | 16480 | 527 | 23030 | 527 | 40660 | 527 | 53190 |
| | MODEL E | 1327 | 77930 | 1383 | 66810 | 309 | 14460 | 471 | 22760 | 471 | 39090 | 471 | 55290 |
| | MODEL F | 1095 | 77740 | 681 | 69340 | 364 | 18750 | 449 | 25060 | 449 | 42420 | 449 | 55440 |
| | MODEL G | 1076 | 81030 | 721 | 69780 | 536 | 18480 | 671 | 32590 | 101 | 43990 | 101 | 53640 |
| MOD. GUYAN | MODEL A | 1294 | 80800 | 733 | 67040 | 586 | 14530 | 475 | 28280 | 155 | 46220 | 176 | 52440 |
| | MODEL B | 1277 | 76120 | 681 | 62000 | 576 | 11710 | 498 | 22310 | 199 | 43850 | 188 | 52470 |
| | MODEL C | 1252 | 74520 | 669 | 60680 | 566 | 13340 | 488 | 23990 | 197 | 43110 | 183 | 51570 |
| | MODEL D | 1090 | 67280 | 589 | 56610 | 481 | 15740 | 439 | 33520 | 177 | 49780 | 143 | 49780 |
| | MODEL E | 1097 | 67730 | 588 | 56280 | 483 | 15210 | 436 | 33330 | 171 | 49520 | 143 | 49520 |
| | MODEL F | 1316 | 81500 | 737 | 67570 | 591 | 19160 | 510 | 30480 | 202 | 46450 | 176 | 52890 |
| | MODEL G | 1319 | 81730 | 741 | 67760 | 593 | 19040 | 510 | 30590 | 201 | 46650 | 174 | 53070 |

Table 11 (cont'd)

PIPING FORCES

| Model | | MEMBER NUMBER | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 13 | | 14 | | 15 | | 16 | | 17 | | 18 | |
| | | P | M | P | M | P | M | P | M | P | M | P | M |
| Exact | | 395 | 52230 | 161 | 45820 | 182 | 19800 | 204 | 19690 | 516 | 66870 | 1049 | 70290 |
| GUYAN | MODEL A | 1071 | 61020 | 324 | 47110 | 324 | 19090 | 324 | 26360 | 973 | 71620 | 1383 | 72970 |
| | MODEL B | 1005 | 65290 | 519 | 53680 | 519 | 26260 | 519 | 18010 | 1144 | 67610 | 1110 | 67680 |
| | MODEL C | 962 | 64470 | 516 | 53530 | 516 | 30850 | 516 | 16510 | 1137 | 81590 | 2608 | 81670 |
| | MODEL D | 527 | 55730 | 426 | 38710 | 426 | 38710 | 426 | 19110 | 397 | 116100 | 1586 | 116200 |
| | MODEL E | 471 | 55290 | 528 | 38630 | 528 | 38630 | 528 | 18590 | 978 | 115200 | 1477 | 115300 |
| | MODEL F | 597 | 55440 | 309 | 44780 | 578 | 21970 | 578 | 10270 | 819 | 67530 | 1295 | 71470 |
| | MODEL G | 584 | 53640 | 268 | 45190 | 392 | 20640 | 392 | 15080 | 532 | 67680 | 1288 | 71660 |
| MOD. GUYAN | MODEL A | 416 | 52440 | 77 | 45180 | 108 | 14830 | 134 | 4638 | 504 | 67370 | 860 | 70580 |
| | MODEL B | 408 | 52470 | 56 | 46890 | 86 | 22060 | 120 | 22650 | 542 | 65390 | 965 | 68900 |
| | MODEL C | 396 | 51570 | 65 | 46170 | 101 | 25520 | 135 | 26250 | 538 | 64760 | 1037 | 68050 |
| | MODEL D | 334 | 47450 | 140 | 37400 | 148 | 34140 | 160 | 34090 | 415 | 71900 | 862 | 74540 |
| | MODEL E | 338 | 47270 | 43 | 37430 | 154 | 34170 | 168 | 34070 | 419 | 71890 | 869 | 74240 |
| | MODEL F | 404 | 52890 | 170 | 46540 | 192 | 19100 | 213 | 19070 | 510 | 67940 | 1077 | 71300 |
| | MODEL G | 403 | 53070 | 172 | 46660 | 192 | 20440 | 213 | 20430 | 518 | 68210 | 1079 | 71590 |

Table 11 (Cont'd)

PIPING FORCES

| Model | | MEMBER NUMBER 19 P | 19 M | 20 P | 20 M | 21 P | 21 M | 22 P | 22 M | 23 P | 23 M | 24 P | 24 M |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Exact | | 1317 | 78150 | 1321 | 81610 | 1254 | 81390 | 72 | 4047 | 1430 | 78040 | 1443 | 64550 |
| GUYAN | MODEL A | 1331 | 75220 | 1381 | 75520 | 1071 | 70980 | 0 | 0 | 1071 | 53590 | 1071 | 36210 |
| | MODEL B | 1106 | 68140 | 1108 | 72920 | 1005 | 102400 | 0 | 0 | 1005 | 102400 | 1005 | 77020 |
| | MODEL C | 2607 | 78100 | 2603 | 76970 | 3150 | 78110 | 0 | 0 | 2584 | 67600 | 2584 | 61190 |
| | MODEL D | 1588 | 109400 | 1586 | 94710 | 1671 | 71140 | 0 | 0 | 2004 | 61000 | 2004 | 53600 |
| | MODEL E | 1480 | 108400 | 1479 | 75280 | 1424 | 71030 | 0 | 0 | 1496 | 60830 | 1496 | 53780 |
| | MODEL F | 1294 | 79140 | 1294 | 82440 | 1212 | 82460 | 0 | 0 | 1447 | 80970 | 1447 | 66320 |
| | MODEL G | 1288 | 79400 | 1288 | 82550 | 1204 | 82590 | 0 | 0 | 1445 | 80760 | 1445 | 66080 |
| MODGUYAN | MODEL A | 1066 | 77460 | 1069 | 80100 | 1147 | 78910 | 8 | 2292 | 1093 | 70150 | 1090 | 52290 |
| | MODEL B | 1095 | 77150 | 1098 | 77380 | 1130 | 78630 | 44 | 3677 | 1074 | 77120 | 1072 | 62930 |
| | MODEL C | 1344 | 76230 | 1348 | 75780 | 1230 | 77630 | 51 | 3664 | 1528 | 77630 | 1549 | 67360 |
| | MODEL D | 1081 | 78370 | 1084 | 75080 | 1057 | 67860 | 88 | 5035 | 1187 | 65090 | 1198 | 53780 |
| | MODEL E | 1118 | 77940 | 1122 | 74490 | 1102 | 68110 | 93 | 5002 | 1288 | 64790 | 1302 | 53400 |
| | MODEL F | 1336 | 79170 | 1341 | 82310 | 1250 | 82260 | 66 | 3524 | 1422 | 79450 | 1436 | 66150 |
| | MODEL G | 1339 | 79490 | 1344 | 82480 | 1295 | 82440 | 67 | 3504 | 1424 | 79460 | 1437 | 66060 |

**Table 11** (Cont'd)

PIPING FORCES

| Model | | MEMBER NUMBER | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 25 | | 26 | | 27 | | 28 | | 29 | | 30 | |
| | | P | M | P | M | P | M | P | M | P | M | P | M |
| Exact | | 1454 | 57430 | 1468 | 51970 | 68 | 3718 | 1781 | 47670 | 1800 | 41430 | 1815 | 42340 |
| GUYAN | MODEL A | 1071 | 25480 | 1071 | 14930 | 0 | 0 | 1071 | 19270 | 1071 | 29400 | 1071 | 33580 |
| | MODEL B | 1005 | 61450 | 1005 | 46270 | 0 | 0 | 1005 | 22980 | 1005 | 26050 | 1005 | 30700 |
| | MODEL C | 2584 | 61010 | 2584 | 67090 | 0 | 0 | 2584 | 67090 | 2584 | 50790 | 2584 | 49820 |
| | MODEL D | 2004 | 50950 | 2004 | 52850 | 0 | 0 | 2004 | 52850 | 2004 | 39330 | 2004 | 40060 |
| | MODEL E | 1496 | 51610 | 1496 | 55380 | | | 1963 | 55380 | 1963 | 42850 | 1963 | 43070 |
| | MODEL F | 1447 | 58970 | 1447 | 53760 | 0 | 0 | 1827 | 50850 | 1827 | 42180 | 1827 | 42980 |
| | MODEL G | 1445 | 58690 | 1445 | 53420 | 0 | 0 | 1820 | 50390 | 1820 | 42000 | 1820 | 42890 |
| MOD. GUYAN | MODEL A | 1088 | 41010 | 1085 | 29760 | 5 | 1079 | 1030 | 11740 | 1027 | 25390 | 1024 | 31660 |
| | MODEL B | 1071 | 54400 | 1070 | 46570 | 13 | 2552 | 1082 | 36280 | 1084 | 27530 | 1086 | 31200 |
| | MODEL C | 1565 | 62440 | 1587 | 59200 | 21 | 3615 | 2042 | 57700 | 2069 | 47240 | 2090 | 46530 |
| | MODEL D | 1207 | 47790 | 1218 | 43160 | 56 | 3776 | 1454 | 39410 | 1470 | 34370 | 1482 | 35340 |
| | MODEL E | 1313 | 47670 | 1327 | 43410 | 64 | 3856 | 1614 | 40630 | 1632 | 36940 | 1646 | 37610 |
| | MODEL F | 1446 | 59160 | 1461 | 53840 | 64 | 3021 | 1780 | 49630 | 1800 | 42100 | 1815 | 42790 |
| | MODEL G | 1448 | 59000 | 1462 | 53610 | 63 | 2983 | 1780 | 49330 | 1799 | 42030 | 1814 | 42800 |

Table 11 (Cont'd)

PIPING FORCES

| Model | | MEMBER NUMBER 31 P | 31 M | 32 P | 32 M | 33 P | 33 M | 34 P | 34 M | 35 P | 35 M | 36 P | 36 M |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Exact | | 1737 | 44290 | 1739 | 50700 | 121.3 | | 905.3 | | 849.5 | | 81.6 | |
| GUYAN | MODEL A | 1329 | 34060 | 1329 | 27650 | 138.8 | | 53.8 | | 509.4 | | 16.6 | |
| | MODEL B | 1520 | 37990 | 1520 | 62180 | 100.6 | | 1539. | | 1768. | | 99.7 | |
| | MODEL C | 7428 | 46310 | 2428 | 80590 | 116.8 | | 1804. | | 642.2 | | 116.9 | |
| | MODEL D | 2203 | 40660 | 2103 | 77260 | 381.8 | | 3097. | | 1069. | | 156.1 | |
| | MODEL E | 2515 | 40890 | 2516 | 77920 | 377.4 | | 3209. | | 1024. | | 155.4 | |
| | MODEL F | 1743 | 44850 | 1743 | 54060 | 232.0 | | 1291. | | 1022. | | 77.4 | |
| | MODEL G | 1745 | 44760 | 1745 | 55220 | 102.1 | | 976.7 | | 961.4 | | 84.5 | |
| MOD. GUYAN | MODEL A | 1503 | 35240 | 1504 | 35740 | 68.1 | | 66.9 | | 505.3 | | 17.3 | |
| | MODEL B | 1470 | 40270 | 1471 | 64850 | 114.5 | | 257.2 | | 813.8 | | 109.6 | |
| | MODEL C | 1792 | 45980 | 1796 | 77740 | 130.7 | | 349.3 | | 1001 | | 126.0 | |
| | MODEL D | 1457 | 44870 | 1459 | 71640 | 210.2 | | 931.8 | | 768.9 | | 165.7 | |
| | MODEL E | 1543 | 44710 | 1546 | 71260 | 210.0 | | 1038.0 | | 733.4 | | 164.6 | |
| | MODEL F | 1732 | 45030 | 1734 | 53510 | 124.1 | | 831.9 | | 927.5 | | 81.9 | |
| | MODEL G | 1737 | 44980 | 1739 | 55020 | 125.1 | | 834.3 | | 914.7 | | 86.9 | |

Table 11(Cont'd)

PIPING FORCES

| Model | | MEMBER NUMBER | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | 37 | | 38 | | | | | | | | | |
| | | P | M | P | M | P | M | P | M | P | M | P | M |
| Exact | | 17.4 | | 3.4 | | | | | | | | | |
| GUYAN | MODEL A | 13.8 | | 1.6 | | | | | | | | | |
| | MODEL B | 17.7 | | 3.2 | | | | | | | | | |
| | MODEL C | 18.1 | | 4.3 | | | | | | | | | |
| | MODEL D | 14.9 | | 3.3 | | | | | | | | | |
| | MODEL E | 14.9 | | 3.3 | | | | | | | | | |
| | MODEL F | 17.8 | | 3.6 | | | | | | | | | |
| | MODEL G | 17.9 | | 3.6 | | | | | | | | | |
| MOD. GUYAN | MODEL A | 16.1 | | 2.3 | | | | | | | | | |
| | MODEL B | 17.8 | | 3.5 | | | | | | | | | |
| | MODEL C | 18.1 | | 4.2 | | | | | | | | | |
| | MODEL D | 13.8 | | 3.0 | | | | | | | | | |
| | MODEL E | 13.7 | | 2.9 | | | | | | | | | |
| | MODEL F | 17.8 | | 3.5 | | | | | | | | | |
| | MODEL G | 17.9 | | 3.6 | | | | | | | | | |

## TABLE 12

## PERCENTAGE ERRORS IN PIPING ELEMENTS

| Case (Table 9) | GUYAN REDUCTION | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | PIPE ELEMENTS | | | | | | | | PIPE SUPPORTS | | |
| | 1 | | 5 | | 12 | | 18 | | 7-37 | 18-38 | 13-34 |
| | P | M | P | M | P | M | P | M | | | |
| A | -18. | 12.7 | +55.2 | +2.9 | 545.2 | +20.3 | +31.8 | +3.8 | +14.4 | -94.1 | -79.7 |
| B | -3.4 | -25.5 | -48.8 | -18.2 | 505.4 | +22.0 | +5.8 | -3.7 | -17.1 | +7[illegible].0 | +22.2 |
| C | -4.1 | -26.0 | -49.2 | -18.8 | 479.5 | +120.0 | 148.6 | +16.2 | -3.7 | +99.3 | +43.3 |
| D | -31.4 | -7.6 | +32.5 | -16.9 | 217.5 | +1.8 | +51.2 | +65.3 | 214.8 | 242.1 | +91.3 |
| E | +26.2 | -7.7 | -1.1 | -27.3 | 183.7 | +5.5 | +40.8 | +64.0 | 211.1 | 254.5 | +90.4 |
| F | +13.9 | -0.1 | -5.5 | -0.4 | 170.5 | +6.1 | +123.5 | +1.7 | 91.3 | +42.6 | -5.1 |
| G | +3.5 | +1.6 | -15.6 | +1.1 | -39.2 | +2.7 | +22.8 | +1.9 | -15.8 | 7.9 | +3.6 |

