CASE WESTERN RESERVE UNIVERSITY
University Circle,
Cleveland, Ohio 44106

p 139

DYNAMIC SUBSTRUCTURING BY THE BOUNDARY FLEXIBILITY VECTOR METHOD OF COMPONENT MODE SYNTHESIS

by

AYMAN AHMED ABDALLAH

Submitted in partial fulfillment of the requirements

for the Degree of Doctor of Philosophy

Thesis Advisor: Dr. Arthur A. Huckelbridge

Department of Civil Engineering

CASE WESTERN RESERVE UNIVERSITY

January 12, 1990

DYNAMIC SUBSTRUCTURING BY THE BOUNDARY FLEXIBILITY VECTOR METHOD OF COMPONENT MODE SYNTHESIS

Abstract

by

AYMAN AHMED ABDALLAH

Component mode synthesis (CMS) is a method of dynamic analysis, for structures having a large number of degrees of freedom (D.O.F.). These structures often required lengthy computer CPU time and large computer memory resources, if solved directly by the finite-element method (FEM). In CMS, the structure is divided into independent components in which the D.O.F. are defined by a set of generalized coordinates defined by displacement shapes. The number of the generalized coordinates are much less than the original number of physical D.O.F., in the component. The displacement shapes are used to transform the component property matrices and any applied external loads, to a reduced system of coordinates. Reduced system property matrices are assembled, and any type of dynamic analysis is carried out in the reduced coordinate system. Any obtained results are back transformed to the original component coordinate systems. In all conventional methods of CMS, the mode shapes used for components,

are dynamic mode shapes, supplemented by static deflected shapes. Historically, all the dynamic mode shapes used in conventional CMS are the natural modes (eigenvectors) of components.

This work presents a new method of CMS, namely the boundary flexibility vector method of CMS. The method provides for the incorporation of a set of static Ritz vectors, referred to as boundary flexibility vectors, as a replacement and/or supplement to conventional eigenvectors, as displacement shapes for components. The generation of these vectors does not require the solution of a costly eigenvalue problem, as in the case of natural modes in conventional CMS, and hence a substantial saving in CPU time can be achieved. The boundary flexibility vectors are generated from flexibility (or stiffness) properties of components. The formulation presented is for both free and fixed-interface components, and for both the free and forced vibration problems. Free and forced vibration numerical examples are presented to verify the accuracy of the method and the saving in CPU time. Compared to conventional methods of CMS, the results indicate that by using the new method, more accurate results can be obtained with a substantial saving in CPU time.

To my Mother, my Wife

and

my Son

ACKNOWLEDGEMENTS

I would like to express my deepest and sincere gratitude to my advisor Dr. Huckelbridge, for his guidance, advice, suggestions and friendship. Working with him was an enlightening experience for me. I would like to express my appreciation to Dr. D. Gasparini, for being an excellent teacher in all courses I have taken with him, thus providing me with an excellent foundation in my area of research and in other areas of great interest to me. Also I would like to thank him for being a member of my graduate committee. Special thanks are due to Dr. C. Lawrence, at NASA-Lewis Research Center, for his thoughtful suggestions and being a member of my graduate committee. I would like to acknowledge the members of the graduate committee, Dr. R. Mullen and Dr. R. Quinn for their valuable suggestions. I wish to express my gratitude to my wife, Nevine, for her understanding, love, support and patience in typing the text.

The support of this research work by the NASA Lewis Research Center at Cleveland, Ohio, under grant NAG 3-707, is gratefully acknowledged.

Table of Contents

List of Figures

Chapter 1

Introduction

Since the late fifties and early sixties, there has been rapid advancement in the field of structural dynamics. The primary cause of this advancement was the introduction of digital computers and the associated analytical software which made the application of numerical methods to dynamic analysis problems more feasible. The most powerful of these numerical techniques was the finite element method, which can be considered the basis for a new era in the field of mathematics (more precisely in the solution of partial differential equations), and consequently many fields of science, including structural dynamics.

Previous to the finite element method, there existed other approximate methods of analysis, such as Rayleigh's method in dynamic analysis. The main disadvantage of this type of analysis is that it requires the assumption of reasonably accurate mode shapes for the structure, which is not always readily accomplished, especially in large and complex structures. Compared to these previous methods of analysis, the finite element method is more accurate, more direct (no mode shapes need to be assumed) and can be applied to a structure of any degree of complexity.

Following the finite element method, a new modeling technique, namely component mode synthesis (CMS), was introduced by Hurty [1] in the mid-sixties. Component mode synthesis is a

method of dynamic substructuring, in which the structure is divided into independent components or substructures. The main motivation for introducing the method arose in the dynamic analysis of large structures, having a large number of physical degrees of freedom. These large structures usually required lengthy computer CPU time and large computer memory resources, when solved by finite elements directly. By using CMS, savings in computer time and memory can be attained. Moreover, other advantages can also be attained through the use of dynamic substructuring such as:

1- Independent design and analysis efforts for various components of a structure.

2- Reducing effort if system can be divided into identical repetitive components having the same constraints and displacement modes.

3- The incorporation of experimental and analytical data can be achieved, for the characterization of components.

4- Reducing computational cost of a reanalysis, in case not all components are modified.

The saving in CPU time and computer memory is achieved by reducing the number of degrees of freedom, associated with each component. This is accomplished by defining the displacements of each component by a set of generalized coordinates, consisting of the amplitudes of a corresponding set of mode shapes. In all conventional CMS methods, the mode shapes used are dynamic mode

shapes, supplemented by static deflected shapes. All the dynamic modes historically used in CMS, are natural modes of vibration (eigenvectors) for the component.

This work presents a new method of CMS, namely the boundary flexibility vector method of CMS, which is based upon utilizing a set of static Ritz vectors, as displacement shapes for components. The generation of these vectors, referred to herein as "boundary flexibility vectors", does not require the solution of a costly eigenvalue problem, as in the case of natural modes in traditional CMS. Thus, a substantial saving in computer time is obtained. The presented method is general and can be utilized for any type of dynamic analysis.

This work is divided into six chapters. Chapter one is an introduction to the work. Chapter two is a brief review of traditional methods of CMS, and previous work in the application of static Ritz vectors in structural dynamics. The formulation of the boundary flexibility method of CMS is presented in chapter three. The presented formulation is for both fixed and free-interface components. Also, a hybrid boundary flexibility and traditional CMS formulation is presented. A comparison of the number of operations required to generate the boundary flexibility vectors versus eigenvectors is given at the end of the chapter. In chapter four, small and large free vibration problems are utilized to verify the accuracy and CPU time saving of the new method compared to finite element representations and traditional

methods of CMS. Chapter five presents the formulation of the boundary flexibility method of CMS for the forced vibration problem. An example is solved numerically by several methods and for different loading conditions. The transient response is compared for all methods. Chapter six includes a work summary, conclusions and ideas for future work in this field.

Chapter 2

Literature Review

2.1) Introduction:

This chapter is divided into two main and independent parts. The first part is a very brief review of component mode synthesis. The main steps of CMS, classification of components and type of displacement shapes are introduced in this part. Part two is a detailed review of previous work, in the application of load dependent Ritz vectors in structural dynamics.

2.2) Review of Component Mode Synthesis:

As mentioned before in chapter one, component mode synthesis (CMS) was introduced by Hurty [1] in 1965. Since then, several different variations of the method were introduced. However, the main idea of reducing the number of degrees of freedom used to characterize components, through use of a truncated set of mode shapes and generalized coordinates, is common to all CMS methods. A detailed review of CMS methods was given in a previous work [2] by the author. The main steps of conventional CMS, as given in reference [2], are summarized as follows:

1- Divide the system into independently characterized components. Each component has a vector {u} of total physical degrees of freedom (D.O.F.). (See Figure (1)). A component can be connected to other components or to the ground through a support, as shown in Figure (1). The common boundary between components is the interface. Let $\{u_c\}$ be the vector of D.O.F. contained in the

interfaces of the component. Thus, {u} is partitioned into

$$\{u\} = \begin{Bmatrix} u_c \\ u_i \end{Bmatrix} \tag{2.1}$$

where $\{u_i\}$ = vector of internal D.O.F. in component.

= complement of $\{u_c\}$ in {u}.

2- Define the displacement of each component by a set of mode shapes and generalized coordinates. The generalized coordinates can be taken as the amplitudes of these mode shapes. The mode shapes used contain static displacement shapes, as well as a truncated set of natural modes of vibration. Usually a very small number of static and dynamic displacement shapes is required, to obtain a good representation of the component in the new coordinate system. Thus, the number of the new set of generalized coordinates is generally much smaller than the original number of physical D.O.F., in the component.

3- For each component, transform the property matrices (mass, stiffness and damping), from the physical coordinate system to the new reduced system of generalized coordinates.

4- Couple all the components by enforcing displacement compatibility requirements at the interfaces between components, thereby assembling reduced system property matrices.

5- Perform any type of dynamic analysis (eigenproblem solution, transient or steady-state response,etc...) in the reduced coordinate system of generalized coordinates.

6- Any obtained results are back transformed to the original

system of physical D.O.F.

For all variations of C.M.S., within the individual characterization of components, interface D.O.F. can be considered either free or totally fixed. Accordingly, components can be classified into either fixed or free-interface components. For a detailed discussion and formulation of fixed-interface components see Hurty [1], Abdallah [2], Craig and Bampton [3] and Craig [4]. While for free-interface components see Abdallah [2], MacNeal [5], Rubin [6] and Martinez and Gregory [7]. The first investigator to introduce fixed-interface components, was Hurty [1] in 1965, while free-interface components were introduced by Goldman [8] in 1969. The method of coupling components, introduced by Goldman [8], was rather complicated. Several alternative methods of coupling free-interface components were introduced by Hou [9], Dowell [10] and Abdallah [2]. In 1971, following Goldman's work, MacNeal [5] gave a better understanding and representation of free-interface components through the use of the idea of residual flexibility. In 1975, Rubin [6] expanded MacNeal's work to free-interface components, having rigid body modes.

It should be noted that the decision to characterize interface D.O.F. into free or fixed often depends on the type of displacement shapes available to the analyst, i.e., the fixed or free-interface type. It also depends on the judgment of the analyst as to what level of constraint each component imposes on the other. The displacement shapes used for the characterization

of components, may be obtained either experimentally or analytically. There are several different types of displacement shapes used in the literature of C.M.S. They are summarized in the following: For fixed-interface components:

1- Fixed-interface normal modes of vibration $[\Phi_n]$: They are the natural modes of vibration (or the eigenvectors), obtained from solving the eigenvalue problem associated with the equation of motion of the component. They are computed assuming all the interface degrees of freedom fixed. Only a truncated set of normal modes $[\Phi_k]$ (kept modes) are used. Usually the low frequency mode shapes are the ones used or kept.

2- Constraint modes $[\Phi_c]$: A constraint mode $\{\phi_c\}$ is defined as, the static displaced shape obtained by applying a unit displacement at one interface D.O.F. while totally constraining all other interface D.O.F. in the component. The number of constraint modes used for each component is equal to the number of interface D.O.F. in that component. The generalized coordinates associated with these constraint modes, are usually taken as the interface D.O.F. in components. Rigid-body modes can be included in these constraint modes.