TABLE 12 (Cont'd)

PERCENTAGE ERRORS IN PIPING ELEMENTS

| Case (Table 9) | MODIFIED GUYAN | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|
| | PIPE ELEMENTS | | | | | | | | PIPE SUPPORTS | | |
| | 1 | | 5 | | 12 | | 18 | | 7-37 | 18-38 | 13-34 |
| | P | M | P | M | P | M | P | M | | | |
| A | -.4 | +0.2 | +.4 | 0 | +6.0 | +.4 | -18.0 | +0.4 | -43.9 | -92.6 | -78.8 |
| B | -4.6 | -7.9 | -12.0 | -6.7 | +13.3 | +.5 | -8.0 | -2.0 | -5.6 | -71.6 | +34.3 |
| C | -6.6 | -9.7 | -13.7 | -8.5 | +10.2 | -1.3 | -1.1 | -3.2 | +7.7 | -61.4 | +54.4 |
| D | -16.8 | -18.0 | -17.7 | -16.9 | -13.9 | -4.5 | -17.8 | +6.0 | 73.3 | +2.9 | 103.1 |
| E | -16.8 | -17.5 | -17.4 | -17.4 | -13.9 | -5.2 | -17.2 | +5.6 | 73.1 | +14.7 | 101.7 |
| F | +1.0 | +1.2 | +1.5 | +1.2 | +6.0 | +1.3 | +2.7 | +1.4 | 2.3 | -8.1 | +0.4 |
| G | +1.2 | +1.5 | +1.3 | +1.5 | +4.8 | +1.6 | +2.9 | +1.8 | 3.1 | -7.8 | +6.5 |

The following conclusion may be drawn from the data on Table 12:

(i) When reduction schemes are used the bending moment data appears to be most reliable with the thrust and pipe support data the poorest.

(ii) The Modified Guyan Reduction gives significantly better results than the Guyan Reduction. The maximum error for pipe thrust and moment for the latter case is 18% no matter which reduction scheme is used. The maximum error for pipe thrust and moment is 545% when Guyan Reduction is used. The comparable maximum errors for pipe support loads are 103% and 254% respectively for the both reduction methods.

(iii) Reduction schemes F and G yield good results (maximum error 8.1%) when Modified Guyan Reduction is used while scheme G still yields rather poor (39% error in pipe 12 thrust) results for Guyan Reduction.

iv) The good results are obtained for scheme G because degrees of freedom normal to the restraints provided by the pipe supports are included as dynamic degrees of freedom. The degree of freedom active in the direction of the pipe support should probably be included to achieve good results for the pipe support loads with Guyan Reduction.

## 4.4 REACTOR CONTAINMENT BUILDING

The reactor containment building shown in Figure 6 is subjected to the response spectrum of Figure 3 in the X direction. Plate elements 1 through 24 are used to model the exterior containment. The reactor floor is modeled with plate elements 25 through 32. A wall supporting the plate is modeled with plate element 33 through 40. All plate elements are taken as concrete with a thicknesss of 36 inches. Beam elements 41 through 43 are used to model the reactor. Symmetric boundary conditions are specified at all nodes at (Y = 0). These boundary conditions restrain the following degrees of freedom: displacement in the Y direction and rotations about the X and Z axes.

An exact solution is obtained and solutions using both Guyan and Modified Guyan Reduction for the reduced models shown in Table 13. There are 102 unrestrained translational degrees of freedom.

Table 13

Reduction Schemes for Containment Building

| Case | No. of Primary Degrees of Freedom | Degrees of Freedom Retained |
|---|---|---|
| A | 3 | 21-X;34-X;50-X |
| B | 6 | 13-Y;21-X;34-X;40-Z;44-Z;50-X |
| C | 13 | 11-X;11-Z;15-X;15-Z;21-X;21-Z;34-X;<br>40-X;40-Z;42-Y;44-X;44-Z;50-X |
| D | 23 | 11-X;11-Z;13-X;13-Y;15-X;15-Z;16-X;<br>16-Z;18-X;18-Y;20-X;20-Z;21-X;21-Z;<br>33-X;34-X;40-X;40-Z;42-X;42-Y;44-X;<br>44-Z;50-X |
| E | 46 | 11-X;11-Z;13-X;13-Y;15-Y;15-Z;16-X;<br>16-Z;16-YY;17-X;17-Y;17-Z;17-YY;<br>17-ZZ;18-X;18-Y;18-Z;18-XX;18-YY;<br>18-ZZ;19-X;19-Y;19-Z;19-XX;19-YY;<br>19-ZZ;20-X;20-Z;20-YY;21-X;21-Z;21-YY;<br>33-X;33-Z;33-YY;34-X;34-Z;34-YY;40-X;<br>40-Z;42-X;42-Y;44-X;44-Z;50-X |

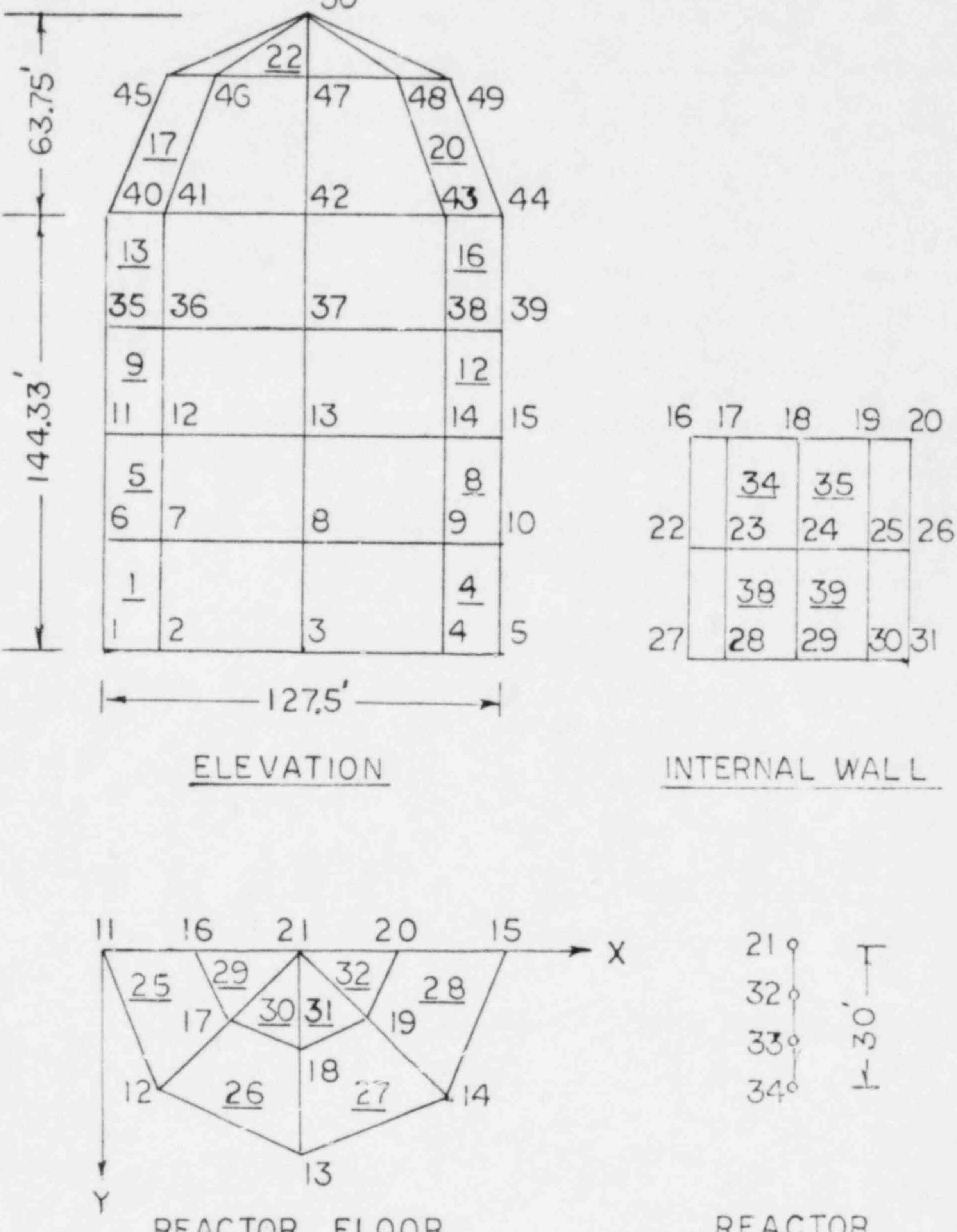


Fig. 6 CONTAINMENT BUILDING

Ten modes are included in the solution with the resulting frequencies shown in Table 14.

Table 14

Frequencies for Containment Building

| Case | Frequency (CPS) for Mode 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
|---|---|---|---|---|---|---|---|---|---|---|
| EXACT | 3.84 | 9.53 | 9.89 | 10.2 | 10.3 | 10.7 | 10.9 | 14.5 | 14.8 | 15.8 |
| A | 4.09 | 13.3 | 113. | - | - | - | - | - | - | - |
| B | 4.08 | 12.6 | 13.6 | 20.8 | 22.2 | 113. | - | - | - | - |
| C | 3.98 | 9.55 | 11.2 | 12.7 | 13.0 | 13.6 | 14.4 | 23.7 | 24.5 | 29.5 |
| D | 3.89 | 9.55 | 10.4 | 12.5 | 13.2 | 13.4 | 13.6 | 20.1 | 22.5 | 25.4 |
| E | 3.89 | 9.54 | 10.4 | 12.5 | 13.1 | 13.3 | 13.5 | 20.0 | 22.3 | 23.3 |

As with the piping system it appears that several of the modes from the exact model are missing from the reduced model.

Stresses in typical plate element and the peak bending moment in the reactor beam model are shown in Table 15.

Table 15

Stresses and Moments in Containment Building

| Case | | Peak Stress in Plate Element 1 | 5 | 13 | 25 | 29 | 33 | Peak Moment in Beam |
|---|---|---|---|---|---|---|---|---|
| EXACT | | 649 | 481 | 186 | 72 | 83 | 138 | 47,740 |
| GUYAN | A | 545 | 419 | 225 | 65 | 175 | 109 | 2,527,000 |
| | B | 548 | 426 | 244 | 63 | 180 | 109 | 2,738,000 |
| | C | 600 | 465 | 222 | 87 | 149 | 122 | 2,148,000 |
| | D | 635 | 475 | 192 | 46 | 104 | 140 | 350,900 |
| | E | 635 | 475 | 192 | 56 | 89 | 141 | 45,640 |
| M. GUYAN | A | 569 | 420 | 166 | 60 | 24 | 219 | 103,300 |
| | B | 570 | 422 | 167 | 60 | 29 | 117 | 56,450 |
| | C | 601 | 444 | 173 | 68 | 82 | 127 | 45,170 |
| | D | 632 | 466 | 178 | 71 | 93 | 135 | 46,410 |

The plate stresses for all of the elements appear to be quite good. Modified Guyan Reduction appears to be a little better than Guyan Reduction but not to the same degree as for the first three problems considered. The very significant difference shows up with the peak moment in the beam model of the reactor. Guyan Reduction yields totally unreliable results while Modified Guyan gives good results except for Reduction scheme A. As with the stiff pipe supports the floor provides a very stiff support for the reactor and undoubtedly leads to these very poor results. Guyan Reduction gives a reasonable value for this moment only when the rotational degree of freedom connecting the beam to the plate is included as a primary degree of freedom. Note that there is no mass attached to this degree of freedom.

## 4.5 COMPUTER REQUIREMENTS

For the four sample problems considered the Modified Guyan and Guyan Reduction solutions required almost exactly the same computer time. Both of the reduction methods required solution times about 3 times longer than would be required using the complete SAPV solution. Undoubtedly some of this difference resulted from "patching in" the reduction modification. Changes were made in a manner to expedite the code modification rather than to develop an efficient code. If this facet of the problem were to be investigated further it would be recommended that the current code be used to pinpoint the calculations that require the most time. This could then be used to evaluate the likelihood that the SAPV with reduction would be expected to run faster than SAPV without reductions.

## 5.0 CONCLUSIONS AND RECOMMENDATIONS

Based upon the four sample problems the following conclusions are made:

1. The errors resulting from Guyan Reduction are significantly larger than those which result from Modified Guyan Reduction. For example the errors in member bending moments in the ten story structure are always less than 2% when Modified Guyan Reduction is used even when only 3 degrees of freedom are retained. The same errors for Guyan Reduction are 100% when 3 degrees of freedom are retained and only become acceptable when 6 or more degrees of freedom are retained.
2. For a given reduction scheme the errors resulting from Modified Guyan Reduction are much more uniform for all members in the problem than when Guyan Reduction is used.

Again using the ten story frames as an example for the same reduction scheme (3 degrees of freedom retained) the error in the bending moments of members 20 and 41 are 100% and 1% when Guyan Reduction is used. The corresponding errors with Modified Guyan Reduction are 1.3% and 1.6%. This variation in error which Guyan Reduction exhibits would make it very difficult to validate a given set of results. If we looked at the forces in member 41, the results for the 3 degrees of freedom model would appear to be quite good.

3. Frequencies calculated with Guyan Reduction appear to be quite good as long as the degrees of freedom retained can adequately represent the mode shapes of interest.
4. The bending moments in piping systems predicted based upon Guyan Reduction are more reliable than are the axial pipe loads. For example element 12 has errors in the axial pipe load in the range of 200% to 300% (for different degrees of freedom retained) while the same errors in moments are in the range of 5% to 20%. The same errors when Modified Guyan Reduction is used are 6 to 14% for axial load and 1% to 5% for bending moments. This again illustrates the better and more uniform results obtained using Modified Guyan Reduction.
5. Much larger errors result with the reduction schemes when there are both "stiff" and "soft" elements in the system. This occurs in the piping problem (supports relative to pipe) and in the containment building problem (floor in plane stiffness to reactor beam model). In either case Guyan Reduction gives totally unreliable results unless

care is taken to isolate the stiff element by containing it within degrees of freedom retained. The same effect is also observed with Modified Guyan Reduction but again the errors are much smaller.