For free-interface components, the following displacement shapes are used:

1- Free-interface normal modes of vibration $[\Phi_n]$: They are the same as in fixed-interface components except they are computed assuming all interface D.O.F. free.

2- Rigid-body modes $[\Phi_r]$: They are used for free-interface components having no constraints preventing them from rigid-body motion.

3- Attachment modes $[\Phi_a]$: They are defined on a subset $\{u_a\}$ of the total physical degrees of freedom $\{u\}$. An attachment mode $\{\phi_a\}$ is defined as the static displaced shape obtained by imposing a unit force (or moment) on one D.O.F. of $\{u_a\}$ with zero forces on all remaining D.O.F. in $\{u_a\}$. According to Meirovitch [11], Bamford was the first investigator to introduce attachment modes in 1967. They were used to account for the effect of concentrated loads on unconstrained D.O.F.

4- Residual flexibility modes $[\Psi_d]$: They are static displacement shapes introduced by MacNeal [5] in 1971, to obtain a better representation for free-interface components. (See (5)). They were introduced to replace attachment modes as static mode shapes for the component.

5- Inertia relief attachment modes $[\Psi_a]$: They are the static displacement shapes, for components having rigid-body modes, analogous to attachment modes $[\Phi_a]$ in case of components having no rigid-body modes. They were introduced by Rubin [6] in 1975.

6- Residual inertia relief attachment modes $[\Psi_r]$: They are static displacement shapes introduced by Rubin [6] in 1975. They were introduced to be used for free-interface components having rigid-body modes analogous to residual flexibility modes $[\Psi_d]$, used for components having no rigid-body modes.

2.3) Review of Previous Work with Load-Dependent Ritz Vectors:

The application of load-dependent Ritz vectors was introduced by Wilson et al. [12] in 1982. They were the first investigators to point out the usefulness of load-dependent Ritz vectors in dynamic analysis. The vectors were used to calculate transient dynamic response by Ritz vector superposition. The load-dependent Ritz vectors were used, instead of natural modes of vibration (eigenvectors), in a fashion equivalent to mode shapes. In all presented examples, it was shown that the superposition of load-dependent Ritz vectors yielded more accurate forced dynamic response results, with fewer number of vectors, than if the natural modes (eigenvectors) were used. Furthermore, the load-dependent Ritz vectors are generated with less computational effort than eigenvectors. The method used by Wilson et. al [12] is summarized in the following:

Consider a damped system, having n degrees of freedom. Its equation of motion is:

$$[M]\{\ddot{u}\} + [C]\{\dot{u}\} + [K]\{u\} = \{f(s)\}\, g(t) \qquad (2.2)$$

where $[M]$ = mass matrix of system

$[C]$ = damping matrix of system

$[K]$ = stiffness matrix of system

$\{u\}$ = displacement vector

$\{\dot{u}\}$ = velocity vector

$\{\ddot{u}\}$ = acceleration vector

$\{f(s)\}$ = vector of spatial distribution of

external load

$g(t)$ = time-dependent amplitude of $\{f(s)\}$

The load-dependent Ritz vectors are generated by Wilson's algorithm given in reference [12]. The main steps are as follows:

1- Given [M], [K] and $\{f(s)\}$.

2- Solve for first vector $\{x_1^*\}$ and normalize with respect to the mass matrix to obtain $\{x_1\}$.

$$[K]\ \{x_1^*\} = \{f(s)\}$$

$$\{x_1\} = \{x_1^*\} \ / \ (\ \{x_1^*\}^T\ [M]\ \{x_1^*\}\ )^{1/2}$$

where $\{x_1^*\}^T$ is the transpose of $\{x_1^*\}$

3- Solve for additional vectors (i = 2,3,4,.....,L)

$$[K]\ \{x_i^*\} = [M]\ \{x_{i-1}\}$$

- Orthogonalize with respect to [M], with all previous (i-1) vectors

$$\{x_i^{**}\} = \{x_i^*\} - \sum_{j=1}^{i-1} (\ c_j\ \{x_j\})$$

where $c_j = \{x_j\}^T\ [M]\ \{x_i^*\}$

- Normalize vector $\{x_i^{**}\}$ with respect to [M] obtaining $\{x_i\}$

$$\{x_i\} = \{x_i^{**}\} \ / \ (\ \{x_i^{**}\}^T\ [M]\ \{x_i^{**}\}\ )^{1/2}$$

4- Orthogonalize all obtained vectors with respect to the stiffness matrix [K] (optional step). This step is done as follows:

- Form [X] = matrix containing all previously calculated load-dependent Ritz vectors in its columns. Its dimension is (n x L)

$$[A] = [X]^T [K] [X]$$

$$[I] = [X]^T [M] [X] = \text{identity matrix of order L}$$

- Solve the following reduced-order (L x L) eigenvalue problem. (Solve for $\{y_i\}$, i = 1,.....,L)

$$\left[[A] - \omega_i^2 [I] \right] \{y_i\} = \{0\}$$

where $\{y_i\}$ = i^{th} eigenvector

ω_i^2 = i^{th} eigenvalue

- Compute final orthogonal vectors $[X^\circ]$ from

$$[X^\circ] = [X] [Y]$$

where $[X^\circ]$ = an (n x L) matrix containing final load-dependent Ritz vectors in its columns. All vectors are orthogonal with respect to the mass and stiffness matrices.

[Y] = an (L x L) matrix containing the eigenvectors $\{y_i\}$ in its columns.

According to Wilson's algorithm, load-dependent Ritz vectors can be defined as static displacement shapes generated from the spatial distribution of the external loading applied on the system. It can be seen that the initial load-dependent Ritz vector is the static displacement of the system, due to the applied load, while all subsequent vectors are obtained from the

inertial loading of the response. Thus, it is ensured that all the generated load-dependent Ritz vectors, are excited by the applied external load pattern, and hence, will contribute to the response. While in the case where natural modes of vibration (eigenvectors) are used, there could be mode shapes which contribute little or nothing to the response (as the case of a symmetric spatial distribution of load and an anti-symmetric mode shape) even if the natural frequency of the mode shape is well represented in the loading frequency content.

After obtaining the load-dependent Ritz vectors $[X^{\circ}]$, the displacement vector $\{u\}$ is approximated by

$$\{u\} \cong \sum_{i=1}^{L} \{X_i^{\circ}\} \; Z_i(t) \qquad (2.3)$$

where $Z_i(t)$ = time-dependent generalized coordinates associated with load-dependent vector i.

The mass, stiffness and damping matrices are transformed to a reduced set of coordinates by:

$$[M^*] = [X^{\circ}]^T [M] [X]$$

$$= \text{identity matrix } [I] \text{ of order L}$$

$$[K^*] = [X^{\circ}]^T [K] [X]$$

$$[C^*] = [X^{\circ}]^T [C] [X] \qquad (2.4)$$

Where $[M^*]$, $[K^*]$ and $[C^*]$ are the transformed property matrices. $[M^*]$ and $[K^*]$ are diagonal because the generated load-dependent Ritz vectors $[X^{\circ}]$ are [M]-orthonormal and [K]-orthogonal. If proportional damping is used,

$$[C] = a_o [M] + a_1 [K] \qquad (2.5)$$

where a_o and a_1 are constants, then $[C^*]$ is also diagonal. Thus, the final system equations of motion are diagonalized and reduced to L uncoupled, linear, second order, ordinary differential equations, as follows:

$$[M^*] \{\ddot{Z}\} + [C^*] \{\dot{Z}\} + [K^*] \{Z\} = [X^\circ]^T \{f(s)\} g(t) \qquad (2.6)$$

These uncoupled equations are solved using any numerical technique, obtaining the desired response. Wilson et al. [12] also introduced an expression that provides an error estimation for the dynamic analysis. As mentioned before, the numerical examples introduced by Wilson indicate that by using a fewer number of load-dependent Ritz vectors, more accurate results can be obtained than by using eigenvectors.

In a paper by Nour-Omid and Clough [13] in 1984, it was shown that by orthogonalizing any obtained load-dependent Ritz vector with only the two preceding vectors (step 3 in Wilson's algorithm), one can theoretically ensure orthogonality with all previously calculated vectors. The main problem in this procedure is that orthogonality can be lost due to roundoff errors, accumulating from one step to another. Thus, orthogonality must be checked, while generating each vector, by a certain scheme given in reference [13]. Also, it was shown how the loss of orthogonality can be corrected if it exceeds a certain tolerance limit. The method of generating the load-dependent Ritz vectors in reference [13], followed that outlined in Wilson's algorithm

through step number three. Hence, no orthogonalization was carried out with respect to the stiffness matrix, hence avoiding the solution of the (L x L) eigenvalue problem. The transformed stiffness and damping matrices thus obtained are not diagonal. However, it was shown in reference [13], that the equation of motion (2.2) can be reduced to a tridiagonal form, through a prescribed set of transformations. (Note that proportional damping, given by equation (2.5), is used). The final transformed equation of motion will be as follows:

$$[T_L]\ \{\ddot{Z}\} + \left[\ a_o\ [T_L] + a_1\ [I_L]\ \right]\ \{\dot{Z}\} + \{Z\} = \{e\}\ g(t) \qquad (2.7)$$

where $[T_L]$ = Transformed mass matrix of order L x L. It is a tridiagonal matrix.

$[I_L]$ = Transformed stiffness matrix.

= Identity matrix of order L x L.

$\{e\}$ = Transformation of $\{f(s)\}$ vector. All its elements are zeros except the first element is non-zero.

a_o, a_1, $\{\ddot{Z}\}$, $\{\dot{Z}\}$, $\{Z\}$ and $g(t)$ are as before.

As seen in equation (2.7), the excitation is applied only in the first of these equations of motion. While equations 2, 3, ...up to L are equations of free vibration.

In another paper by Arnold et al. [14] in 1985, it was stated that Wilson's algorithm had been implemented into the MSC/NASTRAN program. The method presented by Wilson et al. [12] was applied to structures having 1000 degrees of freedom. The results

presented in reference [14] showed that by using load-dependent Ritz vectors, instead of eigenvectors, the following was obtained:

1- A saving of 90 % for eigenvalue extraction.

2- A saving of 50 % for response calculations.

In 1986, Wilson and Bayo [15] introduced load-dependent Ritz vectors to fixed-interface component mode synthesis, in the calculation of forced responses by dynamic substructuring. The presented method had the following limitations:

1- The method is only suitable for forced vibration problems where there are external loads, thus it cannot be applied to free vibration problems.

2- The presented formulation is only applicable to fixed-interface components.

3- The displacement shapes used for components are constraint modes $[\Phi_c]$ and load-dependent Ritz vectors for components having external loads. The load-dependent Ritz vectors are generated from those external loads, assuming the interfaces of components totally fixed. Thus, if there are no external loads or they are applied at the interfaces of a component, then no load-dependent Ritz vectors are generated for this component. Hence, only constraint modes will be used, as displacement shapes for such a component, and the component representation is not expected to be good in that case. Wilson and Bayo suggest, for such components, to generate load-dependent Ritz vectors from a fictitious uniform external load, applied to the

component. In the presented problems, the components were planar beams and the fictitious uniform load was taken perpendicular to the length of the beams. The assumption of the uniform load does not have any theoretical basis and cannot be justified for complex three-dimensional components, where an assumption of a suitable uniform loading is not possible.

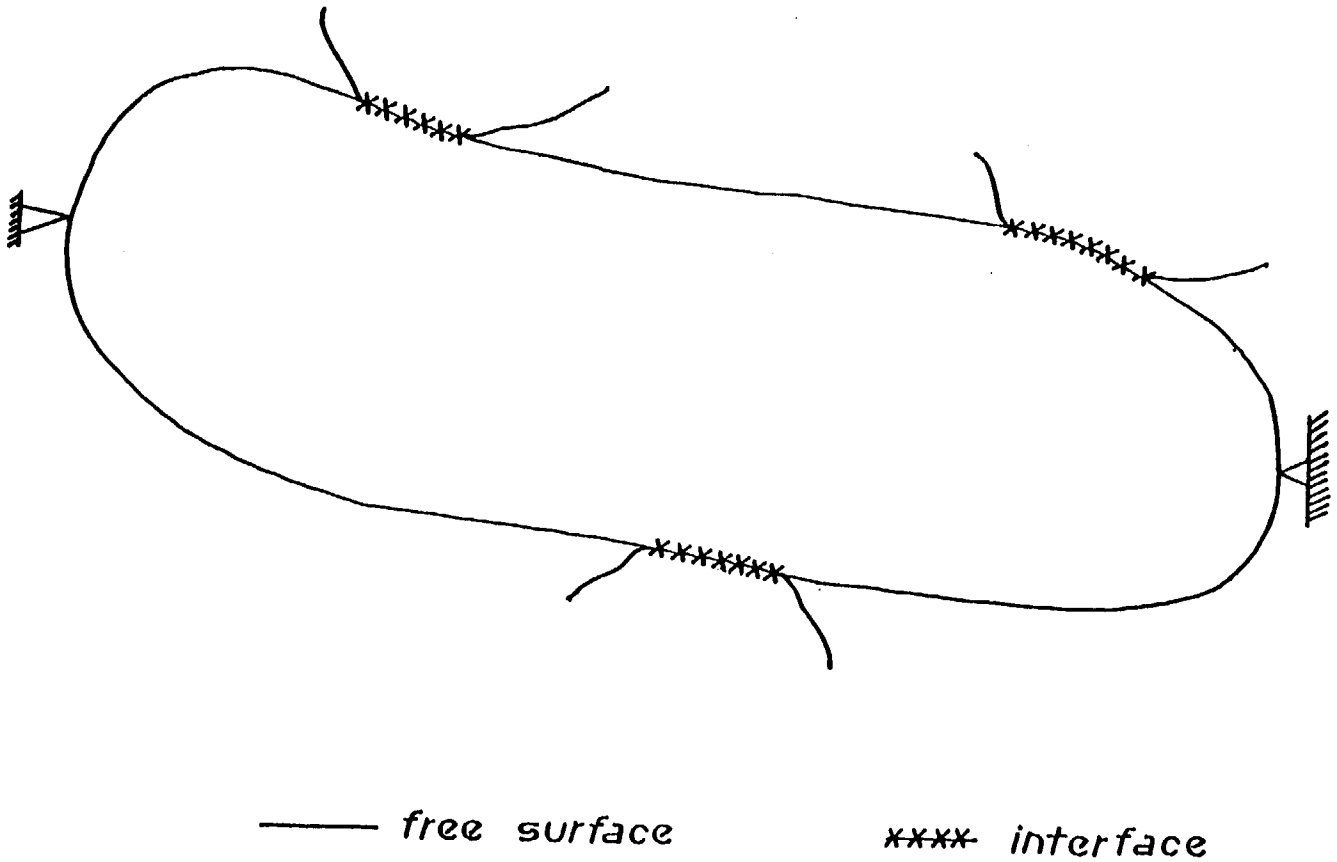


Figure (2.1) - A Structural Component

Chapter 3

Boundary Flexibility Vector Method of Component Mode Synthesis

3.1) Introduction:

The boundary flexibility vector method of CMS is presented in this chapter. All of the underlying equations will be derived. The method is described for both fixed and free-interface components. It should be noted that the developed formulation has many similarities to that of conventional CMS methods, thus facilitating the combination of conventional normal modes and boundary flexibility vectors, as a set of generalized coordinates for components. The boundary flexibility Ritz vectors are generated by an extension of Wilson's load-dependent Ritz vector algorithm, (described in the previous chapter). The boundary flexibility Ritz vectors are not, however, generated from external loads, applied to the system. Thus, they can be incorporated into both forced and free vibration problems. The generation of the boundary flexibility Ritz vectors does not require the solution of a costly eigenvalue problem, associated with the equation of motion of a component. Hence, a substantial saving in CPU time is generally achieved by using the boundary flexibility vectors instead of the conventional eigenvectors.

To illustrate the boundary flexibility vector method of CMS, a generic type of structural component will be used, as shown in Figure (1). Assume the total number of D.O.F. in the component is

n. Let $\{u_i\}$ and $\{u_c\}$ be the subvectors of internal and interface degrees of freedom in component. The lengths of $\{u_i\}$ and $\{u_c\}$ are i and c respectively. Hence, the total displacement vector {u} is partitioned into

$$\{u\} = \begin{Bmatrix} u_c \\ u_i \end{Bmatrix} \tag{3.1}$$

Accordingly, the mass [m] and stiffness [k] matrices of undamped component, can be partitioned into:

$$[m] = \begin{bmatrix} [m_{cc}] & [0_{ci}] \\ [0_{ic}] & [m_{ii}] \end{bmatrix} \quad \text{and} \quad [k] = \begin{bmatrix} [k_{cc}] & [k_{ci}] \\ [k_{ic}] & [k_{ii}] \end{bmatrix} \tag{3.2}$$

The method will be presented in the following sections, for both fixed and free-interface components.

3.2) Fixed-Interface Boundary Flexibility Vector Method of CMS:

As mentioned before, the formulation can be based entirely on the boundary flexibility Ritz vectors, or on a hybrid boundary flexibility vector/conventional CMS formulation. The two formulations are presented in the following:

3.2.1) Boundary-Flexibility Vector Formulation:

Consider the equation of motion for the undamped fixed-interface component

$$[m_{ii}] \{\ddot{u}_i\} + [k_{ii}] \{u_i\} = \{f(s)\}\, g(t) \tag{3.3}$$

where $\{\ddot{u}_i\}$ = acceleration vector of internal D.O.F.

{f(s)} and g(t) are as before in the previous chapter.

The reason the equation is written in terms of the internal D.O.F. $\{u_i\}$ only, is that the interface D.O.F. $\{u_c\}$ are assumed totally fixed, in the case of fixed-interface components. As the interface D.O.F. are not actually fixed and they undergo displacements within the motion of the whole system, then the forcing function on the component can be considered as support motions (accelerations) at the interface degrees of freedom. The idea of the pseudostatic influence vectors, commonly used in earthquake-response analysis (see Clough and Penzien [16]), is employed to obtain the loading function. According to reference [16], a pseudostatic influence vector defines the static response of the internal D.O.F. due to a unit displacement (motion) applied at one support (interface) D.O.F. In case each interface D.O.F. is subjected to a unit displacement, the total static response of the component is obtained by superposition of the pseudostatic influence vectors obtained from each independent unit displacement. In case the load is defined by a unit acceleration at one interface D.O.F., the pseudostatic influence vector is used to define the distribution of the load (acceleration) within the component. The definition of the pseudostatic influence vectors is thus similar to that of the constraint modes $[\Phi_c]$, previously defined in chapter (2). The size of matrix $[\Phi_c]$ is (n x c) and it contains all the constraint modes $\{\phi_c^j\}$ in its columns, where j = 1, 2,.....c. $[\Phi_c]$ is given by:

$$[\Phi_c] = \begin{bmatrix} [\Phi_{cc}] \\ [\Phi_{ic}] \end{bmatrix} = \begin{bmatrix} [I_{cc}] \\ [\Phi_{ic}] \end{bmatrix} \tag{3.4}$$

where $[I_{cc}]$ = identity matrix of order c

$[\Phi_{ic}]$ = part of constraint modes defining displacements of interior D.O.F.

$[\Phi_{ic}]$ is obtained from solving the lower partitions of the equations

$$\begin{bmatrix} [k_{cc}] & [k_{ci}] \\ [k_{ic}] & [k_{ii}] \end{bmatrix} \begin{bmatrix} [I_{cc}] \\ [\Phi_{ic}] \end{bmatrix} = \begin{bmatrix} [P_{cc}] \\ [0_{ic}] \end{bmatrix} \tag{3.5}$$

where $[P_{cc}]$ = interface forces between components

thus $[\Phi_{ic}] = - [k_{ii}]^{-1} [k_{ic}]$

For an acceleration $\ddot{u}_{cj}(t)$ given at interface D.O.F. number j, the forcing vector will be

$$\{f_j(s)\}\ g_j(t) = - [m_{ii}]\ \{\phi^j_{ic}\}\ \ddot{u}_{cj}(t) \tag{3.6}$$

Thus the total forcing vectors of equation (3.3) is given by:

$$\{f(s)\}\ g(t) = - \sum_{j=1}^{c} \left([m_{ii}]\ \{\phi^j_{ic}\}\ \ddot{u}_{cj}(t) \right) \tag{3.7}$$

The summation is carried out for all the accelerations applied at all the D.O.F. contained in $\{u_c\}$. It is clear that for the j^{th} forcing vector given by equation (3.6)

$$g_j(t) = - \ddot{u}_{cj}(t) \tag{3.8}$$

Thus the j^{th} spatial distribution of the force is

$$\{f_j(s)\} = [m_{ii}]\ \{\phi^j_{ic}\} \tag{3.9}$$

Wilson's algorithm can then be used to generate the boundary flexibility Ritz vectors, from each of the derived spatial distributions of forces. The j^{th} starting vector $\{q^*_{ij}\}$ is obtained from $\{f_j(s)\}$ by

$$[k_{ii}]\ \{q^*_{ij}\} = [m_{ii}]\ \{\phi^j_{ic}\} \tag{3.10}$$

Wilson's algorithm is used without the last step (step 4) of orthogonalization with respect to the $[k_{ii}]$ matrix. Each subsequent vector in the j^{th} set is then orthogonalized with respect to the mass matrix, with all the previous vectors generated in all the sets from 1 through j (step 3 in Wilson's algorithm). Each generated vector is normalized with respect to the mass matrix.