The following recommendations are made:

1. Computer codes which contain Guyan Reduction should be modified to incorporate Modified Guyan Reduction.

2. If computer solutions are to be made using Guyan Reduction care should be taken to identify the stiff elements in the model and to retain all degrees of freedom around these elements as primary degrees of freedom.

3. Additional parametric studies should be made with the following model parameters varied: structural geometry and type; relative stiffness of members; and relative mass. Based in these studies criteria should be developed relating errors to the reduction scheme used. This should be done for both Guyan and Modified Guyan Reduction. The results of such a study would serve two purposes. First, they would provide some insight to the errors which may exist in Guyan Reduction runs made. Second, such criteria could provide the basis for selecting the "optimum" set of primary degrees of freedom to be retained.

## REFERENCES


1. Guyan, R. J., "Reduction of Stiffness and Mass Matrices", American Institute of Aeronautics and Astronautics Journal, Vol. 3, No. 2, February 1965.

2. Irons, B. M., "Structural Eigenvalue Problems: Elimination of Unwanted Variables", American Institute of Aeronautics and Astronautics, Vol. 3, No. 5, May 1965.

3. Kaufman, S., and Hall, D. B., "Reduction of Mass and Loading Matricies", American Institute of Aeronautics and Astronautics, Vol. 6, No. 3, March 1968.

4. Miller, C. A., "Dynamic Reduction of Structural Models", Proceedings American Society of Civil Engineers, Structural Division, Vol. 106, No. ST10, October 1980.

5. Bathe, R. J., Wilson, F. L., Peterson, F. E., "A Structural Analysis Program for Static and Dynamic Response of Linear Systems", Earthquake Engineering Research Center, University of California, June 1973.

6. Bathe, K.J., Wilson, E. L., "Numerical Methods in Finite Element Analysis", Prentice-Hall, 1976.

7. Bezler, P., Hartzman, M. Reich, M., "Piping Benchmark Problems", NUREG/CR-1677, Vol. 1, Brookhaven National Laboratory, August 1980.*


---

*Available for purchase from the NRC/GPO Sales Program, U.S. Nuclear Regulatory Commission, Washington, DC 20555, and the National Technical Information Service, Springfield, VA 22161.

# APPENDIX A

# LISTING OF SAPV

# MODIFICATIONS

POOR ORIGINAL



```
      PROGRAM SAP5 (INPUT,OUTPUT,TAPE2,TAPE3,TAPE4,TAPE5=INPUT,TAPE6=     CORR2      1
     1 OUTPUT,TAPE7,TAPE8,TAPE9,TAPE10,TAPE11,TAPE12,                     CORR2      2
     2 TAPE13,TAPE14,TAPE15,TAPE16,TAPE17,TAPE21,TAPE1,TAPE18)            CAM        1
C  ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** *         SAP5       2
C                                                                         SAP5       3
C                                SAP5                                     SAP5       4
C                   A STRUCTURAL ANALYSIS PROGRAM                         SAP5       5
C        FOR STATIC AND DYNAMIC RESPONSE OF LINEAR SYSTEMS                SAP5       6
C                                                                         SAP5       7
C             K.J. BATHE , E.L. WILSON , F.E. PETERSON                    SAP5       8
C              UNIVERSITY OF CALIFORNIA , BERKELEY                        SAP5       9
C                                                                         SAP5      10
C                CDC 7600 CONVERSION BY D. GARDNER                        CORR3      1
C                BROOKHAVEN NATIONAL LABORATORY                           CORR3      2
C                        UPTON, N. Y.                                     CORR3      3
C                      FEBRUARY, 1977                                     CORR3      4
C                                                                         SAP5      20
C     USC IBM-CALCOMP MODIFICATION --- MARCH,1976                         SAP5      21
C                                                                         SAP5      22
C                                                                         SAP5      23
C  ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** ** *         SAP5      24
C                                                                         SAP5      25
      LOGICAL ELPRT,ELPCH,GENPRT,GENPCH,DEFPCH,GEOST                      SAP5      27
      COMMON/JUNK/ HED(18),JUK(200)                                       COMSIZE    1
      COMMON /ELPAR/ NPAR(14),NUMNP,MBAND,NELTYP,N1,N2,N3,N4,N5,MTOT,NEQ SAP5      29
      COMMON/EM/ QQQ(3000)                                                COMSIZE    2
      COMMON /DYN/    IDUS(11),NDYN,THOPCH,THSPCH                         SAP5      31
      COMMON /TAPES/ NQQ(6)                                               SAP5      32
      COMMON /EXTRA/ MODEX,NT8,NIOSV,NT10,KEQB,NUMEL,T(12),S(12)          SAP5      33
      COMMON /SOL/    NBLOCK,NEQB,LL,NF,IDUM,NEIG,NAD,NVV,ANORM,NFO       SAP5      34
      COMMON /CTL/ GRAV,ELPRT,ELPCH,GENPRT,GENPCH,DEFPCH,IWTCG,           SAP5      35
     1   MINBND,IPLT,NPLOTS,NPLT                                          CORR3      5
      COMMON /GEOSTF/ GEOST,NELGEO                                        SAP5      37
      COMMON/SEIS/ NRSC,KAPG,NRS,IT,DDATE,JFLAG,JNK(2)                    COMSIZE    3
      COMMON/AA/A(75001)                                                  CAM        2
      COMMON/CONTL/NRDOF,NTYP,NST,MBW,NS,NS1,NS2,NS3,NOT,MBS,MODE,        CAM        3
     1 MD,MBLRED,NSEQ,NTO                                                 CAM        4
      LEVEL 2, A                                                          LEVEL2     2
      MTOT=75000                                                          CAM        5
C                                                                         SAP5      40
C                                                                         SAP5      41
C          CDC ROUTINE ERRSET IS CALLED TO ALLOW PROGRAM TO               CORR3      6
C          READ PAST UP TO 100 DATA ERRORS.  IF AT LEAST ONE              CORR3      7
C          IS FOUND THE RUN IS STOPPED AFTER STATEMENT 900.               CORR3      8
C                                                                         SAP5      44
      CALL ERRSET(KOUNT,100)                                              CORR3      9
C                                                                         SAP5      46
      NPLOTS=0                                                            SAP5      47
    5 CALL  HEDPRT(KDYN)                                                  SAP5      49
      JFLAG=0                                                             SAP5      50
      WRITE (13) HED                                                      SAP5      51
C                                                                         SAP5      52
C     RE-START FOR MORE PLOTS                                             SAP5      53
C                                                                         SAP5      54
      IF (IPLT .GE. 0) GO TO 14                                           SAP5      55
      DO 7 I=1,7                                                          SAP5      56
    7 T(I)=T(I)                                                           SAP5      57
```

```
      C     S(I)=S(I)                                                    SAP5      58
            READ (3) LL,NF,KDYN,NEQB,NEQ,NBLOCK                          SAP5      59
            CALL  SAPLOT(LL,NF,KDYN,NEQB,NEQ,NBLOCK)                     SAP5      60
            CALL  SECOND(T(8))                                           SAP5      61
            DO 8 I=9,12                                                  SAP5      62
          8 T(I)=T(8)                                                    SAP5      63
      C     S(I)=S(8)                                                    SAP5      64
            GO TO 90                                                     SAP5      65
      C                                                                  SAP5      66
      C     RE-START MODE ACTIVATED IF  NDYN.EQ.-2  OR  NDYN.EQ.-3       SAP5      67
      C                                                                  SAP5      68
         14 IF(NDYN.LT.0) GO TO 20                                       SAP5      69
      C                                                                  SAP5      70
      C     I N P U T   J O I N T   D A T A                              SAP5      71
      C                                                                  SAP5      72
            N2=N1+6*NUMNP                                                SAP5      73
            N3=N2+NUMNP                                                  SAP5      74
            N4=N3+NUMNP                                                  SAP5      75
            N5=N4+NUMNP                                                  SAP5      76
            N6=N5+NUMNP                                                  SAP5      77
            CALL  ERROR(N6)                                              SAP5      78
      C                                                                  SAP5      79
            CALL INPUTJ(A(N1),A(N2),A(N3),A(N4),A(N5),NUMNP,NEQ)         CORR2      4
      C                                                                  SAP5      82
      C     STORE THE NODAL COORIDINATES FOR LATER USE                   SAP5      83
      C                                                                  SAP5      84
            WRITE (13) (A(I),I=N2,N5)                                    SAP5      85
      C                                                                  SAP5      86
      C     F O R M   E L E M E N T   S T I F F N E S S E S              SAP5      87
      C                                                                  SAP5      88
      C     INPUT PRIMARY DOF AND REORDER ALL DOF SO THAT PRIMARY ARE FIRST   CAM   6
      C                                                                  CAM        7
            CALL REDUCI(A(N1),A(N6),NTYP,NUMNP,NRDOF)                    CAM        8
            CALL  SECOND(T(2))                                           SAP5      89
      C                                                                  SAP5      90
            MBAND=0                                                      SAP5      91
            NUMEL=0                                                      SAP5      92
            REWIND 1                                                     SAP5      93
            REWIND 2                                                     SAP5      94
      C                                                                  SAP5      95
            DO 900 M=1,NELTYP                                            SAP5      96
            READ  (5,1001) NPAR                                          SAP5      97
      C+++  DATA PORTHOLE SAVE                                           SAP5      98
            IF(MODEX.EQ.1) WRITE (NT8) NPAR                              SAP5      99
            NT1=1                                                        SAP5     100
            IF (MINBND .GE. 1 .AND. MODEX .EQ. 0) NT1=4                  SAP5     101
            IF (MINBND .GE. 1 .AND. MODEX .EQ. 1) NT1=7                  SAP5     102
            WRITE (NT1) NPAR                                             SAP5     103
            NUMEL=NUMEL+NPAR(2)                                          SAP5     104
            MTYPE=NPAR(1)                                                SAP5     105
      C                                                                  SAP5     106
            CALL  ELTYPE(MTYPE)                                          SAP5     107
      C                                                                  SAP5     108
        900 CONTINUE                                                     SAP5     109
            IF(KOUNT.GE.1) STOP                                          CORR3     10
      C                                                                  SAP5     110
      C     ADD AN END-OF-FILE FLAG ON FILE 13                           SAP5     111
```

POOR ORIGINAL



```
C
      M=-1
      IF (IPLT .GT. 0)
     XWRITE (13) M,M
C
C     MINIMIZE THE BANDWIDTH
C
      IF (MINBND .GT. 0) CALL  SAPMIN
C
C     D E T E R M I N E   B L O C K S I Z E
C
C     ADDSTF
C
      NEQB=(MTOT - 4*LL)/(MBAND + LL + 1)/2
C
C     OVER-RIDE THE SYSTEM MATRIX BLOCKSIZE WITH THE INPUT (NON-ZERO)
C     VALUE, KEQB.
C     THIS OVER-RIDE ENTRY IS TO ALLOW PROGRAM CHECKING OF MULTI-
C     BLOCK ALGORITHMS WITH WHAT WOULD NORMALLY BE ONE BLOCK DATA.
C
      IF(KEQB.LT.NEQB) NEQB = KEQB
C
      GO TO  (690,700,700,700,730,700), KDYN
C
C     STATIC SOLUTION
C
  690 CONTINUE
      NEQB1=(MTOT - MBAND)/(2*(MBAND+LL) + 1)
      NEQB2=(MTOT - MBAND - LL*(MBAND-2))/(3*LL + MBAND + 1)
      IF (NEQB1.LT.NEQB) NEQB=NEQB1
      IF (NEQB2.LT.NEQB) NEQB=NEQB2
      NBLOCK = (NEQ-1)/NEQB +1
      IF(NEQB.GT.NEQ) NEQB=NEQ
      IF(IPLT.EQ.0)GO TO 695
      NTOT=20*NUMNP+NEQB*NBLOCK*LL
      IF(MTOT.GT.NTOT)GO TO 790
      WRITE(6,2020)NTOT
      MODEX=1
  695 NTOT=6*NUMNP+NEQB*NBLOCK*LL+1
      IF(MTOT.GT.NTOT)GO TO 790
      WRITE(6,2030)NTOT
      MODEX=1
      GO TO 790
C
C     EIGENSOLUTION
C
C
C        1.  DETERMINANT SEARCH ALGORITHM
C
  700 IF (NEQB.LT.NEQ) GO TO 710
      NIM=3
      NC=NF + NIM
      NVM=6
      NCA=NEQ*MAX0(MBAND,NC)
      NTOT=NCA + 4*NEQ + 2*NVM*NEQ + 5*NC
      NEIG=0
      IF(NTOT.LE.MTOT) GO TO 720
C
```

POOR ORIGINAL



```
C        2.  SUBSPACE ITERATION ALGORITHM
C
  710 NY=MIN0(2*NF,NF+8)
      IF (NAD.NE.0) NV=NAD
      NEQB1=(MTOT - MBAND)/(2*MBAND + 1)
      NEQB2=(MTOT - MBAND - 2*NV - NV*(MBAND-2))/(3*NV + MBAND + 1)
      NEQB3=(MTOT - 3*NV*NV - 3*NV)/(2*NV + 1)
      NEQB4=(MTOT - 6*NV)/(1 + MBAND)
      IF (NEQB1.LT.NEQB) NEQB=NEQB1
      IF (NEQB2.LT.NEQB) NEQB=NEQB2
      IF (NEQB3.LT.NEQB) NEQB=NEQB3
      IF (NEQB3.LT.NEQB) NEQB=NEQB3
      IF (NEQB4.LT.NEQB) NEQB=NEQB4
      NEIG=1
C
  720 CONTINUE
      NBLOCK = (NEQ-1)/NEQB +1
      IF (NEQB.GE.NEQ) NEQB=NEQ
C
C     HISTORY OR SPECTRUM ANALYSIS
C
      KREM = 1000
      NTOT = NBLOCK*NEQB*NF + KREM
      IF(MTOT.LT.NTOT)
     XWRITE (6,320) NTOT
      IF(IPLT.EQ.0)GO TO 725
      NTOT=20*NUMNP+NEQB*NBLOCK*NF+2*NF
      IF(MTOT.GE.NTOT)GO TO 790
      WRITE(6,2020)NTOT
      MODEX=1
  725 NTOT=6*NUMNP+NEQB*NBLOCK*NF+NF
      IF(MTOT.GE.NTOT)GO TO 790
      WRITE(6,2030)NTOT
      MODEX=1
      GO TO 790
C
C     STEP-BY-STEP DIRECT INTEGRATION
C
  730 CONTINUE
C     DISPLACEMENT COMPONENTS FOR DIRECT OUTPUT ('NSD')
      NN2 = NEQ
C     DISPLACEMENT COMPONENTS REQUIRED FOR RECOVERY OF ALL OF THE
C     REQUESTED ELEMENT STRESS COMPONENTS ('NSS')
      NN3 = NEQ
C
C        1.  DECOMPOSITION
C
      NEQB1 = (MTOT-NN2-NN3-NEQ-MBAND)/(2*MBAND+1)
C
C        2.  TIME INTEGRATION PHASE
C
C
      NEQB2 = (MTOT-MBAND-2*(NN2+NN3)-5*NEQ)/(MBAND+1)
C
      IF(NEQB1.LT.NEQB) NEQB = NEQB1
      IF(NEQB2.LT.NEQB) NEQB = NEQB2
      IF(NEQB.GT.NEQ)   NEQB = NEQ
```