Let ℓ be the number of generated boundary flexibility Ritz vectors. Let $[Q_\ell]$ be the matrix containing the generated vectors in its columns. The size of $[Q_\ell]$ is (i x ℓ). Let $\{p_\ell\}$ be the vector containing ℓ generalized coordinates associated with the generated boundary flexibility vectors. Then $[\Phi_c]$ and $[Q_\ell]$ are used to transform the coordinates, from a physical coordinate system to a mixed physical and generalized one. The transformation relation is as follows:

$$\begin{Bmatrix} u_c \\ u_i \end{Bmatrix} = [T] \begin{Bmatrix} u_c \\ p_\ell \end{Bmatrix} = \begin{bmatrix} [I_{cc}] & [0]_{c\ell} \\ [\Phi_{ic}] & [Q_\ell] \end{bmatrix} \begin{Bmatrix} u_c \\ p_\ell \end{Bmatrix} \tag{3.11}$$

where [T] = transformation matrix of order (n) x (c+ℓ)

It should be noted that the number of boundary flexibility vectors, ℓ, must be less than the total number of internal D.O.F., i, in the component, for any reduction of the total number of D.O.F., to be achieved. This is usually the case, as a very small number of boundary flexibility vectors, is often sufficient to obtain a good representation of a component, as will be illustrated in the numerical examples. The transformation matrix [T] is used to transform the property matrices, given in equation (3.2), to the reduced mixed coordinate system. The final transformed component mass matrix $[\mu]$ and component stiffness matrix $[\Omega]$ are as follows:

$$[\mu] = [T]^T [m] [T] = \begin{bmatrix} [\mu_{cc}] & [\mu]_{c\ell} \\ [\mu_{\ell c}] & [\mu]_{\ell\ell} \end{bmatrix} \qquad (3.12)$$

where $[\mu_{cc}] = [m_{cc}] + [\Phi_{ic}]^T [m_{ii}] [\Phi_{ic}]$

$[\mu_{\ell c}] = [\mu]_{c\ell}^T = [Q_\ell]^T [m_{ii}] [\Phi_{ic}]$

$[\mu]_{\ell\ell} = [Q_\ell]^T [m_{ii}] [Q_\ell] = [I]_{\ell\ell}$ = identity matrix of order ℓ.

$$[\Omega] = [T]^T [k] [T] = \begin{bmatrix} [\Omega_{cc}] & [\Omega]_{c\ell} \\ [\Omega_{\ell c}] & [\Omega]_{\ell\ell} \end{bmatrix} \qquad (3.13)$$

where $[\Omega_{cc}] = [k_{cc}] + [k_{ci}] [\Phi_{ic}] = [k_{cc}] - [k_{ci}] [k_{ii}]^{-1} [k_{ic}]$

$$[\Omega_{\ell c}] = [\Omega]_{c\ell}^{T} = [0_{\ell c}]$$

$$[\Omega]_{\ell\ell} = [Q_{\ell}]^{T} [k_{ii}] [Q_{\ell}]$$

For every component, the boundary flexibility vectors are formed, and the transformation of coordinates is carried out to obtain reduced property matrices. As all displacements of the interfaces of the components are defined by physical D.O.F., then the assembly of system property matrices, at the interfaces, is simply done by direct summation of the interface portions of property matrices of components, as in conventional direct stiffness assembly of finite element property matrices. Any type of dynamic analysis is carried out using the reduced assembled system property matrices. Any obtained results are back transformed to the physical coordinate system, by utilizing equation (3.11).

3.2.2) Hybrid Boundary Flexibility Vector/ Conventional CMS Formulation:

The method can be applied for the general case where a combination of boundary flexibility vectors and natural vibration modes (eigenvectors), are used as displacement shapes of components. This case could arise when the number of available natural modes of vibrations, obtained from experiments, for example, are insufficient to give adequate representation of a component.

Let ℓ be the number of obtained boundary flexibility vectors, k be the number of any kept natural vibration modes, $[\Phi_k]$ be the

matrix containing the kept natural modes in its columns, and $\{p_k\}$ be the vector of generalized coordinates associated with the natural modes. $[Q_\ell]$ and $\{p_\ell\}$ are as before. It should be noted that the generated boundary flexibility vectors, contained in $[Q_\ell]$, are orthogonal to each other and they are orthogonalized with all the natural modes contained in $[\Phi_k]$, with respect to the mass matrix $[m_{ii}]$. $[Q_\ell]$, $[\Phi_{ic}]$ and $[\Phi_k]$ are used to transform the coordinates from the physical coordinate system to the reduced mixed physical and generalized one. The transformation is as follows:

$$\begin{Bmatrix} u_c \\ u_i \end{Bmatrix} = [T'] \begin{Bmatrix} u_c \\ p_\ell \\ p_k \end{Bmatrix} = \begin{bmatrix} [I_{cc}] & [0]_{c\ell} & [0]_{ck} \\ [\Phi_{ic}] & [Q_\ell] & [\Phi_k] \end{bmatrix} \begin{Bmatrix} u_c \\ p_\ell \\ p_k \end{Bmatrix} \qquad (3.14)$$

where $[T']$ = transformation matrix of order (n) x (c+ℓ+k)

The property matrices of equation (3.2) are transformed, giving the following reduced mass $[\mu]$ and stiffness $[\Omega]$ matrices.

$$[\mu] = [T']^T [m] [T'] = \begin{bmatrix} [\mu_{cc}] & [\mu]_{c\ell} & [\mu_{ck}] \\ [\mu_{\ell c}] & [\mu]_{\ell\ell} & [\mu_{\ell k}] \\ [\mu_{kc}] & [\mu]_{k\ell} & [\mu_{kk}] \end{bmatrix} \qquad (3.15)$$

where $[\mu_{cc}] = [m_{cc}] + [\Phi_{ic}^T] [m_{ii}] [\Phi_{ic}]$

$$[\mu_{\ell c}] = [\mu]_{c\ell}^T = [Q_\ell]^T [m_{ii}] [\Phi_{ic}]$$

$$[\mu_{kc}] = [\mu_{ck}]^T = [\Phi_k]^T [m_{ii}] [\Phi_{ic}]$$

$$[\mu]_{\ell\ell} = [Q_\ell]^T [m_{ii}] [Q_\ell]$$

$$= [I]_{\ell\ell} = \text{identity matrix of order } \ell$$

$$[\mu]_{k\ell} = [\mu_{\ell k}]^T = [0]_{k\ell}$$

$$[\mu_{kk}] = [\Phi_k]^T [m_{ii}] [\Phi_k]$$

$$= [I_{kk}] = \text{identity matrix of order k}$$

$$[\Omega] = [T']^T [k] [T'] = \begin{bmatrix} [\Omega_{cc}] & [\Omega]_{c\ell} & [\Omega_{ck}] \\ [\Omega_{\ell c}] & [\Omega]_{\ell\ell} & [\Omega_{\ell k}] \\ [\Omega_{kc}] & [\Omega]_{k\ell} & [\Omega_{kk}] \end{bmatrix} \qquad (3.16)$$

where $[\Omega_{cc}] = [k_{cc}] + [k_{ci}] [\Phi_{ic}] = [k_{cc}] - [k_{ci}] [k_{ii}]^{-1} [k_{ic}]$

$$[\Omega_{\ell c}] = [\Omega]_{c\ell}^T = [0_{\ell c}]$$

$$[\Omega_{kc}] = [\Omega_{ck}]^T = [0_{kc}]$$

$$[\Omega]_{\ell\ell} = [Q_\ell]^T [k_{ii}] [Q_\ell]$$

$$[\Omega]_{k\ell} = [\Omega_{\ell k}]^T = [\Phi_k]^T [k_{ii}] [Q_\ell]$$

$[\Omega_{kk}] = [\Phi_k]^T [k_{ii}] [\Phi_k]$ = diagonal matrix, having the squares of natural frequencies in its diagonal.

The rest of the procedure is similar to what was explained in the previous section.

3.3) Free-Interface Boundary Flexibility Vector Method of CMS:

Two formulations are presented; formulation based only on free-interface boundary flexibility vectors and a hybrid

free-interface boundary flexibility vector/conventional CMS formulation.

3.3.1) Boundary Flexibility Vector Formulation:

Analogous to conventional method of free-interface CMS, two cases have to be considered separately; systems in which components have rigid-body modes and systems in which components have no rigid-body modes.

3.3.1.1) Components Having No Rigid-Body Modes:

The equation of motion for an undamped free-interface component in physical coordinates is

$$\begin{bmatrix} [m_{aa}] & [0_{aw}] \\ [0_{wa}] & [m_{ww}] \end{bmatrix} \begin{Bmatrix} \ddot{u}_a \\ \ddot{u}_w \end{Bmatrix} + \begin{bmatrix} [k_{aa}] & [k_{aw}] \\ [k_{wa}] & [k_{ww}] \end{bmatrix} \begin{Bmatrix} u_a \\ u_w \end{Bmatrix} = \begin{Bmatrix} f_a \\ 0_w \end{Bmatrix} \qquad (3.17)$$

where $\{u_a\}$ = subset of total displacement vector $\{u\}$ for which internal or external forces are applied.

$\{u_w\}$ = complement of $\{u_a\}$ in $\{u\}$.

$\{f_a\}$ = vector of external and internal forces acting on component.

Without a loss of generality, the case of free vibration of a component will be considered. The method that will be presented in this section can also be applied to the case of components having external loads, as will be explained in another section. In the case of free vibration, there are only internal forces acting at the interfaces between components. Thus $\{u_a\}$ is equal to $\{u_c\}$ and equation (3.18) will be

$$\begin{bmatrix} [m_{cc}] & [0_{ci}] \\ [0_{ic}] & [m_{ii}] \end{bmatrix} \begin{Bmatrix} \ddot{u}_c \\ \ddot{u}_i \end{Bmatrix} + \begin{bmatrix} [k_{cc}] & [k_{ci}] \\ [k_{ic}] & [k_{ii}] \end{bmatrix} \begin{Bmatrix} u_c \\ u_i \end{Bmatrix} = \begin{Bmatrix} f_c \\ 0_i \end{Bmatrix} \qquad (3.18)$$

where $\{f_c\}$ = subvector of internal forces acting on the interfaces of component.

The static response of the component to any individual force (or moment) in $\{f_c\}$ acting at one of the degrees of freedom of the interface $\{u_c\}$, is a multiple of its response to a unit force (or moment) applied at this degree of freedom. According to the previous chapter, the static responses of the free-interface component to a set of unit forces (or moments) applied individually and successively on every D.O.F. in $\{u_c\}$, were defined as the attachment modes $[\Phi_a]$. The number of attachment modes in the component are equal to the number of the degrees of freedom in $\{u_c\}$, where the unit forces are applied. Hence the dimension of $[\Phi_a]$ is (n x c) and it contains all the attachment modes $\{\phi_a^j\}$ in its columns, where j = 1, 2,.....c. The static response of the component to the vector $\{f_c\}$ is a linear combination of the attachment modes in $[\Phi_a]$. According to the definition of attachment modes, they are the columns of the flexibility matrix ($[g] = [k]^{-1}$) corresponding to the D.O.F. in $\{u_a\}$ (or $\{u_c\}$ in case of free vibration). Hence

$$[\Phi_a] = \begin{bmatrix} [\Phi_a]_{cc} \\ [\Phi_a]_{ic} \end{bmatrix} = \begin{bmatrix} [g_{cc}] \\ [g_{ic}] \end{bmatrix} \qquad (3.19)$$

The free-interface boundary flexibility Ritz vectors are derived from the force vector $\{f_c\}$. They are obtained from the displacement shapes of the static responses of the component to the forces (or moments) in $\{f_c\}$. The j^{th} starting boundary flexibility vector, $\{q^*_{1j}\}$ is obtained from the inertial loading of the static response $\{\phi^j_a\}$, as follows:

$$[k]\ \{q^*_{1j}\} = [m]\ \{\phi^j_a\} \qquad (3.20)$$

Note that the j^{th} starting vector is not taken as $\{\phi^j_a\}$, which is the static response to the applied unit load at interface D.O.F. number j. This approach is different than Wilson's method, where the starting vector is taken as the static response to the external applied load. This approach is also different than that presented in fixed-interface boundary flexibility method of CMS, where the starting vector was taken as the static response to the load derived in equation (3.9). The reasons for this approach will be clarified in a subsequent section of the thesis.

After obtaining the j^{th} starting vector, subsequent vectors are also obtained in a somewhat different procedure than that of the fixed-interface algorithm. The difference is that all the normalizations and orthogonalizations of vectors are performed with respect to the stiffness matrix instead of the mass matrix. The reason for this difference will also be clarified in a subsequent section of the thesis. It should be noted that any obtained vector in the j^{th} set is orthogonalized with respect to the stiffness matrix, with all previous vectors obtained in sets 1

through j.

In a paper presented by Hintz [17], it was pointed out that in order to obtain a good representation of components in free-interface CMS, an accurate determination of displacements at D.O.F. in $\{u_a\}$ (or $\{u_c\}$ in free vibration), where forces are applied, is required. In other words, the static displacement response of the component to interface forces must be the same for both the CMS model and an acceptable finite-element model of the component. In this case, the static representation of the interface D.O.F. is said to be complete, in the new system of generalized coordinates. In order to obtain the same displacement response, for both the CMS and finite-element models, the representation of the columns of the flexibility matrix corresponding to the D.O.F. in $\{u_a\}$ must be complete for the CMS model. If the complete set of free-interface normal modes is used as displacement shapes for the component, then the representation of the flexibility of interface D.O.F., is by definition, complete. Since a truncated set of the free-interface normal modes is generally used in conventional CMS, the flexibility of the component is generally less than that of the corresponding finite-element model. That is, the contribution of the truncated set of free-interface normal modes to the flexibility matrix is incomplete, and the free-interface normal modes need to be supplemented with static displacement shapes. The static displacement shapes used in conventional CMS are the residual

flexibility modes $[\Psi_d]$, defined before in chapter two (see references [2], [5], [6], and [17]). The residual flexibility modes $[\Psi_d]$ were introduced by MacNeal [5] to replace attachment modes $[\Phi_a]$, as static displacement shapes for components. The reason for introducing the residual flexibility modes is that by using attachment modes $[\Phi_a]$ to supplement the truncated set of free-interface normal modes, the obtained flexibility representation of the interface D.O.F. in the CMS model is greater than that of the finite-element model. The more flexible representation is attributed to the complete flexibility representation of the interface D.O.F., provided by using the attachment modes as the only displacement shapes for components. This fact explains why the static responses to the interface forces (which are the attachment modes), were not used as the starting boundary flexibility Ritz vectors. In the event they were utilized as starting vectors, the obtained interface flexibility of a component would actually be greater than that of the corresponding finite-element model.

In the free-interface boundary flexibility method of CMS, the contribution of the obtained boundary flexibility vectors to the interface flexibility is also incomplete. To obtain a complete flexibility, the contribution of the free-interface boundary flexibility vectors to the flexibility matrix must be determined. Assume the number of the obtained boundary flexibility vectors is ℓ. Let $[Q_\ell]$ be the matrix containing the ℓ boundary flexibility

vectors in its columns and let $\{p_\ell\}$ be the vector of generalized coordinates associated with the free-interface boundary flexibility vectors. The sizes of $[Q_\ell]$ and $\{p_\ell\}$ are $(n \times \ell)$ and (ℓ) respectively. Consider the static equilibrium equation of the component

$$[k]\ \{u\} = \begin{Bmatrix} f_c \\ 0_i \end{Bmatrix} \tag{3.21}$$

where $[k]$ = total stiffness matrix of component.

$\{u\}$ = total displacement vector.

$\begin{Bmatrix} f_c \\ 0_i \end{Bmatrix}$ = force vector containing only forces at interface degrees of freedom.

The boundary flexibility vectors $[Q_\ell]$ are used to transform the physical coordinates $\{u\}$ to the reduced set of generalized coordinates $\{p_\ell\}$, according to the following

$$\{u\} = [Q_\ell]\ \{p_\ell\} \tag{3.22}$$

Substituting equation (3.22) in equation (3.21) and pre-multiplying by $[Q_\ell]^T$, then

$$[Q_\ell]^T\ [k]\ [Q_\ell]\ \{p_\ell\} = [Q_\ell]^T \begin{Bmatrix} f_c \\ 0_i \end{Bmatrix} \tag{3.23}$$

From which

$$\{p_\ell\} = \left[\ [Q_\ell]^T\ [k]\ [Q_\ell]\ \right]^{-1}\ [Q_\ell]^T \begin{Bmatrix} f_c \\ 0_i \end{Bmatrix} \tag{3.24}$$

Substituting equation (3.24) in equation (3.22)

$$\{u\} = [Q_\ell] \left[[Q_\ell]^T [k] [Q_\ell] \right]^{-1} [Q_\ell]^T \begin{Bmatrix} f_c \\ 0_i \end{Bmatrix} \tag{3.25}$$

The static response, $\{u\}$, to the applied force vector $\begin{Bmatrix} f_c \\ 0_i \end{Bmatrix}$ is equal to

$$\{u\} = [g_k] \begin{Bmatrix} f_c \\ 0_i \end{Bmatrix} \tag{3.26}$$

where $[g_k]$ = flexibility matrix of component represented by boundary flexibility vectors.

Hence by comparing equation (3.25) to equation (3.26), the contribution of the free-interface boundary flexibility vectors to the flexibility matrix is

$$[g_k] = [Q_\ell] \left[[Q_\ell]^T [k] [Q_\ell] \right]^{-1} [Q_\ell]^T \tag{3.27}$$

For free-interface boundary flexibility vectors which are orthogonalized and normalized with respect to the stiffness matrix, $[g_k]$ becomes

$$[g_k] = [Q_\ell] [Q_\ell]^T \tag{3.28}$$

This relationship explains why the generated boundary flexibility vectors were orthogonalized and normalized with respect to the stiffness matrix [k] instead of the mass matrix [m]. Notice that the indicated inversion of the matrix in equation (3.27) is avoided by the orthogonalization and normalization with respect to the [k] matrix. The unrepresented flexibility, or residual

flexibility $[g_d]$, analogous to that of conventional free-interface CMS, is given by

$$[g_d] = [g] - [g_k] \tag{3.29}$$

where $[g]$ = full flexibility matrix = $[k]^{-1}$

$[g_k]$ is given by equation (3.28)

The residual flexibility matrix $[g_d]$ is used to obtain the residual attachment modes (residual flexibility modes) $[\Psi_d]$, by applying unit forces or moments at D.O.F. in $\{u_a\}$ (or $\{u_c\}$ in free vibration). Hence,

$$[\Psi_d] = \begin{bmatrix} [\Psi_d]_{cc} \\ [\Psi_d]_{ic} \end{bmatrix} = [g_d] \begin{bmatrix} [I_{cc}] \\ [0_{ic}] \end{bmatrix} = \begin{bmatrix} [g_d]_{cc} \\ [g_d]_{ic} \end{bmatrix} \tag{3.30}$$

where $[I_{cc}]$ = identity matrix of order c.

= matrix of unit forces or moments applied at D.O.F. in $\{u_c\}$.

Notice that the residual attachment modes $[\Psi_d]$ are the columns of the residual flexibility matrix $[g_d]$, corresponding to the degrees of freedom in $\{u_c\}$.

The boundary flexibility vectors $[Q_\ell]$ and the residual attachment modes $[\Psi_d]$, will be used to transform the equation of motion from the physical system of coordinates to the reduced generalized one. $[Q_\ell]$ and $[\Psi_d]$ will provide the complete flexibility for the D.O.F. in $\{u_a\}$ (or $\{u_c\}$ in free vibration), where forces are applied on the component. Let $\{p_d\}$ and $\{p_\ell\}$ be the generalized coordinates associated with $[\Psi_d]$ and $[Q_\ell]$

respectively. The sizes of $\{p_d\}$ and $\{p_\ell\}$ are c and ℓ respectively. The transformation of coordinates is given by

$$\begin{Bmatrix} u_c \\ u_i \end{Bmatrix} = [T_1] \begin{Bmatrix} p_d \\ p_\ell \end{Bmatrix} = \begin{bmatrix} [\Psi_d]_{cc} & [Q_c]_\ell \\ [\Psi_d]_{ic} & [Q_i]_\ell \end{bmatrix} \begin{Bmatrix} p_d \\ p_\ell \end{Bmatrix} \qquad (3.31)$$

where $[T_1]$ = transformation matrix of order (n) x (c+ℓ).

$[T_1]$ can be used to transform the property matrices of component. However coupling of components, whose interface displacements are expressed by a set of generalized coordinates and mode shapes, is not straightforward as in fixed-interface boundary flexibility method of CMS. To overcome this problem, the displacements of the interface D.O.F. are back transformed to physical D.O.F. (see Abdallah [2]), in order to allow for a direct stiffness assembly process. Assume the total number of generalized coordinates is j, and if c is the number of interface D.O.F., and m = j - c, then let

$\{p_j\}$ = vector of all generalized coordinates = $\begin{Bmatrix} p_d \\ p_\ell \end{Bmatrix}$

$\{p_c\}$ = any subset of $\{p_j\}$ having a size of c.

$\{p_m\}$ = complement of $\{p_c\}$ in $\{p_j\}$.

Accordingly, the transformation matrix $[T_1]$ in equation (3.31), is partitioned and rewritten as follows

$$\begin{Bmatrix} u_c \\ u_i \end{Bmatrix} = [T_1] \begin{Bmatrix} p_c \\ p_m \end{Bmatrix} = \begin{bmatrix} [T_1]_{cc} & [T_1]_{cm} \\ [T_1]_{ic} & [T_1]_{im} \end{bmatrix} \begin{Bmatrix} p_c \\ p_m \end{Bmatrix} \qquad (3.32)$$

From the first matrix equation in (3.32)

$$\{p_c\} = [T_1]_{cc}^{-1} \{u_c\} - [T_1]_{cc}^{-1} [T_1]_{cm} \{p_m\} \tag{3.33}$$

And from the second matrix equation in (3.32)

$$\{u_i\} = [T_1]_{ic} \{p_c\} + [T_1]_{im} \{p_m\} \tag{3.34}$$

Substitute equation (3.33) in equation (3.34), then

$$\{u_i\} = [T_1]_{ic} [T_1]_{cc}^{-1} \{u_c\} + ([T_1]_{im} - [T_1]_{ic} [T_1]_{cc}^{-1} [T_1]_{cm}) \{p_m\} \tag{3.35}$$

By using equation (3.35), the transformation of coordinates in its final form is as follows

$$\begin{Bmatrix} u_c \\ u_i \end{Bmatrix} = [T_2] \begin{Bmatrix} u_c \\ p_m \end{Bmatrix} = \begin{bmatrix} [I]_{cc} & [0]_{cm} \\ [\Phi^*]_{ic} & [Q^*]_{im} \end{bmatrix} \begin{Bmatrix} u_c \\ p_m \end{Bmatrix} \tag{3.36}$$

where $[T_2]$ = transformation matrix of size (n) x (j)

$$[\Phi^*]_{ic} = [T_1]_{ic} [T_1]_{cc}^{-1}$$

$$[Q^*]_{im} = [T_1]_{im} - [T_1]_{ic} [T_1]_{cc}^{-1} [T_1]_{cm}$$

Notice that the form of equation (3.36) is similar to that of equation (3.11), for fixed-interface components. The property matrices are transformed to the reduced coordinate system by using the transformation matrix $[T_2]$, of equation (3.36). As the interface D.O.F. are defined with respect to the physical coordinate system, then the assembly of system property matrices is straightforward and similar to that performed for fixed-interface components, in the previous section.