POOR ORIGINAL



```
      NBLOCK = (NEQ-1)/NEQB +1                                          SAP5     226
C                                                                       SAP5     227
C        3.  INPUT PHASE                                                SAP5     228
C                                                                       SAP5     229
C     NUMBER OF TIME FUNCTIONS ( NFN*)                                  SAP5     230
C     NUMBER OF TIME FUNCTIONS (*NFN*)                                  SAP5     231
      NN2 = 10                                                          SAP5     232
C     MAXIMUM NUMBER OF FUNCTION DEFINITION POINTS (*MXLP*)             SAP5     233
      NN3 = 40                                                          SAP5     234
C                                                                       SAP5     235
      NN4 = 6*NUMNP + 2*NN2*NEQ                                         SAP5     236
      IF(NN4.GT.MTOT)                                                   SAP5     237
     XWRITE (6,320) NN4                                                 SAP5     238
      NN4 = NEQ*2*(NN2+1) + NN2*(1+2*NN3)                               SAP5     239
      IF(NN4.GT.MTOT)                                                   SAP5     240
     XWRITE (6,320) NN4                                                 SAP5     241
C                                                                       SAP5     242
  790 IF (NEQB .GE. 2) GO TO 792                                        SAP5     243
      IF (NEQ .LT. 3) GO TO 792                                         SAP5     244
      WRITE (6,2010) NEQB                                               SAP5     245
      NEQB=2                                                            SAP5     246
      NBLOCK=(NEQ-1)/NEQB + 1                                           SAP5     247
      MODEX=1                                                           SAP5     248
  792 CONTINUE                                                          SAP5     249
C                                                                       SAP5     250
C     I N P U T   N O D A L   L O A D S                                 SAP5     251
C                                                                       SAP5     252
      NSB=NEQB*NBLOCK                                                   SAP5     253
      N2=N1+6*NUMNP                                                     SAP5     254
      N3=N2+LL*NSB                                                      SAP5     255
      N4=N3+NSB                                                         SAP5     256
      CALL  ERROR(N4)                                                   SAP5     257
      WRITE (6,201) NEQ,MBAND,NEQB,NBLOCK                               SAP5     258
C                                                                       SAP5     259
      CALL  SECOND(T(3))                                                SAP5     260
C                                                                       SAP5     261
      CALL  INL(A(N1),A(N2),A(N3),NUMNP,NEQB,NSB,LL,NBLOCK)             SAP5     262
C                                                                       SAP5     263
      CALL  SECOND(T(4))                                                SAP5     264
C                                                                       SAP5     265
C     CHECK FOR UNRESTRAINED DEGREES OF FREEDOM                         SAP5     266
C                                                                       SAP5     267
      N3=N2+NEQ                                                         SAP5     268
      CALL CHICK(A(N1),A(N2),NEQ,NUMEL,NELTYP,NUMNP)                    CORR2      5
C                                                                       SAP5     270
C     F O R M   T O T A L   S T I F F N E S S                           SAP5     271
C                                                                       SAP5     272
      NE2B=2*NEQB                                                       SAP5     273
      N2=N1+NEQB*MBAND                                                  SAP5     274
      N3=N2+NEQB*LL                                                     SAP5     275
      N4=N3+4*LL                                                        SAP5     276
      NN2=N1+NE2B*MBAND                                                 SAP5     277
      NN3=NN2+NE2B*LL                                                   SAP5     278
      NN4=NN3+4*LL                                                      SAP5     279
C                                                                       SAP5     280
      CALL  ADDSTF (A(N1),A(NN2),A(NN3),A(NN4),NUMEL,NBLOCK,NE2B,LL,MBAN SAP5    281
     XD,ANORM,NVV)                                                      SAP5     282
```

```
      C                                                                 SAP5  283
      C     W T   A N D   C G   C A L C U L A T I O N S               SAP5  284
      C                                                                 SAP5  285
            IF (IWTCG .EQ. 0) GO TO 18                                  SAP5  286
            N2=N1+NUMNP*6                                               SAP5  287
            N3=N2+NUMNP                                                 SAP5  288
            N4=N3+NUMNP                                                 SAP5  289
            N5=N4+NUMNP                                                 SAP5  290
            N6=N5+NSB                                                   SAP5  291
            CALL  ERROR(N6)                                             SAP5  292
            CALL  WTCG(A(N1),A(N2),A(N3),A(N4),A(N5),NUMNP,NEQB,NEQ,NBLOCK,NSB SAP5  293
           X)                                                           SAP5  294
         18 CONTINUE                                                    SAP5  295
            CALL  SECOND(T(5))                                          SAP5  296
      C                                                                 CAM     9
      C     IF NTYP.NE.0  GO TO ROUTINES TO FORM EIGENVECTORS           CAM    10
      C                                                                 CAM    11
            IF(NTYP.EQ.0) GO TO 100                                     CAM    12
      C     DETERMINE SIZE REQUIREMENYS                                 CAM    13
            NST=NEQ                                                     CAM    14
            MBW=MBAND                                                   CAM    15
            NS=NST-NRDOF                                                CAM    16
            NS1=(NRDOF*NRDOF+NRDOF)/2                                   CAM    17
            IF(NRDOF.GT.MBW) GO TO 101                                  CAM    18
            NS2=(MBW-NRDOF)*NRDOF+((NRDOF-1)*(NRDOF-1)+(NRDOF-1))/2     CAM    19
            GO TO 102                                                   CAM    20
        101 NS2=(MBW*MBW+MBW)/2                                         CAM    21
        102 NOT=NS*NRDOF                                                CAM    22
            NADD=0                                                      CAM    23
            MODE=NF                                                     CAM    24
            MD=MODE+NADD                                                CAM    25
            IF(MD.GT.NRDOF) MD=NRDOF                                    CAM    26
            IF(MODE.GT.NRDOF) MODE=NRDOF                                CAM    27
            IF(NF.GT.NRDOF) NF=NRDOF                                    CAM    28
            NR2=1+NEQB*MBAND                                            CAM    29
            CALL SIZE(A(1),A(NR2),MTOT,NEQB,NBLOCK,MBAND)               CAM    30
            NR1=1                                                       CAM    31
            NR2=NR1+NS1                                                 CAM    32
            NR3=NR2+NS2                                                 CAM    33
            NR4=NR3+NSEQ*MBS                                            CAM    34
            NR5=NR4+NEQB*MBAND                                          CAM    35
            CALL STIFF(A(NR1),A(NR2),A(NR3),A(NR4),A(NR5),NBLOCK,NEQB,NRDOF, CAM    36
           1  NTYP,NST,MBW,NS,NS1,NS2,NS3,NOT,MBS,MODE,MD,MBLRED,NSEQ,NTO) CAM    37
            NSZ=NST                                                     CAM    38
            CALL MASS(A(NR3),NEQB,NBLOCK,NSZ)                           CAM    39
      C     INVERT KSS AND STORE ON TAPE 17                             CAM    40
            NR4=NR3+NST                                                 CAM    41
            NR5=NR4+NSEQ*MBS                                            CAM    42
            NR6=NR5+NSEQ*MBS                                            CAM    43
            NR7=NR6+NST                                                 CAM    44
            CALL GAUS(A(NR4),A(NR5),A(NR6),A(NR7),NSEQ,NS,MBS,MBLRED,   CAM    45
           1 18,18,17,1)                                                CAM    46
      C     FORM K* AND M*                                              CAM    47
            NR5=NR4+NS1                                                 CAM    48
            NR6=NR5+NS                                                  CAM    49
            NR7=NR6+NRDOF                                               CAM    50
            CALL REDUCE (A(NR1),A(NR2),A(NR3),A(NR4),A(NR5),A(NR6),A(NR7), CAM    51
```

POOR ORIGINAL



```
      1          NRDOF,NTYP,NST,MBW,NS,NS1,NS2,NS3,NOT,MB6,MODE,              CAM    52
      2 MD,MBLRED,NSEQ,NTO)                                                   CAM    53
C     SAVE MASS ON 15 FOR NORMALIZATION OF EIGENVECTORS                       CAM    54
      REWIND 15                                                               CAM    55
      DO 108 I=1,NST                                                          CAM    56
  108 WRITE(15) (A(NR3+I-1))                                                  CAM    57
      REWIND 15                                                               CAM    58
      DO 104 I=1,NS1                                                          CAM    59
  104 A(NR2+I-1)=A(NR4+I-1)                                                   CAM    60
      NR1=1                                                                   CAM    61
      NR2=NR1+NS1                                                             CAM    62
      NR3=NR2+NS1                                                             CAM    63
      NR4=NR3+NRDOF*MD                                                        CAM    64
      NR5=NR4+NRDOF*NRDOF                                                     CAM    65
      NR6=NR5+NRDOF                                                           CAM    66
      NR7=NR6+NRDOF                                                           CAM    67
      CALL POWER(A(NR1),A(NR2),A(NR3),A(NR4),A(NR5),A(NR6),A(NR7),            CAM    68
     1 NS1,NRDOF,MODE)                                                        CAM    69
      NR4=NR3+NRDOF*MD                                                        CAM    70
      NR5=NR4+NTO*NRDOF                                                       CAM    71
      NR6=NR5+NRDOF                                                           CAM    72
      NR7=NR6+NEQB*MODE                                                       CAM    73
      CALL EXPAND(A(NR3),A(NR4),A(NR5),A(NR6),NRDOF,MD,MODE,NTO,NEQB,         CAM    74
     1  NBLOCK,NTYP,NS,A(NR7),NST)                                            CAM    75
  100 CONTINUE                                                                CAM    76
C                                                                             SAP5   297
C     S O L U T I O N   P H A S E                                             SAP5   298
C                                                                             SAP5   299
      IF (GEOST) GO TO 30                                                     SAP5   300
   20 GO TO (30,40,50,60,70,80), KDYN                                         SAP5   301
C                                                                             SAP5   302
C     STATIC SOLUTION                                                         SAP5   303
C                                                                             SAP5   304
   30 IF(MODEX.EQ.0) GO TO 32                                                 SAP5   305
      T(6)=T(5)                                                               SAP5   306
C     S(6)=S(5)                                                               SAP5   307
      T(7)=T(5)                                                               SAP5   308
C     S(7)=S(5)                                                               SAP5   309
      IF (IPLT .GT. 0.AND..NOT.GEOST) CALL  SAPLOT(LL,NF,KDYN,NEQB,NEQ        SAP5   310
     X,NBLOCK)                                                                SAP5   311
      CALL  SECOND(T(8))                                                      SAP5   312
      DO 31 I=9,12                                                            SAP5   313
   31 T(I) = T(8)                                                             SAP5   314
C     S(I)=S(8)                                                               SAP5   315
      GO TO 90                                                                SAP5   316
C                                                                             SAP5   317
   32 REWIND 12                                                               SAP5   318
      CALL  SOLEQ                                                             SAP5   319
      CALL  SECOND(T(6))                                                      SAP5   320
      T(7)=T(6)                                                               SAP5   321
C     S(7)=S(6)                                                               SAP5   322
      IF (IPLT .GT. 0) CALL  SAPLOT(LL,NF,KDYN,NEQB,NEQ,NBLOCK)               SAP5   323
      CALL  SECOND(T(8))                                                      SAP5   324
      DO 33 I=9,12                                                            SAP5   325
   33 T(I) = T(8)                                                             SAP5   326
C     S(I)=S(8)                                                               SAP5   327
      IF (.NOT. GEOST) GO TO 90                                               SAP5   328
```

POOR ORIGINAL



```
      WRITE (6,2000) NUMEL,NELGEO
      CALL  ADDGEO (A(N1),A(NN2),NE2B,MBAND,NUMEL,NBLOCK,LL,ANORM)
      GO TO 20
C
C     EIGENVALUE EXTRACTION
C
   40 T(6) = T(5)
C     S(6)=S(5)
      CALL  SOLEIG
      CALL  SECOND(T(7))
      IF (IPLT .GT. 0) CALL  SAPLOT(LL,NF,KDYN,NEQB,NEQ,NBLOCK)
      CALL  SECOND(T(8))
      T(9) = T(8)
      T(10)= T(8)
      T(11)=T(8)
      T(12)=T(8)
C     S(9) = S(8)
C     S(10)= S(8)
C     S(11)= S(8)
C     S(12)= S(8)
      GO TO 90
C
C     FORCED DYNAMIC RESPONSE ANALYSIS
C
   50 T(6) = T(5)
C     S(6)=S(5)
      IF(NDYN.LT.0) GO TO 52
      CALL  SOLEIG
      CALL  SECOND(T(7))
      CALL  SECOND(T(7))
      IF (IPLT .GT. 0) CALL  SAPLOT(LL,NF,KDYN,NEQB,NEQ,NBLOCK)
      CALL  SECOND(T(8))
      GO TO 54
   52 DO 53 I=1,7
   53 T(I+1)=T(I)
C     S(I+1)=S(I)
      REWIND 2
      READ (2) NEQ,NBLOCK,NEQB,MBAND,NI,NF,(QQQ(I),I=1,NF)
      REWIND 7
      IMAX=NEQB*NF
      READ (7) (A(I),I=1,NF)
      DO 56 L=1,NBLOCK
   56 READ (7) (A(I),I=1,IMAX)
   54 CALL  HISTRY
      CALL  SECOND(T(9))
      T(10)= T(9)
      T(11)= T(9)
      T(12)= T(9)
C     S(10)= S(9)
C     S(11)= S(9)
C     S(12)= S(9)
      GO TO 90
C
C     RESPONSE SPECTRUM ANALYSIS
C
   60 T(6) = T(5)
C     S(6) = S(5)
```