3.3.1.2) Components Having Rigid-Body Modes:

The main complication of components having rigid-body modes is that the stiffness matrix is singular; thus the flexibility matrix cannot be directly obtained. This problem was investigated by Rubin [6] in conventional CMS. He introduced an alternative method for obtaining the residual attachment modes. To overcome the singularity of the component stiffness matrix, he used inertia relief loading (see references [2], [4], [6] and [7] for a detailed explanation of the method). The results obtained by Rubin [6] are stated briefly here.

Assume the number of rigid-body modes is r. Let $[\Phi_r]$ be the matrix containing the r rigid-body modes in its columns. The rigid-body modes are normalized with respect to the mass matrix. Let $\{u\}$ be the vector of all physical D.O.F. in component and let $\{u_a\}$ be a subset of $\{u\}$ where forces are to be applied to obtain attachment modes. In Rubin's method, the component is constrained and prevented from rigid-body motion by applying r constraints to a set $\{u_r\}$ of D.O.F., such that $\{u_r\}$ is exclusive of $\{u_a\}$. The sizes of $\{u\}$, $\{u_a\}$ and $\{u_r\}$ are n, a and r respectively. Let $\{u_w\}$ be the complement of $\{u_a\}$ and $\{u_r\}$ in $\{u\}$, having a size of w. Define a special flexibility matrix $[g_c]$ relative to $\{u_r\}$ for the constrained component.

$$[g_c] = \begin{bmatrix} [g]_{aa} & [g]_{aw} & [0]_{ar} \\ [g_{wa}] & [g_{ww}] & [0_{wr}] \\ [0_{ra}] & [0_{rw}] & [0_{rr}] \end{bmatrix} \tag{3.37}$$

Define a square projection matrix [A] as follows

$$[A] = [I_{nn}] - [m]\,[\Phi_r]\,[\Phi_r]^T$$

where $[I_{nn}]$ = identity matrix of order n.

Rubin showed that the elastic flexibility matrix $[g_e]$ is given by

$$[g_e] = [A]^T\,[g_c]\,[A] \tag{3.38}$$

The flexibility matrix $[g_e]$ is used to obtain inertia relief attachment modes $[\Psi_a]$, by applying unit forces at $\{u_a\}$ (or $\{u_c\}$ in case of free vibration). Inertia relief attachment modes are the columns of the flexibility matrix $[g_e]$ corresponding to the degrees of freedom in $\{u_a\}$.

Similar to components having no rigid-body modes, where attachment modes $[\Phi_a]$ were used to generate the boundary flexibility vectors, the inertia relief attachment modes $[\Psi_a]$ are used here to generate the starting vectors. The j^{th} boundary flexibility starting vector $\{q_1^*\}_j$ is obtained from

$$\{q_1^*\}_j = [g_e]\,[m]\,\{\psi_a^j\} \tag{3.39}$$

where $\{\psi_a^j\}$ = j^{th} inertia relief attachment mode.

$[g_e]$ = flexibility matrix given by equation (3.38).

The subsequent boundary flexibility vectors are normalized and orthogonalized with respect to the stiffness matrix, with all

preceding vectors, as in the previous section. The contribution of the boundary flexibility vectors to the flexibility matrix is as before and given by equation (3.28). Hence the residual flexibility matrix $[g_d]$ is obtained from

$$[g_d] = [g_e] - [Q_\ell]\,[Q_\ell]^T \tag{3.40}$$

The residual flexibility matrix $[g_d]$ is used to obtain residual inertia relief attachment modes $[\Psi_s]$, by applying unit forces at $\{u_a\}$ (or $\{u_c\}$ in case of free vibration). The residual inertia relief attachment modes are the columns of the residual flexibility matrix $[g_d]$, corresponding to the D.O.F. in $\{u_a\}$. $[\Phi_r]$, $[Q_\ell]$ and $[\Psi_s]$ are used to transform the physical coordinate system to a generalized one as follows:

$$\begin{Bmatrix} u_a \\ u_w \\ u_r \end{Bmatrix} = \begin{bmatrix} [\Psi_{as}] & [Q]_{a\ell} & [\Phi_{ar}] \\ [\Psi_{ws}] & [Q]_{w\ell} & [\Phi_{wr}] \\ [\Psi_{rs}] & [Q]_{r\ell} & [\Phi_{rr}] \end{bmatrix} \begin{Bmatrix} p_s \\ p_\ell \\ p_r \end{Bmatrix} \tag{3.41}$$

where $\{p_s\}$ = generalized coordinates of $[\Psi_s]$, of size a.

$\{p_\ell\}$ = generalized coordinates of $[Q_\ell]$, of size ℓ.

$\{p_r\}$ = generalized coordinates of $[\Phi_r]$, of size r.

The size of the transformation matrix of equation (3.41) is (n) x (a + ℓ + r). The transformation of property matrices and the assembly of the system property matrices will be similar to what was presented in the previous section.

3.3.2) Hybrid Boundary Flexibility Vector/ Conventional CMS Formulation:

In this case a truncated set of free-interface normal modes $[\Phi_k]$ is used to supplement the free-interface boundary flexibility vectors, as displacement shapes for components. Let $\{p_k\}$ be the vector of generalized coordinates associated with $[\Phi_k]$. The sizes of $\{p_k\}$ and $[\Phi_k]$ are (k) and (n x k) respectively. The following steps are general regardless the component has rigid-body motion or not.

1- Normalize all the free-interface natural modes $[\Phi_k]$ with respect to the component stiffness matrix [k]. (optional step).

2- Generate the free-interface boundary flexibility vectors $[Q_\ell]$ from the starting vectors of equation (3.20), in case of components having no rigid-body modes, or equation (3.39), for components having rigid-body modes. It should be noted that every generated vector is orthogonalized with respect to the stiffness matrix [k], with all natural modes contained in $[\Phi_k]$ and with all previously generated boundary flexibility vectors. Every generated vector is normalized with respect to the stiffness matrix. Let ℓ be the number of the generated boundary flexibility vectors and let $\{p_\ell\}$ be the vector of generalized coordinates associated with them.

3- Form a matrix [V] containing the normal modes of vibration $[\Phi_k]$ and the generated boundary flexibility vectors $[Q_\ell]$, in its columns. The size of [V] is (n) x (k + ℓ). Hence,

$$[V] = \left[[\Phi_k] \; [Q_\ell] \right] \tag{3.42}$$

Accordingly the vector of total generalized coordinates, $\{p_v\}$, is formed as follows

$$\{p_v\} = \begin{Bmatrix} p_k \\ p_\ell \end{Bmatrix} \tag{3.43}$$

The contribution of the [V] matrix to the flexibility matrix $[g_k]$ is

$$[g_k] = [V] \; [V]^T \tag{3.44}$$

Or in case the natural modes of vibration $[\Phi_k]$ are not normalized with respect to the [k] matrix (step 1 above), $[g_k]$ is given by (see equation (3.27))

$$[g_k] = [V] \left[[V]^T \; [k] \; [V] \right]^{-1} [V]^T = [V] \begin{bmatrix} [\Lambda_{kk}] & [0]_{k\ell} \\ [0_{\ell k}] & [I]_{\ell\ell} \end{bmatrix}^{-1} [V]^T \tag{3.45}$$

where $[\Lambda_{kk}]$ = a diagonal square matrix of order k. It contains the eigenvalues, corresponding to natural modes $[\Phi_k]$, in its diagonal.

$[I]_{\ell\ell}$ = identity matrix of order ℓ.

Although equation (3.45) involves an inversion of a matrix, it can be used because the matrix is diagonal and its inverse can be obtained readily.

The subsequent steps, whether for components with or without rigid-body modes, are similar to that explained in the previous section, except that every $[Q_\ell]$ is replaced by [V] and every $\{p_\ell\}$

is replaced by $\{p_v\}$, in the equations.

3.4) Computational Effort for Boundary Flexibility Vectors vs Normal Modes of Vibration:

In this section, the number of operations required to generate (ℓ) boundary flexibility vectors versus (k) normal modes of vibration, is estimated. In the numerical problems of chapters four and five, the method used for obtaining the eigenvalues is the determinant search method (see Bathe [18]) with bisection iteration. While the method used to extract the mode shapes (eigenvectors) is the inverse iteration method (see Bathe [18]). The number of required operations is estimated by using some basic criteria from reference [19]. The detailed calculations are listed in appendix A, while in this section only the final results are presented here.

It was found that the number of operations required to obtain (k) eigenpairs, for a component, is as follows:

Total number of multiplications and divisions =

$$\frac{k}{3}(4n^3-n) + \sum_{i=1}^{k}\left[\frac{y_i}{6}(2n^3+9n^2-5n) + x_i(2n^2+4n+1)\right] \quad (3.46)$$

Total number of additions and subtractions =

$$\frac{k}{6}(8n^3-9n^2+n) + \sum_{i=1}^{k}\left[\frac{y_i}{3}(n^3+3n^2-n) + x_i(2n^2+n-3)\right] \quad (3.47)$$

where n = total number of physical D.O.F. in component.

x_i = number of inverse iterations required to obtain eigenvector i.

y_i = number of iterations, in determinant search method, required to obtain eigenvalue i.

While for the generation of ℓ boundary flexibility vectors per component, the number of operations is as follows:

Total number of multiplications and divisions =

$$\left(\frac{4}{3}\right) n^3 + \left(\frac{\ell^2 + 5\ell}{2}\right) n^2 + \left(\frac{3\ell^2 + 3\ell - 1}{3}\right) n \qquad (3.48)$$

Total number of additions and subtractions =

$$\left(\frac{4}{3}\right) n^3 + \left(\frac{\ell^2 + 5\ell - 3}{2}\right) n^2 + \left(\frac{3\ell^2 - 15\ell + 1}{6}\right) n - \left(\frac{\ell^2 + \ell}{2}\right) \qquad (3.49)$$

It can be seen from equations (3.46) through (3.49), that the number of operations is proportional to n^3. Thus in case of components having a large number of D.O.F., the number of operations will be governed by the n^3 terms in the above equations. By using only the highest order cubic terms in the above expressions, approximate values for the number of operations can be obtained as follows:

- For the generation of k eigenpairs:

Approximate number of multiplications and divisions =

$$\left(4 + \bar{y}\right) \left(\frac{k}{3}\right) n^3 \qquad (3.50)$$

where $\bar{y}$ = average number of iterations in determinant search method.

The approximate number of additions and subtractions is found to be the same as in equation (3.50).

- For the generation of ℓ boundary flexibility vectors:

Approximate number of multiplications and divisions =

$$\left(\frac{4}{3} \right) n^3 \qquad (3.51)$$

Also the approximate number of additions and subtractions is the same as in equation (3.51).

It is worth noting that the number of generated boundary flexibility vectors , ℓ, does not appear in equation (3.51). Hence, once the stiffness matrix is inverted to obtain the starting vector, the subsequent vectors are generated almost without any computational cost.