POOR ORIGINAL



```
          IF(NTYP.NE.0) GO TO 105                                    CAM    77
          IF(NDYN.LT.0) GO TO 82                                     SAP5   385
          CALL  SOLEIG                                               SAP5   38
      105 CONTINUE                                                   CAM    78
          CALL  SECOND(T(7))                                         SAP5   388
          IF (IPLT .GT. 0) CALL  SAPLOT(LL,NF,KDYN,NEQB,NEQ,NBLOCK)  SAP5   389
          CALL  SECOND(T(8))                                         SAP5   390
          T(9)=T(8)                                                  SAP5   391
C         S(9) = S(8)                                                SAP5   392
          GO TO 64                                                   SAP5   393
       62 DO 63 I=1,8                                                SAP5   394
       63 T(I+1)=T(I)                                                SAP5   395
C         S(I+1)=S(I)                                                SAP5   396
          REWIND 2                                                   SAP5   397
          READ (2) NEQ,NBLOCK,NEQB,MBAND,NI,NF                       SAP5   398
          REWIND 7                                                   SAP5   399
          IMAX=NEQB*NF                                               SAP5   400
          READ (7) (A(I),I=1,NF)                                     SAP5   401
          DO 66 L=1,NBLOCK                                           SAP5   402
       66 READ (7) (A(I),I=1,IMAX)                                   SAP5   404
       64 IF(NRSC.LE.0)NRSC=1                                        SAP5   405
          DO 69 KAPG=1,NRSC                                          SAP5   406
       69 CALL RESPEC                                                SAP5   407
          CALL  SECOND(T(10))                                        SAP5   408
          T(11)=T(10)                                                SAP5   409
          T(12)=T(10)                                                SAP5   410
C         S(11)= S(10)                                               SAP5   411
C         S(12)= S(10)                                               SAP5   412
          GO TO 90                                                   SAP5   413
C                                                                    SAP5   414
C         STEP-BY-STEP (DIRECT INTEGRATION) ANALYSIS                 SAP5   415
C                                                                    SAP5   416
       70 DO 71 I=6,10                                               SAP5   417
       71 T(I) = T(5)                                                SAP5   418
C         S(I) = S(5)                                                SAP5   419
          CALL  STEP                                                 SAP5   420
          CALL  SECOND(T(11))                                        SAP5   421
          T(12)=T(10)                                                SAP5   422
C         S(12)=S(10)                                                SAP5   423
          GO TO 90                                                   SAP5   424
C                                                                    SAP5   425
C         FREQUENCY RESPONSE ANALYSIS                                SAP5   426
C                                                                    SAP5   427
       80 T(6)=T(5)                                                  SAP5   428
C         S(6)=S(5)                                                  SAP5   429
          IF (NDYN .LT. 0) GO TO 82                                  SAP5   430
          CALL  SOLEIG                                               SAP5   431
          CALL  SECOND(T(7))                                         SAP5   432
          IF (IPLT .GT. 0) CALL  SAPLOT(LL,NF,KDYN,NEQB,NEQ,NBLOCK)  SAP5   433
          CALL  SECOND(T(8))                                         SAP5   434
          T(9)=T(8)                                                  SAP5   435
          T(10)=T(8)                                                 SAP5   436
          T(11)=T(8)                                                 SAP5   437
C         S(9)=S(8)                                                  SAP5   438
C         S(10)=S(8)                                                 SAP5   439
C         S(11)=S(8)                                                 SAP5   440
          GO TO 88                                                   SAP5   441
```

# POOR ORIGINAL



```
      82 DO 83 I=1,10                                                       SAP5   442
      83 T(I+1)=T(I)                                                        SAP5   443
C        S(I+1)=S(I)                                                        SAP5   444
         REWIND 2                                                           SAP5   445
         READ (2) NEQ,NBLOCK,NEQB,MBAND,NI,NF                               SAP5   446
         REWIND 7                                                           SAP5   447
         IMAX=NEQB*NF                                                       SAP5   448
         READ (7) (A(I),I=1,NF)                                             SAP5   449
         DO 86 L=1,NBLOCK                                                   SAP5   450
      86 READ (7) (A(I),I=1,IMAX)                                           SAP5   451
      88 CALL  FREQ                                                         SAP5   452
         CALL  SECOND(T(12))                                                SAP5   453
C                                                                           SAP5   454
C        COMPUTE AND PRINT OVERALL TIME LOG                                 SAP5   455
C                                                                           SAP5   456
      90 TT = 0.0                                                           SAP5   457
C        ST = 0.000                                                         SAP5   458
         GO TO 5                                                            CORR2    6
         DO 95 I=1,11                                                       SAP5   459
         T(I) = T(I+1)-T(I)                                                 SAP5   460
         TT = TT + T(I)                                                     SAP5   461
      95 CONTINUE                                                           SAP5   462
C        S(I) = S(I+1)-S(I)                                                 SAP5   463
C        ST = ST + S(I)                                                     SAP5   464
C                                                                           SAP5   465
C        WRITE (6,203) (T(K),S(K),K=1,11),TT,ST                             SAP5   466
         WRITE (6,203) (T(K),K=1,11),TT                                     SAP5   467
C                                                                           SAP5   468
         GO TO 5                                                            SAP5   469
C                                                                           SAP5   470
  201 FORMAT (38H1E Q U A T I O N   P A R A M E T E R S, //                 SAP5   471
     X         34H  TOTAL NUMBER OF EQUATIONS        =,I5,                  SAP5   472
     X        /34H  BANDWIDTH                        =,I5,                  SAP5   473
     X        /34H  NUMBER OF EQUATIONS IN A BLOCK   =,I5,                  SAP5   474
     X        /34H  NUMBER OF BLOCKS                 =,I5)                  SAP5   475
  203 FORMAT (1H1,31H0 V E R A L L   T I M E   L O G, //                    SAP5   476
     X 5X,30HNODAL POINT INPUT              =,  F8.2 /                      SAP5   477
     X 5X,30HELEMENT STIFFNESS FORMATION    =,  F8.2 /                      SAP5   478
     X 5X,30HNODAL LOAD INPUT               =,  F8.2 /                      SAP5   479
     X 5X,30HTOTAL STIFFNESS FORMATION      =,  F8.2 /                      SAP5   480
     X 5X,30HSTATIC ANALYSIS                =,  F8.2 /                      SAP5   481
     X 5X,30HEIGENVALUE EXTRACTION          =,  F8.2 /                      SAP5   482
     X 5X,30HSTRUCTURE PLOTTING             =,  F8.2 /                      SAP5   483
     X 5X,30HFORCED RESPONSE ANALYSIS       =,  F8.2 /                      SAP5   484
     X 5X,30HRESPONSE SPECTRUM ANALYSIS     =,  F8.2 /                      SAP5   485
     X 5X,30HSTEP-BY-STEP INTEGRATION       =,  F8.2 /                      SAP5   486
     X 5X,30HFREQUENCY RESPONSE ANALYSIS    =,  F8.2 //                     SAP5   487
     X 5X,30HTOTAL SOLUTION TIME            =,  F8.2 /)                     SAP5   488
C                                                                           SAP5   489
  320 FORMAT (// 47H ** WARNING.  ESTIMATE OF STORAGE FOR A DYNAMIC,        SAP5   490
     X                32H ANALYSIS EXCEEDS AVAILABLE CORE,                  SAP5   491
     X                /,* REQUIRED MTOT=*,I10)                              SAP5   492
C                                                                           SAP5   493
C                                                                           SAP5   494
 1001 FORMAT (14I5)                                                         SAP5   495
 2000 FORMAT (*0TOTAL NUMBER OF ELEMENTS                        =*,I5,/,    SAP5   496
     X          * NUMBER OF ELEMENTS WITH GEOMETRIC STIFFNESS=*,I5)         SAP5   497
 2010 FORMAT (*0*** WARNING *** THE AVAILABLE BLANK COMMONT (MTOT) IS*,     SAP5   498
     X* TOO SMALL*,/,* COMPUTED NO. OF EQUATIONS PER BLOCK=*,I10,/,         SAP5   499
     X* PROCEED IN THE DATA CHECK MODE WITH NEQB=2*)                        SAP5   500
 2020 FORMAT(*0*** FATAL ERROR *** AVAILABLE CORE IS TOO SMALL*             SAP5   501
     1,* FOR PLOTTING. REQUIRED MTOT=*,I10/                                 SAP5   502
     2*  CONTINUE IN THE DATA CHECK MODE*)                                  SAP5   503
 2030 FORMAT(*0*** FATAL ERROR ***  AVAILABLE CORE IS INADAQUATE*           SAP5   504
     1,*  REQUIRED MTOT=*,I10/*CONTINUE IN THE DATA CHECK MODE*)            CORR2    7
         END                                                                SAP5   507
```

```
      SUBROUTINE REDUCI (ID,IDN,NTYP,NUMNP,NRDOF)                       REDUCI    2
      DIMENSION MASN(200),MADF(200),ID(NUMNP,6),IDN(NUMNP,6)            CAM      79
      LEVEL 2,ID,IDN                                                    REDUCI    4
C                                                                       REDUCI    5
C     INPUT PRIMARY DOF AND REORDER ALL DOF SO THAT PRIMARY DOF ARE FIRS REDUCI   6
C        NTYP = 0  NO REDUCTION                                         REDUCI    7
C               1 EXACT REDUCTION                                       REDUCI    8
C               2 GUYAN REDUCTION                                       REDUCI    9
C               3 MODIFIED GUYAN                                        REDUCI   10
C      M'.4(I) = NODE NUMBER OF I PRIMARY DOF , I=1,NRDOF               REDUCI   11
C      MADF(I) = DOF NUMBER  (1 TO 6)                                   REDUCI   12
C      NRDOF = NUMBER OF PRIMARY DOF < 200                              CAM      80
      READ(5,300) NTYP                                                  REDUCI   14
  300 FORMAT(14I5)                                                      REDUCI   15
      IF(NTYP.GT.1) GO TO 301                                           REDUCI   16
      WRITE(6,302)                                                      REDUCI   17
  302 FORMAT(5X,19HNO REDUCTION OF DOF)                                 REDUCI   18
      RETURN                                                            REDUCI   19
  301 IF(NTYP.EQ.2) WRITE(6,303)                                        REDUCI   20
      IF(NTYP.EQ.3) WRITE(6,304)                                        REDUCI   21
  303 FORMAT(5X,15HGUYAN REDUCTION)                                     REDUCI   22
  304 FORMAT(5X,24HMODIFIED GUYAN REDUCTION)                            REDUCI   23
      READ(5,300) NRDOF,(MASN(I),MADF(I),I=1,NRDOF)                     REDUCI   24
      WRITE(6,305)                                                      REDUCI   25
  305 FORMAT(20X,11HPRIMARY DOF/10X,4HNODE,6X,3HDOF)                    REDUCI   26
      WRITE(6,306) (MASN(I),MADF(I),I=1,NRDOF)                          REDUCI   27
  306 FORMAT(9X,I5,4X,I5)                                               REDUCI   28
C                                                                       REDUCI   29
C     READ ID FROM TAPE 8 AND WRITE NEW ID ON TAPE 8                    REDUCI   30
C                                                                       REDUCI   31
      REWIND 8                                                          REDUCI    3
      READ(8) ID                                                        REDUCI   33
      IP=1                                                              REDUCI   34
      IS=NRDOF+1                                                        REDUCI   35
C*****                                                                  REDUCI   36
      WRITE(6,307)                                                      REDUCI   37
  307 FORMAT(5X,13HORIGINAL ID =)                                       REDUCI   38
      DO 310 I=1,NUMNP                                                  REDUCI   39
      WRITE(6,311) (ID(I,J),J=1,6)                                      REDUCI   40
  311 FORMAT(5X,6(2X,I5))                                               REDUCI   41
      DO 312 J=1,6                                                      CAM      81
      IF(ID(I,J).EQ.0) GO TO 312                                        CAM      82
      NODE=MASN(IP)                                                     CAM      83
      NDF=MADF(IP)                                                      CAM      84
      IF(I.EQ.NODE.AND.J.EQ.NDF) GO TO 313                              CAM      85
C     SECONDARY DOF                                                     CAM      86
      IDN(I,J)=IS                                                       CAM      87
      IS=IS+1                                                           CAM      88
      GO TO 312                                                         CAM      89
C     PRIMARY DOF                                                       CAM      90
  313 IDN(I,J)=IP                                                       CAM      91
      IP=IP+1                                                           CAM      92
  312 CONTINUE                                                          CAM      93
  310 CONTINUE                                                          REDUCI   58
C                                                                       REDUCI   59
C     WRITE ONTO TAPE 8 AND PRINT NEW ID                                REDUCI   60
C                                                                       REDUCI   61
      REWIND 8                                                          REDUCI   62
      WRITE(8) IDN                                                      REDUCI   63
      WRITE(6,320)                                                      REDUCI   64
  320 FORMAT(5X,8HNEW ID =)                                             REDUCI   65
      DO 321 I=1,NUMNP                                                  REDUCI   66
      WRITE(6,311) (IDN(I,J),J=1,6)                                     REDUCI   67
      DO 321 J=1,6                                                      REDUCI   68
      ID(I,J)=IDN(I,J)                                                  REDUCI   69
  321 CONTINUE                                                          REDUCI   70
      REWIND 8                                                          REDUCI   71
      RETURN                                                            REDUCI   72
      END                                                               REDUCI   73
```