The ratio of the number of operations to extract k eigenpairs / number of operations to generate ℓ boundary flexibility (B.F.) vectors can be approximated by:

$$\frac{\text{Approximate \# of operations for extraction of k eigenpairs}}{\text{Approximate \# of operations for generation of } \ell \text{ B.F. vectors}} =$$

$$\left(1 + \frac{\bar{y}}{4} \right) k \qquad (3.52)$$

The approximated ratio of equation (3.52) is dependent only on k and $\bar{y}$ and it can be seen that it is always greater than unity. Hence the number of operations required to extract k eigenpairs is always greater than the number required to generate ℓ boundary flexibility vectors. Furthermore, for more complex components,

where one would require a substantial number of normal modes (k) for a realistic characterization, the ratio given in equation (3.52) will be relatively high.

3.5) Summary:

A new method of CMS namely, the boundary flexibility vector method of CMS, was presented in this chapter. The displacement modes used in the method, represent a new type of vector, named boundary flexibility Ritz vectors. Those vectors are generated from internal forces derived from the flexibility (or stiffness) properties of the interface degrees of freedom of components. It should be pointed out that the presented method avoids the solution of a costly component eigenvalue problem, which is required in conventional methods of CMS, to obtain the natural vibration modes of the components.

A comparison of the number of operations required to obtain natural modes of vibration versus the boundary flexibility Ritz vectors was made. The comparison indicated that a substantial reduction of the number of required operations was obtained by using the boundary flexibility vectors instead of natural modes in CMS.

It should be noted that, in all of the methods of fixed or free-interface boundary flexibility method of CMS outlined above, for the case where there are external loads acting at internal D.O.F. $\{u_i\}$ of components, those forces are used to obtain load-dependent Ritz vectors (as in Wilson's algorithm presented in

chapter (2)). For this case, the formulation of the boundary flexibility method of CMS will be presented in chapter five.

Chapter (4)

Results of Numerical Problems in Free Vibration

4.1) Introduction:

This chapter presents three sample problems of free vibration solved numerically by the boundary flexibility Ritz vector method of CMS (B.F. Ritz CMS). The problems are also solved by conventional CMS and finite-element methods. The results obtained by the boundary flexibility method of CMS are compared to the other methods. Also a comparison of the saving in CPU time, obtained by using the boundary flexibility vectors instead of eigenvectors, is presented in this chapter. The first two problems are composed of small components, having a few number of physical D.O.F. While the third problem is substantially larger, and demonstrates very well the saving in CPU time obtained by using the boundary flexibility method of CMS.

4.2) Sample Problem One:

Sample problem one consists of a beam fixed at both ends and supported at mid-span by a spring, discretized as shown in Figure (4.1) (all units are lbs and inches). The properties of the beam are given in the figure. Each node has two D.O.F.; vertical displacement and flexural rotation.

The eigen-problem for this problem was solved for eleven different values of spring stiffness k, ranging from 0.0 to 50000.0 lb/in. For each value of k, five mode shapes and natural

frequencies were obtained by solving the problem by nine different methods. The first solution for the problem was obtained from a finite-element model of the system. The problem was then solved using CMS by dividing the system into two identical components. One component was the beam from node 1 to node 9 and the other from node 9 to node 17. The constraint and attachment modes, of component number one, used to generate the boundary flexibility Ritz vectors are shown in Figures (4.2) and (4.3) respectively. The various fixed-interface and free-interface CMS methods, utilized in solving the problem, are listed in the following sections.

4.2.1) Solution by Fixed-interface Methods of CMS:

Four methods were used to solve the problem:

Method 1- Conventional CMS, using two fixed-interface normal modes and two constraint modes, for each component.

Method 2- Boundary flexibility CMS, using two fixed-interface boundary flexibility Ritz vectors and two constraint modes, for each component.

Method 3- Boundary flexibility CMS, using three fixed-interface boundary flexibility Ritz vectors and two constraint modes, for each component.

Method 4- Mixed conventional and boundary flexibility CMS, using one fixed-interface normal mode, one fixed-interface boundary flexibility Ritz vector and two constraint modes, for each component.

For all methods, the values of the obtained results for natural frequencies were compared to that obtained by the finite-element method. The percentage of error was computed for all the obtained frequencies, over the given range of spring stiffness. The computed percentage errors, for all methods of solution, for the first two natural frequencies were less than 0.06 % and these results are not listed. For the third, forth and fifth natural frequencies, the comparative results are shown in Figures (4.4), (4.5) and (4.6), respectively. As indicated from the figures, by increasing the spring stiffness, the % errors are generally increasing, except for mode shapes having a node at the spring location. It can be seen from the figures, that the results obtained by conventional CMS are better than those obtained by boundary flexibility CMS with two boundary flexibility Ritz vectors per component. It should be noted, however, that the third natural frequency computed by using boundary flexibility CMS, with two boundary flexibility Ritz vectors per component, are quite reasonable for engineering purposes. Moreover, a substantial saving in CPU time was attained by using two boundary flexibility Ritz vectors rather than two normal modes, for each component, as will be discussed in the next section. A substantial reduction in CPU time can even be obtained when using three boundary flexibility Ritz vectors per component, compared to two normal modes (as will be seen in the next section). The results obtained by the mixed conventional/ boundary flexibility

CMS were generally in between those obtained by conventional and boundary flexibility methods of CMS, with two boundary flexibility vectors per component. For certain values of spring stiffness, however, they were inferior to the boundary flexibility method (for the third and forth natural frequencies) and superior to conventional CMS (for the fifth natural frequency). The results obtained by boundary flexibility CMS, with three boundary flexibility Ritz vectors per component, were in better agreement with the finite-element method solution than those obtained by conventional CMS.

4.2.2) Solution by Free-Interface Methods of CMS:

Four methods were used to solve the problem:

Method 1- conventional free-interface CMS, using two free-interface normal modes and two residual attachment modes per component.

Method 2- Free-interface boundary flexibility CMS, using two free-interface boundary flexibility Ritz vectors and two residual attachment modes per component.

Method 3- Free-interface boundary flexibility CMS, using three free-interface boundary flexibility Ritz vectors and two residual attachment modes per component.

Method 4- Mixed conventional and boundary flexibility CMS, using one free-interface normal mode, one free-interface boundary flexibility Ritz vector and two residual attachment modes per component.

As before, the percentage errors of the first two natural frequencies, obtained by all four methods, are very small (less than 0.075 %) and are not listed. The comparative results for the third, forth and fifth natural frequencies are shown in Figures (4.7), (4.8) , and (4.9) respectively. Comparing the results to that obtained by fixed-interface components, it can be seen that for conventional and mixed methods of CMS, the results obtained by using fixed-interface components were generally better than that obtained by using free-interface components. While for the boundary flexibility method of CMS (whether two or three boundary flexibility vectors are used per component), the results were nearly the same for both fixed and free-interface components. It can be seen from Figures (4.7), (4.8) and (4.9) that the results obtained by conventional free-interface CMS were better than those obtained by the boundary flexibility method, with two boundary flexibility vectors per component. However, they were comparable and the difference was not as large as in fixed-interface methods. The results obtained by the mixed conventional/ boundary flexibility model were always in between those obtained by conventional and boundary flexibility (with two vectors per component) methods of CMS. The results obtained by the boundary flexibility CMS, with three boundary flexibility Ritz vectors per component, were in the best agreement with the finite-element method solution, and they were nearly equal. It can be seen from Figure (4.9) that the conventional CMS, boundary

flexibility CMS (with two vectors per component), and mixed methods gave unreasonable values for the fifth natural frequency.

CPU time was compared for the extraction of normal modes of vibration versus the generation of the boundary flexibility vectors, for the free-interface solutions. For each component, it was found that a saving of 47 % of CPU time was achieved in generating two free-interface boundary flexibility vectors rather than two free-interface normal modes. Moreover a saving of 31 % of CPU time was achieved, for the case of generating three boundary flexibility vectors versus two normal modes. It should be noted that the most costly operation in generating the boundary flexibility vectors is the inversion of the full stiffness matrix [k], in case of free-interface components (see equation (3.20)), or the stiffness matrix $[k_{ii}]$ of internal D.O.F., in case of fixed-interface components (see equation (3.10)). This inversion of the stiffness matrix, is also required in conventional free-interface CMS (see equations (3.29) and (3.30)) for the generation of residual attachment modes, or in conventional fixed-interface CMS (see equation (3.5)) for the generation of constraint modes. As this fact was not taken into consideration in computing the saving in CPU time, the above results regarding the saving in CPU time can be considered conservative.

4.3) Sample Problem Two:

This problem is used to illustrate the solution of free vibration problems by free-interface boundary flexibility vector

method of CMS, for components having rigid-body modes. The problem is similar to problem one with the support at node one being hinged instead of being fixed (see Figure (4.10)). Hence component number one has one rigid-body mode, which is the rotation of the whole component about the hinged node. For a fixed spring stiffness, the problem is solved by four methods. The first method of solution is by the finite-element method, while the other three solutions are obtained by CMS. The three methods of CMS used, are as follows:

Method 1- Conventional free-interface CMS, using one rigid-body mode, one free-interface normal mode and two residual inertia relief attachment modes for the first component. Two free-interface normal modes and two residual attachment modes were used for the second component.

Method 2- Boundary flexibility CMS, using one rigid-body mode, one free-interface boundary flexibility Ritz vector and two residual inertia relief attachment modes, for the first component. Two free-interface boundary flexibility Ritz vectors and two residual attachment modes were used for the second component.

Method 3- Boundary flexibility CMS as in method 2 above, but the number of free-interface boundary flexibility vectors, in the two components, is increased by one.

Five natural frequencies were obtained for each method. The % discrepancy of all the obtained natural frequencies, compared to

the finite-element solution, were computed for all the three methods (see table (4.1)). For conventional CMS (method 1), the results were slightly better than those obtained by boundary flexibility CMS (method 2). Both of the two methods gave unreasonable values for the fifth natural frequency. The results obtained for all five natural frequencies by boundary flexibility CMS (method 3) were in the best agreement with the finite-element method solution.

4.4) Sample Problem Three:

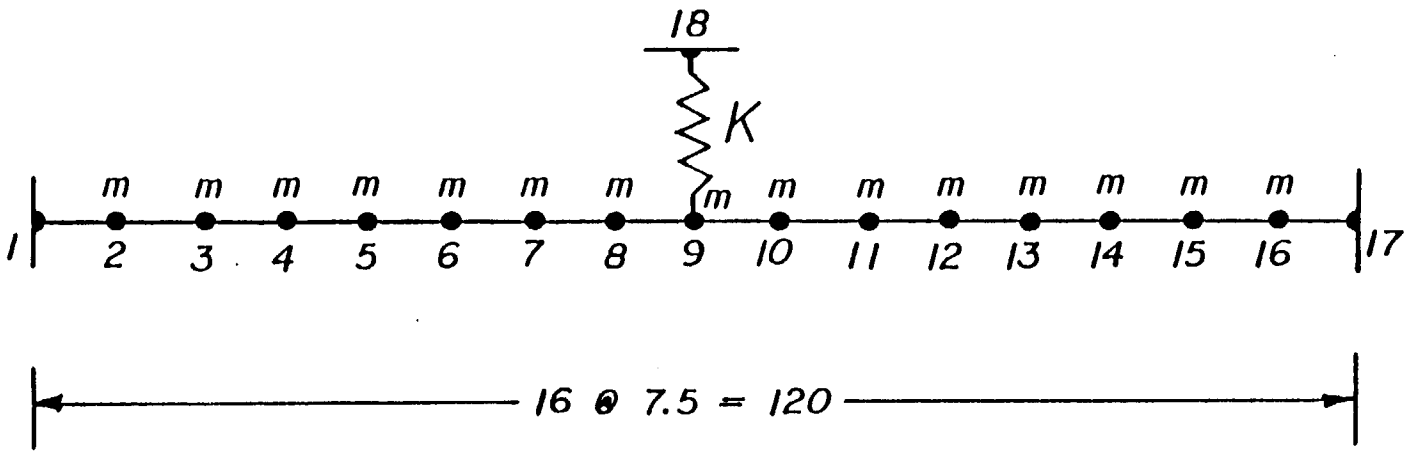
This sample problem is used to verify the accuracy of the results obtained by the boundary flexibility method of CMS, in case of components having a large number of D.O.F. The problem is also used to compute the saving in CPU time obtained by using boundary flexibility CMS instead of conventional methods.