POOR ORIGINAL



```
      SUBROUTINE SIZE(X,XL,MTOT,NEQB,NBLOCK,MBAND)                      CAM      94
      COMMON/CONTL/NRDOF,NTYP,NST,MBW,NS,NS1,NS2,NS3,NOT,MBS,MODE,      SIZE      3
     1 MD,MBLRED,NSEQ,NTO                                               SIZE      4
      DIMENSION X(NEQB,MBAND),XL(NEQB)                                  CAM      95
      LEVEL 2,X,XL                                                      CAM      96
C     FIND BANDWIDTH OF KSS                                             SIZE      7
      MBS=0                                                             SIZE      8
      REWIND 4                                                          SIZE      9
      NPP=NRDOF/NEQB                                                    SIZE     10
      IROW=0                                                            SIZE     11
      IF(NPP.EQ.0) GO TO 1                                              SIZE     12
      DO 2 I=1,NPP                                                      SIZE     13
      IROW=IROW+NEQB                                                    SIZE     14
      READ(4) ((X(IC,JC),IC=1,NEQB),JC=1,MBAND),(XL(IL),IL=1,NEQB)      CAM      97
    2 CONTINUE                                                          SIZE     16
    1 NF=NRDOF-IROW+1                                                   SIZE     17
      NSEC=0                                                            SIZE     18
    3 READ(4) ((X(IC,JC),IC=1,NEQB),JC=1,MBAND),(XL(IL),IL=1,NEQB)      CAM      98
      NSEC=NSEC+NEQB-NF+1                                               SIZE     20
      NLST=NEQB                                                         CAM      99
      IF(NSEC.GT.NS) NLST=NEQB-(NSEC-NS)                                CAM     100
      IF(NSEC.GT.NS) NSEC=NS                                            SIZE     21
      DO 4 I=NF,NLST                                                    CAM     101
      MTST=0                                                            CAM     102
      DO 7 J=1,MBAND                                                    CAM     103
    7 IF(X(I,J).NE.0.) MTST=J                                           CAM     104
      IF(MTST.GT.MBS) MBS=MTST                                          CAM     105
    4 CONTINUE                                                          CAM     106
      NF=1                                                              SIZE     25
      IF(NSEC.NE.NS) GO TO 3                                            SIZE     26
      REWIND 4                                                          SIZE     27
      NS3=NS*MBS                                                        SIZE     28
      WRITE(6,100) NST,NRDOF,NS,NS1,NS2,NS3,NOT,MBS                     SIZE     29
  100 FORMAT(//5X,26HSUMMARY OF SIZE PARAMETERS/                        SIZE     30
     1          5X,34HNUMBER OF UNRESTRAINED DOF (NST) =,I5/            SIZE     31
     2          5X,31HNUMBER OF PRIMARY DOF (NRDOF) =,I5/               SIZE     32
     3          5X,30HNUMBER OF SECONDARY DOF (NS) =,I5/                SIZE     33
     4          5X,19HSIZE OF KPP (NS1) =,I5/                           SIZE     34
     5          5X,19HSIZE OF KPS (NS2) =,I5/                           SIZE     35
     6          5X,19HSIZE OF KSS (NS3) =,I5/                           SIZE     36
     7          5X,18HSIZE OF TO (NOT) =,I5/                            SIZE     37
     8          5X,18HBANDWIDTH OF KSS =,I5)                            SIZE     38
C     DETERMINE BAND SIZE  OR KSS                                       SIZE     39
C     FOR STIFF                                                         SIZE     40
      NSEQ=(MTOT-NS1-NS2-NEQB*MBW)/MBS                                  SIZE     41
C     FOR KSS-1                                                         SIZE     42
      NDUM=(MTOT-NS1-NS2-3*NST)/(2*MBS)                                 SIZE     43
      IF(NDUM.LT.NSEQ) NSEQ=NDUM                                        SIZE     44
      IF(NSEQ.GE.NS) GO TO 5                                            SIZE     45
      MBLRED=NS/NSEQ+1                                                  SIZE     46
      GO TO 6                                                           SIZE     47
    5 NSEQ=NS                                                           SIZE     48
      MBLRED=1                                                          SIZE     49
    6 NTO=(MTOT-2*NS1-NS2-NS-NST-NRDOF)/NRDOF                           SIZE     50
      IF(NTO.GT.NS) NTO=NS                                              SIZE     51
      WRITE(6,101) MBLRED,NSEQ,NTO                                      SIZE     52
  101 FORMAT(//5X,21HNO OF BLOCKS IN KSS =,I5/                          SIZE     53
     1          5X,24HNO OF EQUATIONS IN KSS =,I5/                      SIZE     54
     2          5X,23HNO OF EQUATIONS IN TO =,I5)                       SIZE     55
      NA4=2*NS1+NRDOF*(MD+NRDOF+1+MD)                                   SIZE     56
      WRITE(6,102) NA4                                                  SIZE     57
  102 FORMAT(5X,27HSIZE OF A DURING EIGENSOL =,I5)                      SIZE     58
      RETURN                                                            SIZE     59
      END                                                               SIZE     60
```

```
      SUBROUTINE STIFF(XKPP,XKPS,XKSS,X,XL,NBLOCK,NEQB,NRDOF,NTYP,NST,  CAM      107
     1 MBW,NS,NS1,NS2,NS3,NOT,MBS,MODE,MD,MBLRED,NSEQ,NTO)             STIFF      3
      DIMENSION XKPP(NS1),XKPS(NS2),XKSS(NSEQ,MBS),X(NEQB,MBW),XL(NEQB) CAM      108
      LEVEL 2,XKPP,XKPS,XKSS,X,XL                                       CAM      109
C                                                                       STIFF      6
C     FORM KPP KPS,KSS FROM TOTAL STIFFNESS ON TAPE 4 WRITE KSS ON      STIFF      7
C     TAPE 16 IN BLOCKS                                                 STIFF      8
C                                                                       STIFF      9
      REWIND 16                                                         STIFF     10
      ILPP=0                                                            STIFF     11
      ILPS=0                                                            STIFF     12
      IRS=0                                                             STIFF     13
      IBS=0                                                             STIFF     14
      IROW=0                                                            STIFF     15
      DO 1 IR=1,NBLOCK                                                  STIFF     16
      READ(4) ((X(IC,JC),IC=1,NEQB),JC=1,MBW),(XL(IL),IL=1,NEQB)        CAM      110
      DO 2 I=1,NEQB                                                     STIFF     18
      IROW=IROW+1                                                       STIFF     19
      IF(IROW.GT.NST) GO TO 2                                           STIFF     20
      IF(IROW.GT.NRDOF) GO TO 3                                         STIFF     21
C                                                                       STIFF     22
C     KPP OR KPS                                                        STIFF     23
      DO 4 J=1,MBW                                                      STIFF     24
      IF(J+IROW-1.GT.NRDOF) GO TO 5                                     STIFF     25
C     KPP                                                               STIFF     26
      ILPP=ILPP+1                                                       STIFF     27
      XKPP(ILPP)=X(I,J)                                                 STIFF     28
      GO TO 4                                                           STIFF     29
C     KPS                                                               STIFF     30
    5 ILPS=ILPS+1                                                       STIFF     31
      XKPS(ILPS)=X(I,J)                                                 STIFF     32
    4 CONTINUE                                                          STIFF     33
      GO TO 2                                                           STIFF     34
C     KSS                                                               STIFF     35
    3 IRS=IRS+1                                                         STIFF     36
      DO 6 J=1,MBS                                                      STIFF     37
    6 XKSS(IRS,J)=X(I,J)                                                STIFF     38
      IF(IRS.LT.NSEQ) GO TO 2                                           STIFF     39
      WRITE(16) XKSS                                                    STIFF     40
      IRS=0                                                             STIFF     41
    2 CONTINUE                                                          STIFF     42
    1 CONTINUE                                                          STIFF     44
      REWIND 16                                                         STIFF     45
      RETURN                                                            STIFF     86
      END                                                               STIFF     87
```

POOR ORIGINAL

SUBROUTINE MASS 76/76 OPT=1 FTN 4.5+414 11/19/80 17.24.07

```
      SUBROUTINE MASS(XM,NEQB,NBLOCK,NST)                               MASS       2
      DIMENSION XM(NST)                                                 MASS       3
      LEVEL 2,XM                                                        MASS       4
C                                                                       MASS       5
C     LOAD MASS VECTOR                                                  MASS       6
C                                                                       MASS       7
      REWIND 9                                                          MASS       8
       IROW=1                                                           CAM      111
      DO 1 I=1,NBLOCK                                                   CAM      112
      JLST=IROW+NEQB-1                                                  CAM      113
      IF(JLST.GT.NST) JLST=NST                                          CAM      114
      READ(9) (XM(J),J=IROW,JLST)                                       CAM      115
      IROW=JLST+1                                                       CAM      116
    1 CONTINUE                                                          CAM      117
      REWIND 9                                                          MASS      10
      RETURN                                                            MASS      17
      END                                                               MASS      18
```

POOR ORIGINAL



```
      SUBROUTINE GAUS(A1,A2,B,G,NSB,NST,MBS, NBLK,NT1,NT2,NT3,INDEX)     GAUS       2
      DIMENSION A1(NSB,MBS),A2(NSB,MBS),B(NST),G(NST)                    GAUS       3
      LEVEL 2,A1,A2,B,G                                                  GAUS       4
C                                                                        GAUS       5
C  INDEX = 1 FIND INVERSE OF A (ON TAPE NT1) STOTE RESULTS ON TAPE NT3   GAUS       6
C         2  FACTOR A STORE RESULTS ON TAPE NT2                          GAUS       7
C         3 SOLVE A X = B  ENTER WITH 2 FIRST TO FACTOR A                GAUS       8
      IF (INDEX.EQ.3) GO TO 1                                            GAUS       9
C                                                                        GAUS      10
C     FACTOR A                                                           GAUS      11
C                                                                        GAUS      12
      REWIND NT1                                                         GAUS      13
      REWIND NT2                                                         GAUS      14
      N1=MBS-1                                                           GAUS      15
      N2=NSB-MBS+1                                                       GAUS      16
      IBMX=NBLK-1                                                        GAUS      17
      IF(NBLK.EQ.1) IBMX=1                                               GAUS      18
      DO 10 IB=1,IBMX                                                    GAUS      19
      IF(IB.EQ.1) READ(NT1) A1                                           GAUS      20
      IF(NBLK.NE.1) READ(NT1) A2                                         GAUS      21
      JSTRT=2                                                            GAUS      22
      IF(IB.GT.1) JSTRT=JLST+1                                           GAUS      23
      JLST=2.*NSB                                                        GAUS      24
      IF(IB.GT.1) JLST=JSTRT+N1+N2-1                                     GAUS      25
      IF(JLST.GT.NST) JLST=NST                                           GAUS      26
      DO 11 J=JSTRT,JLST                                                 GAUS      27
      MSK=J-MBS                                                          GAUS      28
      IF(MSK.LT.1) MSK=1                                                 GAUS      29
      CALL LOCATE(MSK,J,A,A1,A2,MBS,NSB,IB)                              GAUS      30
      G(MSK)=A                                                           GAUS      31
      IF=MSK+1                                                           GAUS      32
      IL=J-1                                                             GAUS      33
      DO 12 I=IF,IL                                                      GAUS      34
      CALL LOCATE(I,J,A A1,A2,MBS,NSB,IB)                                GAUS      35
      G(I)=A                                                             GAUS      36
      KF=MSK                                                             GAUS      37
      IF(I-MBS.GT.KF) KF=I-MBS                                           GAUS      38
      KL=I-1                                                             GAUS      39
      IF(KF.GT.KL) GO TO 12                                              GAUS      40
      DO 13 K=KF,KL                                                      GAUS      41
      CALL LOCATE(K,I,A,A1,A2,MBS,NSB,IB)                                GAUS      42
   13 G(I)=G(I)-A*G(K)                                                   GAUS      43
   12 CONTINUE                                                           GAUS      44
      IF=J-MBS                                                           GAUS      45
      IF(IF.LT.1) IF=1                                                   GAUS      46
      IL=J-1                                                             GAUS      47
      DO 14 I=IF,IL                                                      GAUS      48
      KA=I-(IB-1)*NSB                                                    GAUS      49
      IF(KA.GT.NSB) GO TO 15                                             GAUS      50
      LA=J-(IB-1)*NSB-KA+1                                               GAUS      51
      IF(LA.GT.MBS) GO TO 14                                             CAM      118
      A1(KA,LA)=G(I)/A1(KA,1)                                            GAUS      52
      GO TO 14                                                           GAUS      53
   15 KA=I-IB*NSB                                                        GAUS      54
      LA=J-IB*NSB-KA+1                                                   GAUS      55
      IF(LA.GT.MBS) GO TO 14                                             CAM      119
      A2(KA,LA)=G(I)/A2(KA,1)                                            GAUS      56
```

```
   14 CONTINUE
      KF=J-MBS
      IF(KF.LT.1) KF=1
      KL=J-1
      KA=J-(IB-1)*NSB
      DO 16 K=KF,KL
      CALL LOCATE(K,J,A,A1,A2,MBS,NSB,IB)
      IF(KA.GT.NSB) GO TO 17
      A1(KA,1)=A1(KA,1)-A*G(K)
      GO TO 16
   17 KA=J-IB*NSB
      A2(KA,1)=A2(KA,1)-A*G(K)
   16 CONTINUE
   11 CONTINUE
      WRITE(NT2) A1
      IF(NBLK.EQ.1) GO TO 10
      DO 18 IT=1,NSB
      DO 18 JT=1,MBS
   18 A1(IT,JT)=A2(IT,JT)
   10 CONTINUE
      REWIND NT1
      REWIND NT2
      IF(INDEX.EQ.2) RETURN
      REWIND NT3
C   SET UP LOAD FOR INVERT STORE RESULTS ON TAPE NT3
      DO 20 ICOL=1,NST
      DO 21 J=1,NST
   21 B(J)=0.
      B(ICOL)=1.
      GO TO 1
   22 WRITE(NT3) B
   20 CONTINUE
      REWIND NT3
      RETURN
C
C     SOLVE A X = B HAVING FACTORED A ON TAPE NT2 RETURN SOLUTION IN B
C
C     REDUCE B
    1 G(1)=B(1)
      IBMX=NBLK-1
      IF(NBLK.EQ.1) IBMX=1
      DO 30 IB=1,IBMX
      IF(IB.EQ.1) READ(NT2) A1
      IF(NBLK.NE.1) READ(NT2) A2
      JSTRT=2
      IF(IB.GT.1) JSTRT=JLST+1
      JLST=2.*NSB
      IF(IB.GT.1) JLST=JSTRT+N1+N2-1
      IF(JLST.GT.NST) JLST=NST
      DO 31 I=JSTRT,JLST
      G(I)=B(I)
      KS=I-MBS
      IF(KS.LT.1) KS=1
      KL=I-1
      DO 32 K=KS,KL
      CALL LOCATE(K,I,A,A1,A2,MBS,NSB,IB)
   32 G(I)=G(I)-A*G(K)
```

```
   31 CONTINUE
      DO 33 IT=1,NSB
      DO 33 JT=1,MBS
   33 A1(IT,JT)=A2(IT,JT)
   30 CONTINUE
C     BACKSUBSTITUTE
      REWIND NT2
      IROW=0
      DO 40 IB=1,NBLK
      READ(NT2) A1
      IMX=NSB
      IF(IROW+IMX.GT.NST) IMX=NST
      IFST=IROW+1
      ILST=IROW+NSB
      DO 41 I=IFST,ILST
   41 G(I)=G(I)/A1(I-IROW,1)
      IROW=IROW+ILST-IFST+1
   40 CONTINUE
      REWIND NT2
      B(NST)=G(NST)
      IB=NBLK-1
      IF(IB.LT.1) IB=1
   47 IDUM=IB-1
      IF(IDUM.LE.0) GO TO 42
      DO 43 I=1,IDUM
   43 READ(NT2) A1
   42 READ(NT2) A1
      IF(NBLK.EQ.1) GO TO 44
      READ(NT2) A2
   44 ISTRT=(IB-1)*NSB+N1+1
      ILST=ISTRT+NI+2*N2
      IF(ILST.GT.NST) ILST=NST
      IF(IB.EQ.1) ISTRT=2
      NI=ILST-ISTRT+1
      DO 45 I=1,NI
      IROW=ILST-I+1
      KS=IROW-MBS
      IF(KS.LT.1) KS=1
      KL=IROW-1
      DO 46 K=KS,KL
      CALL LOCATE(K,IROW,A,A1,A2,MBS,NSB,IB)
   46 G(K)=G(K)-A*B(IROW)
      B(IROW-1)=G(IROW-1)
   45 CONTINUE
      IB=IB-1
      REWIND NT2
      IF(IB.GT.0) GO TO 47
      IF(INDEX.EQ.1) GO TO 22
      RETURN
      END
```

SUBROUTINE LOCATE     76/76    OPT=1          FTN 4.5+414          11/19/80    17.24.07

```
      SUBROUTINE LOCATE(I,J,A,A1,A2,MBS,NSB,IB)                       LOCATE     2
      DIMENSION A1(NSB,MBS),A2(NSB,MBS)                               LOCATE     3
      LEVEL 2,A1,A2                                                   LOCATE     4
      ILOW=(IB-1)*NSB                                                 LOCATE     5
      IHGH=IB*NSB                                                     LOCATE     6
      INDEX=3                                                         LOCATE     7
      IF(I.GT.ILOW.AND.I.LE.IHGH) INDEX=1                             LOCATE     8
      ISHG=(IB+1)*NSB                                                 LOCATE     9
      IF(I.GT.IHGH.AND.I.LE.ISHG) INDEX=2                             LOCATE    10
      GO TO (1,2,3),INDEX                                             LOCATE    11
    1 K=I-(IB-1)*NSB                                                  LOCATE    12
      L=J-(IB-1)*NSB-K+1                                              LOCATE    13
      IF(L.GT.MBS) GO TO 5                                            CAM      120
      A=A1(K,L)                                                       LOCATE    14
      RETURN                                                          LOCATE    15
    2 K=I-IB*NSB                                                      LOCATE    16
      L=J-IB*NSB-K+1                                                  LOCATE    17
      IF(L.GT.MBS) GO TO 5                                            CAM      121
      A=A2(K,L)                                                       LOCATE    18
      RETURN                                                          LOCATE    19
    5 A=0.                                                            CAM      122
      RETURN                                                          CAM      123
    3 WRITE(6,4) I,J,IB                                               LOCATE    20
    4 FORMAT(//5X,39HERROR IN LOCATE ELEMENT NOT IN A1 OR A2/        LOCATE    21
     1  10X,3HI =,I5/10X,3HJ =,I5/10X,4HIB =,I5)                      LOCATE    22
      STOP                                                            LOCATE    23
      END                                                             LOCATE    24
```

SUBROUTINE PPLOC     76/76    OPT=1          FTN 4.5+414          11/19/80    17.24.07

```
      SUBROUTINE PPLOC(IR,JC,ILOC)                                    PPLOC      2
      COMMON/CONTL/ NDOF,NTYP,NST,MBW,NS,NS1,NS2,NS3,NOT,MBS,MODE,   PPLOC      3
     1 MD,MBLRED,NSEQ,NTO                                             PPLOC      4
C     LOCATES I,J IN VECTOR STORING KPP                               PPLOC      5
      IDR=IR                                                          PPLOC      6
      JDC=JC                                                          PPLOC      7
      IF(IR.LT.JC) GO TO 3                                            PPLOC      8
      IDR=JC                                                          PPLOC      9
      JDC=IR                                                          PPLOC     10
    3 CONTINUE                                                        PPLOC     11
      NL=NDOF                                                         PPLOC     12
      ILOC=0                                                          PPLOC     13
      IS=IDR-1                                                        PPLOC     14
      IF(IS.EQ.0) GO TO 1                                             PPLOC     15
      DO 2 I=1,IS                                                     PPLOC     16
      ILOC=ILOC+NL                                                    PPLOC     17
    2 NL=NL-1                                                         PPLOC     18
    1 ILOC=ILOC+JDC-IDR+1                                             PPLOC     19
      RETURN                                                          PPLOC     20
      END                                                             PPLOC     21
```

POOR ORIGINAL



```
      SUBROUTINE REDUCE(XKPP,XKPS,XM,XMR,XSS,T1,TO,                     REDUCE    2
     1            NRDOF,NTYP,NST,MBW,NS,NS1,NS2,NS3,NOT,MBS,MODE,        REDUCE    3
     2 MD,MBLRED,NSEQ,NTO)                                              REDUCE    4
      DIMENSION XKPP(NS1),XKPS(NS2),XM(NST),XMR(NS1),XSS(NS),T1(NRDOF), REDUCE    5
     1 TO(NTO,NRDOF)                                                    REDUCE    6
      LEVEL 2,XKPP,XKPS,XM,XMR,XSS,T1,TO                                REDUCE    7
C                                                                       REDUCE    8
C      FORM TO = KSS-1 * KSP ONE BLOCK AT A TIME AND STORE ON 18        REDUCE    9
C                                                                       REDUCE   10
      NBL=NS/NTO+1                                                      REDUCE   11
      IF(NS.EQ.NTO) NBL=1                                               CAM     124
      REWIND 18                                                         REDUCE   12
      REWIND 17                                                         REDUCE   13
      DO 1 IB=1,NBL                                                     REDUCE   14
      DO 2 I=1,NTO                                                      REDUCE   15
      DO 2 J=1,NRDOF                                                    REDUCE   16
    2 TO(I,J)=0.                                                        REDUCE   17
      DO 3 I=1,NTO                                                      REDUCE   18
      IROW=(IB-1)*NTO+I                                                 REDUCE   19
      IF(IROW.GT.NS) GO TO 3                                            REDUCE   20
      READ(17) XSS                                                      REDUCE   21
      DO 5 J=1,NRDOF                                                    REDUCE   22
      DO 5 K=1,NS                                                       REDUCE   23
      CALL PSLOC(2,K,J,ILOC)                                            REDUCE   24
      IF(ILOC.EQ.0) GO TO 5                                             CAM     125
      TO(I,J)=TO(I,J)+XSS(K)*XKPS(ILOC)                                 CAM     126
    5 CONTINUE                                                          CAM     127
    3 CONTINUE                                                          REDUCE   26
    1 WRITE(18) TO                                                      REDUCE   27
      REWIND 17                                                         REDUCE   28
      REWIND 18                                                         REDUCE   29
C                                                                       REDUCE   30
C     FORM T1 = KSS-1 * MSS * TO ONE ROW AT A TIME AND SAVE ON 16       REDUCE   31
C                                                                       REDUCE   32
      REWIND 16                                                         REDUCE   33
      DO 10 I=1,NS                                                      REDUCE   34
      READ(17) XSS                                                      REDUCE   35
      DO 11 IM=1,NS                                                     REDUCE   36
   11 XSS(IM)=XSS(IM)*XM(IM+NRDOF)                                      CAM     128
      DO 12 J=1 NRDOF                                                   REDUCE   38
   12 T1(J)=0.                                                          REDUCE   39
      DO 13 IB=1,NBL                                                    REDUCE   40
      READ(18) TO                                                       REDUCE   41
      DO 14 J=1,NRDOF                                                   REDUCE   42
      DO 14 K=1,NTO                                                     REDUCE   43
      KROW=(IB-1)*NTO+K                                                 REDUCE   44
      IF(KROW.GT.NS) GO TO 14                                           REDUCE   45
      T1(J)=T1(J)+XSS(KROW)*TO(K,J)                                     REDUCE   46
   14 CONTINUE                                                          REDUCE   47
   13 CONTINUE                                                          REDUCE   48
      WRITE(16) T1                                                      REDUCE   49
      REWIND 18                                                         REDUCE   50
   10 CONTINUE                                                          REDUCE   51
      REWIND 16                                                         REDUCE   52
      REWIND 17                                                         CAM     129
C                                                                       REDUCE   53
C     COMPUTE K* = KPP   KPS * TO AND STORE IN KPP                      REDUCE   54
```

POOR ORIGINAL



```
      C
            DO 20 IB=1,NBL
            READ(18) TO
            DO 21 K=1,NRDOF
            DO 21 J=K,NRDOF
            CALL PPLOC(K,J,ILOC)
            DO 22 L=1,NTO
            LROW=(IB-1)*NTO+L
            IF(LROW.GT.NS) GO TO 22
            CALL PSLOC(I,K,LROW,INDEX)
            IF(INDEX.EQ.0) GO TO 22
            XKPP(ILOC)=XKPP(ILOC)-XKPS(INDEX)*TO(L,J)
         22 CONTINUE
         21 CONTINUE
         20 CONTINUE
            REWIND 18
      C
      C     COMPUTE M* = MPP + KPS * TI AND STORE IN XMR
      C
            DO 30 I=1,NSI
         30 XMR(I)=0.
            DO 31 I=1,NRDOF
            CALL PPLOC(I,I,ILOC)
         31 XMR(ILOC)=XM(I)
            DO 32 L=1,NS
            READ(16) TI
            DO 33 K=1,NRDOF
            DO 33 J=K,NRDOF
            CALL PSLOC(I,K,L,INDEX)
            IF (INDEX.EQ.0) GO TO 33
            CALL PPLOC(K,J,ILOC)
            XMR(ILOC)=XMR(ILOC)+XKPS(INDEX)*TI(J)
         33 CONTINUE
         32 CONTINUE
            REWIND 16
            RETURN
            END
```

```
SUBROUTINE PSLOC       76/76    OPT=1                          FTN 4.5+414          11/19/80  17.24.07

      SUBROUTINE PSLOC(IT,I,J,ILOC)
      COMMON/CONTL/ NDOF,NTYP,NST,MBW,NS,NS1,NS2,NS3,NCT,MBS,MODE,
     1 MD,MBLRED,NSEQ,NTO
C
C     LOCATE I,J COMPONENT OF KPS IN A IF IT=1 OR J,I COMPONENT
C     OF KSP IF IT=2 ...ILOC IS LOCATION OF COMPONENT IN A..IF
C     ILOC=0 COMPONENT IS 0
C
      ILOC=0
      IF(IT.EQ.2) GO TO 1
      IR=I
      JC=J
      GO TO 2
    1 IR=J
      JC=I
    2 IF(NDOF.GT.MBW) GO TO 10
      IF(JC.GT.MBW-NDOF+IR-1) RETURN
      IL=MBW-NDOF-1
      IF(IR.EQ.1) GO TO 4
      IN=IR-1
      DO 3 IS=1,IN
      IL=IL+1
    3 ILOC=ILOC+IL
    4 ILOC=ILOC+JC
      RETURN
C
   10 IF(IR.LT.NDOF-MBW+1) RETURN
      IF(JC.GT.IR-NDOF+MBW) RETURN
      KMIN=NDOF-MBW+1
      IF(IR.LE.KMIN) GO TO 11
      IN=IR-1
      IL=0
      DO 12 IS=KMIN,IN
      IL=IL+1
   12 ILOC=ILOC+IL
   11 ILOC=ILOC+JC
      RETURN
      END
```

POOR ORIGINAL

```
SUBROUTINE ORAMS       76/76    OPT=1                          FTN 4.5+414          11/19/80  17.24.07

      SUBROUTINE ORAMS(XM,EV,B,DUM,IM,NS1,NDOF,MODE)
      DIMENSION EV(NDOF,MODE),B(NDOF),DUM(NDOF),ALPH(20),XM(NS1)
      LEVEL 2,XM,EV,B,DUM
C
C     COMPUTE NEW B ORTHOGONAL TO ALL LOWER MODES
C     BNEW=BOLD - SUM ( (EV' *M * BOLD) * EV )
C
      M=IM-1
      DO 1 I=1,M
      DO 2 J=1,NDOF
    2 DUM(J)=0.
      DO 3 J=1,NDOF
      DO 3 K=1,NDOF
      CALL PPLOC(J,K,ILOC)
    3 DUM(J)=DUM(J)+XM(ILOC)*B(K)
      ALPH(I)=0.
      DO 4 J=1,NDOF
    4 ALPH(I)=ALPH(I)+EV(J,I)*DUM(J)
    1 CONTINUE
      DO 5 J=1,NDOF
      DO 5 I=1,M
    5 B(J)=B(J)-ALPH(I)*EV(J,I)
      RETURN
      END
```

POOR ORIGINAL



```
      SUBROUTINE INVERT(N,MB,A,B,JS,INDEX)
      DIMENSION A(N,MB),MS(1000),G(1000),B(N)
      LEVEL 2,A,B
C
C     PROGRAM TO OBTAIN A-1 USING GAUSS ELIMINATION AS OUTLINED
C     IN BATHE-WILSON PAGES 248-258....A STORED IN BANDED FORM
C     N EQUATIONS  MB  BANDWIDTH...INVERSE STORED BANDED IN B
C
C     INDEX= 1 FIND INVERSE OF A ONE COL (JS) AT A TIME
C          = 2 FACTOR A
C          = 3 FIND SOL A X = B...ENTER WITH 2 FIRST TO FACTOR A
      IF(INDEX.EQ.3) GO TO 21
      IF(JS.GT.1.AND.INDEX.EQ.1) GO TO 15
      IF(N.LE.1000) GO TO 16
      WRITE(6,100) N
  100 FORMAT(//5X,27HNO OF EQUATIONS IN INVERT =,I5,2X
     1 ,12HEXCEEDS 1000)
      STOP
   16 CONTINUE
C      CALC SKYLINE FOR ALL COLUMNS
      DO 1 J=1,MB
    1 MS(J)=1
      JC=MB+1
      DO 2 J=JC,N
    2 MS(J)=MS(J-1)+1
C      REDUCE A
      DO 5 J=2,N
      JLOC=J-MS(J)+1
      G(MS(J))=A(MS(J),JLOC)
      IS=MS(J)+1
      IL=J-1
      IF(IS.GT.IL) GO TO 10
      DO 6 I=IS,IL
      ILOC=J-I+1
      G(I)=A(I,ILOC)
      MM=MS(J)
      IF(MS(I).GT.MM)MM=MS(I)
      KL=I-1
      IF(MM.GT.KL) GO TO 6
      DO 7 KR=MM,KL
    7 G(I)=G(I)-A(KR,I-KR+1)*G(KR)
    6 CONTINUE
   10 CONTINUE
C
      IS=MS(J)
      IF(I$.GT.IL) GO TO 11
      DO 8 I=IS,IL
      ILOC=J-I+1
    8 A(I,ILOC)=G(I)/A(I,1)
C
   11 CONTINUE
      IF(I$.GT.IL) GO TO 5
      DO 9 KR=I$,IL
    9 A(J,1)=A(J,1)-A(KR,J-KR+1)*G(KR)
C
    5 CONTINUE
      IF(INDEX.EQ.2) RETURN
```

```
   21 IF(INDEX.EQ.3) GO TO 22
C     FORM INVERSE
   15 CONTINUE
      J=JS
C     REDUCE RHS
      DO 23 I=1,N
   23 B(I)=0.
      B(J)=1.
   22 G(1)=B(1)
      DO 41 I=2,N
      G(I)=B(I)
      KS=MS(I)
      KL=I-1
      IF(KS.GT.KL) GO TO 41
      DO 42 K=KS,KL
      ILOC=I-K+1
   42 G(I)=G(I)-A(K,ILOC)*G(K)
   41 CONTINUE
C     BACKSUBSTITUTE TO GENERATE J COL OF A-1 FROM N TO 1
      DO 49 I=1,N
   49 G(I)=G(I)/A(I,1)
      B(N)=G(N)
      IT=N-1
      IF(IT.EQ.0) GO TO 40
      DO 45 IN=1,IT
      I=N+1-IN
      KS=MS(I)
      KL=I-1
      DO 46 K=KS,KL
   46 G(K)=G(K)-A(K,I-K+1)*B(I)
      B(I-1)=G(I-1)
   45 CONTINUE
   40 CONTINUE
      RETURN
      END
```

POOR ORIGINAL



```
      SUBROUTINE EXPAND(EV,TO,TI,ESAP,NRDOF,MD,MODE,NTO,NEQB,NBLOCK,
     1 NTYP,NS,XM,NST)
      COMMON/EVAL/EF(20)
      DIMENSION EV(NRDOF,MD),TO(NTO,NRDOF),TI(NRDOF),ESAP(NEQB,MODE)
     1  ,XM(NST),XMTOT(20)
      LEVEL 2,EV,TO,TI,ESAP,XM
C
C     EXPAND MODES AND WRITE ONTO 7 IN NBLOCKS
C
C
C      STORE TO ONE ROW AT A TIME ON 17
      REWIND 18
      REWIND 17
      NBL=NS/NTO+1
      IF(NS.EQ.NTO) NBL=1
      NROW=0
      DO 1 IB=1,NBL
      READ(18) TO
      DO 2 I=1,NTO
      IF(NROW+I.GT.NS) GO TO 2
      WRITE(17) (TO(I,J),J=1,NRDOF)
    2 CONTINUE
    1 NROW=NROW+NTO
C
C     EXPAND MODES
C
      REWIND 16
      REWIND 17
      REWIND 18
      NROW=0
      DO 10 IB=1,NBLOCK
      DO 12 IRW=1,NEQB
      DO 16 IFR=1,MODE
   16 ESAP(IRW,IFR)=0.
      IF(NROW+IRW.GT.NS+NRDOF) GO TO 12
      IF(NROW+IRW.GT.NRDOF) GO TO 13
      DO 20 IFR=1,MODE
   20 ESAP(IRW,IFR)=EV(IRW+NROW,IFR)
      GO TO 12
   13 READ(17) (TO(I,J),J=1,NRDOF)
      DO 14 L=1,NRDOF
      DO 14 IFR=1,MODE
   14 ESAP(IRW,IFR)=ESAP(IRW,IFR)-TO(I,L)*EV(L,IFR)
      IF(NTYP.EQ.2) GO TO 12
      READ(16) TI
      DO 15 L=1,NRDOF
      DO 15 IFR=1,MODE
      EIV=(EF(IFR)*6.28)**2.
   15 ESAP(IRW,IFR)=ESAP(IRW,IFR)-EIV*TI(L)*EV(L,IFR)
   12 CONTINUE
      WRITE(18) ESAP
   10 NROW=NROW+NEQB
C     NORMALIZE VECTORS AND WRITE ON 7
      DO 30 I=1,NST
   30 READ(15) XM(I)
      REWIND 15
      REWIND 18
```

```
      DO 31 I=1,MODE
   31 XMTOT(I)=0.
      IROW=0
      DO 32 IB=1,NBLOCK
      READ(18) ESAP
      DO 33 JM=1,MODE
      DO 33 I=1,NEQB
      NROW=IROW+I
      IF(NROW.GT.NST) GO TO 33
      XMTOT(JM)=XMTOT(JM)+ESAP(I,JM)*ESAP(I,JM)*XM(NROW)
   33 CONTINUE
      IROW=IROW+NEQB
   32 CONTINUE
      REWIND 18
      IROW=0
      DO 34 IB=1,NBLOCK
      READ(18) ESAP
      DO 35 JM=1,MODE
      DO 35 I=1,NEQB
      IF(IROW+I.GT.NST) GO TO 35
      ESAP(I,JM)=ESAP(I,JM)/SQRT(XMTOT(JM))
   35 CONTINUE
      IROW=IROW+NEQB
      WRITE(7) ESAP
   34 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE POWER(XK,XM,EV,XD,B,BB,EDUM,NSI,NDOF,MODE)
      COMMON/EVAL/EF(20)
      DIMENSION XK(NSI),XM(NSI),EV(NDOF,MODE),B(NDOF),BB(NDOF)
     1 ,XD(NDOF,NDOF),EDUM(MODE)
      LEVEL 2,XK,XM,EV,XD,B,BB,EDUM
C
C     INVERSE ITERATION TO COMPUTE (MODE) LOWEST MODES....
C     EACH ITERATION STATRS WITH UNIT VECTOR FOLLOWED WITH
C     GRAM SCHMIDT ORTHOGONALIZATION
C
      WRITE(6,100) MODE
  100 FORMAT(//5X,22HINVERSE ITERATION FOR ,I5,6H MODES//
     1          2X,4HMODE,2X,9HITERATION,2X,10HEIGENVALUE,2X,5HERROR)
      TOL=.0001
      IMAX=30
C     FACTOR K
      DO 1 I=1,NDOF
      DO 1 J=1,NDOF
    1 XD(I,J)=0.
      DO 2 I=1,NDOF
      DO 2 K=1,NDOF
      CALL PPLOC(I,K,ILOC)
      KCOL=K-I+1
    2 XD(I,KCOL)=XK(ILOC)
      CALL INVERT(NDOF,NDOF,XD,B,JDUM,2)
      DO 20 IM=1,MODE
C     TRIAL VECTOR
      DO 3 I=1,NDOF
    3 B(I)=1.
      IF(IM.GT.1) CALL GRAMS(XM,EV,B,BB,IM,NSI,NDOF,MODE)
      ITER=1
      ERR=0.
C     M * X
   12 DO 4 I=1,NDOF
      BB(I)=0.
      DO 4 J=1,NDOF
      CALL PPLOC(I,J,ILOC)
    4 BB(I)=BB(I)+XM(ILOC)*B(J)
      CALL INVERT(NDOF,NDOF,XD,BB,JDUM,3)
C     X' * M * X
      DO 5 I=1,NDOF
      B(I)=0.
      DO 5 J=1,NDOF
      CALL PPLOC(I,J,ILOC)
    5 B(I)=B(I)+XM(ILOC)*BB(J)
      DUM=0.
      DO 6 I=1,NDOF
    6 DUM=DUM+B(I)*BB(I)
C     XLAM = X' * K * X / DUM
      DO 7 I=1,NDOF
      B(I)=0.
      DO 8 J=1,NDOF
      CALL PPLOC(I,J,ILOC)
    8 B(I)=B(I)+XK(ILOC)*BB(J)
    7 CONTINUE
      DUMK=0.
      DO 9 I=1,NDOF
```

POOR ORIGINAL



```
    9 DUMK=DUMK+B(I)*BB(I)
      XLAM=DUMK/DUM
      DUM=SQRT(DUM)
      DO 10 I=1,NDOF
   10 B(I)=BB(I)/DUM
      WRITE(6,101) IM,ITER, XLAM,ERR
  101 FORMAT(1X,I5,3X,I5,2X,E13.3,2X,E13.3)
      IF(IM.GT.1) CALL GRAMS(XM,EV,B,BB,IM,NS1,NDOF,MODE)
      ITER=ITER+1
      IF(ITER.GT.2) GO TO 11
      XLAML=XLAM
      GO TO 12
   11 IF(ITER.GT.IMAX) GO TO 15
      ERR=ABS(XLAM-XLAML)/XLAM
      IF(ERR.LE.TOL) GO TO 13
      XLAML=XLAM
      GO TO 12
   15 WRITE(6,104)
  104 FORMAT(5X,25HNO OF ITERATIONS EXCEEDED)
   13 DO 14 I=1,NDOF
   14 EV(I,IM)=B(I)
      EF(IM)=SQRT(XLAM)/6.28
   20 CONTINUE
      WRITE(6,110)
  110 FORMAT(//5X,31HITERATION CONVERGED  MODAL DATA//)
      DO 21 IM=1,MODE
      WRITE(6,111) IM,EF(IM)
  111 FORMAT(5X,5HMODE ,I5,12HEIGENVALUE =,E13.3,4H CPS/
     1        10X,12HMODE SHAPE =)
      WRITE(6,112)(EV(I,IM),I=1,NDOF)
  112 FORMAT(5X,6(E13.3,2X))
      EDUM(IM)=EF(IM)*6.28
   21 CONTINUE
      REWIND 7
      WRITE(7) EDUM
      RETURN
      END
```

APPENDIX B

USE OF MODIFIED SAPV AT

BROOKHAVEN NATIONAL LABORATORY

The following card deck should be used to run the modified SAPV program on Brookhaven National Laboratories' CDC 7600 computer:

```
MSAP, STMFZ, T(t), P2.

              t = running time

ACCOUNT (Name, Number)

              Name, Number = Valid Account

ATTACH (OLDPL, SAP5, ID=ZZGAJPH)

ATTACH (CORR, GUYAN, ID=ZZGCAM)

ATTACH (LIB1, FR80LIB)

ATTACH (LIB2, CALCOMPLIB)

LIBRARY (LIB1, LIB2)

UPDATE (F, M=CORR, L=A12)

FTN (I=COMPILE, L=0)

MAP(OFF)

SEGLOAD.

LD SET (PRESET = ZERO)

LGO.

EOR

 * ID CAM

 * D PLOTTT.1

 * I CORR2.2

              Card CAM.1   (All CAM - cards are listed in Appendix A).
```

```
  * D LEVEL2.1

  * I COMSIZE.3

              Cards CAM.2 through CAM.4

  * D LEVEL 2.3

  * I LEVEL 2.2

              Card  CAM.5

  * I SAP 5.88

              Card  CAM.6 through CAM.8

  * I SAP 5.296

              Card  CAM.9 through CAM.76

  * I SAP.5 385

              Card  CAM.77

  * I SAP.5 387

              Card CAM.78

EOR

 Segment cards as for normal SAPV run

EOR

 Data as specified in SAPV manual Ref. 5 and modified in Section 3.2

EOF
```

NRC FORM 335 (7-77)

U.S. NUCLEAR REGULATORY COMMISSION

**BIBLIOGRAPHIC DATA SHEET**

| | |
|---|---|
| 4. TITLE AND SUBTITLE (Add Volume No., if appropriate)<br>Reduction of Structural Degrees of Freedom | 1. REPORT NUMBER (Assigned by DDC)<br>NUREG/CR-1938<br>BNL-NUREG-51341 |
| | 2. (Leave blank) |
| | 3. RECIPIENT'S ACCESSION NO. |
| 7. AUTHOR(S)<br>C. A. Miller | 5. DATE REPORT COMPLETED<br>MONTH: December YEAR: 1980 |
| 9. PERFORMING ORGANIZATION NAME AND MAILING ADDRESS (Include Zip Code)<br>Brookhaven National Laboratory<br>Upton, NY 11973 | DATE REPORT ISSUED<br>MONTH: April YEAR: 1981 |
| | 6. (Leave blank) |
| | 8. (Leave blank) |
| 12. SPONSORING ORGANIZATION NAME AND MAILING ADDRESS (Include Zip Code)<br>Office of Nuclear Regulatory Research<br>Division of Reactor Safety Research<br>U.S. Nuclear Regulatory Commission<br>Washington, DC 20555 | 10. PROJECT/TASK/WORK UNIT NO. |
| | 11. CONTRACT NO.<br>FIN A3216 |
| 13. TYPE OF REPORT<br>Technical Report | PERIOD COVERED (Inclusive dates)<br>June 1980 - December 1980 |
| 15. SUPPLEMENTARY NOTES | 14. (Leave blank) |

16. ABSTRACT (200 words or less)

An assessment is made of the errors in large structural systems when both Guyan and Modified Guyan Reduction are used. Second, the economies that may be achieved by using these reduction methods as compared with obtaining complete solutions (i.e., no reduction) are investigated.

17. KEY WORDS AND DOCUMENT ANALYSIS

17a. DESCRIPTORS

17b. IDENTIFIERS/OPEN-ENDED TERMS

| | | |
|---|---|---|
| 18. AVAILABILITY STATEMENT<br>Unlimited | 19. SECURITY CLASS (This report)<br>Unclassified | 21. NO. OF PAGES |
| | 20. SECURITY CLASS (This page)<br>Unclassified | 22. PRICE<br>$ |

NRC FORM 335 (7-77)