The problem consists of a three dimensional three storey frame, as shown in Figure (4.11), fixed to the ground at four columns. The properties of the beams and columns are given in Figure (4.11). Also given is the finite-element model for the frame. The total number of physical D.O.F. for this frame is 648, corresponding to 6 D.O.F. per node. The problem was first solved by the finite-element model obtaining the first twelve natural frequencies. It was then solved by the boundary flexibility and conventional methods of CMS. The structure was divided into four components by two vertical planes; the first plane passing through nodes 29, 30, 31, 82, 83 and 84 and the second plane passing

through nodes 26, 27, 28, 85, 86 and 87. Component number one is shown in Figure (4.12). Fixed-interface components were used for the two CMS methods. For each component, 36 constraint modes were used, corresponding to six D.O.F. per node at the interfaces. For the boundary flexibility method of CMS, three boundary flexibility vectors were generated, for each component. The same number of eigenvectors were used in conventional CMS, for each component. Hence the total number of D.O.F. for the whole frame in CMS methods was reduced to 84, compared to 648 D.O.F. in the finite-element model. The results obtained for the first twelve natural frequencies, by the boundary flexibility and conventional methods of CMS were compared to that obtained by the finite-element method, and the results were presented in Table (4.2). From Table (4.2), it is clear that the results obtained by both methods of CMS, for the twelve natural frequencies, are in a very good agreement to that obtained by the finite-element method.

The percentage saving in CPU time obtained by using the boundary flexibility vectors instead of eigenvectors, was computed for each component. Figure (4.13) shows the percentage saving in CPU time, in the case where equal numbers of boundary flexibility vectors and eigenvectors were generated, up to eight boundary flexibility vectors or eigenvectors per component. Two curves are shown in Figure (4.13). The lower curve of Figure (4.13) is the saving computed for the case where the CPU time required to invert the stiffness matrix in the boundary flexibility method of CMS, is

taken into consideration. The upper curve of Figure (4.13) represents the saving when the CPU time required to invert the stiffness matrix is not taken into consideration. The reason for showing the upper curve is that the stiffness matrix inversion is also required in conventional method of CMS to generate the constraint modes. An average percentage savings in CPU time of 75% in the lower curve and 80% in the upper curve are observed. Figure (4.14) shows the percentage saving in CPU time obtained by using up to five boundary flexibility vectors versus one eigenvector. For the case where the CPU time required to invert the stiffness matrix in the boundary flexibility method is included, it can be seen from Figure (4.14) that 12% saving in CPU time can be achieved, if four boundary flexibility vectors are generated instead of one eigenvector. While for the case where the time required to invert the stiffness matrix is not included, a saving of 20% in CPU time can be achieved, even if five boundary flexibility vectors are generated instead of one eigenvector.



$m = .01554$ lb-sec^2/in
$A = 4.0$ in^2
$I = 10.0$ in^4
$E = 208000$ lb/in^2

Figure (4.1) - Sample Problem One

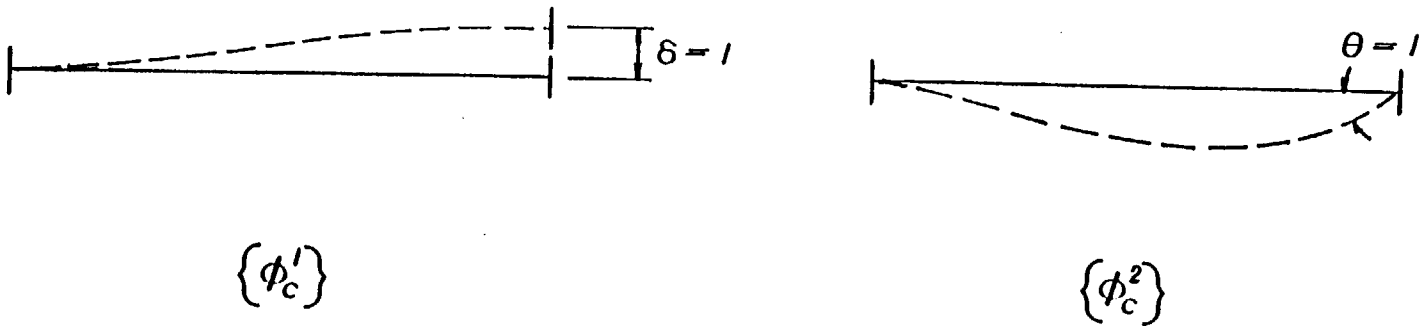


Figure (4.2) - Component Constraint Modes

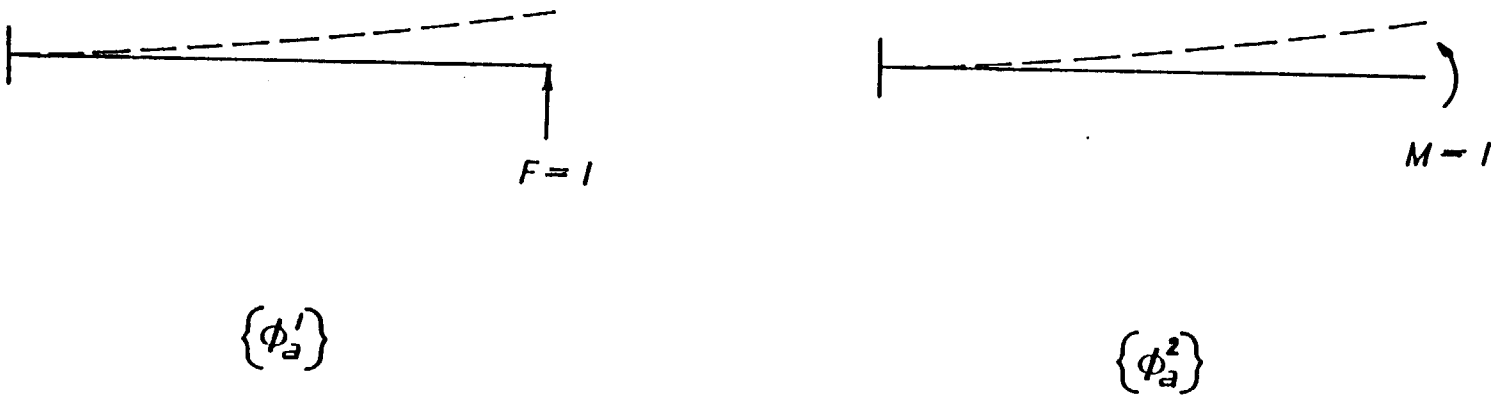


Figure (4.3) - Component Attachment Modes

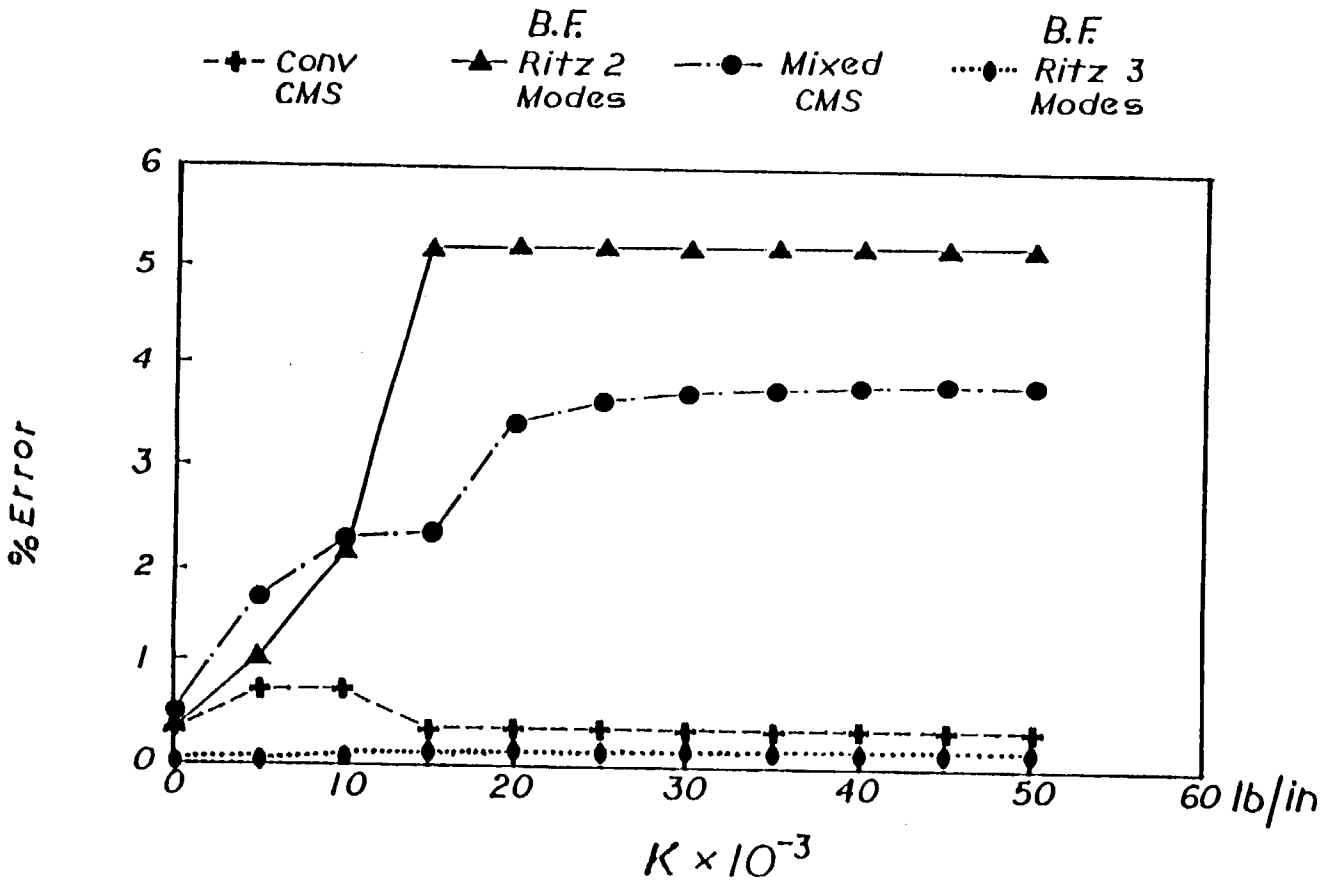


Figure (4.4) - Variation of Error in ω_3 with K for Fixed Methods

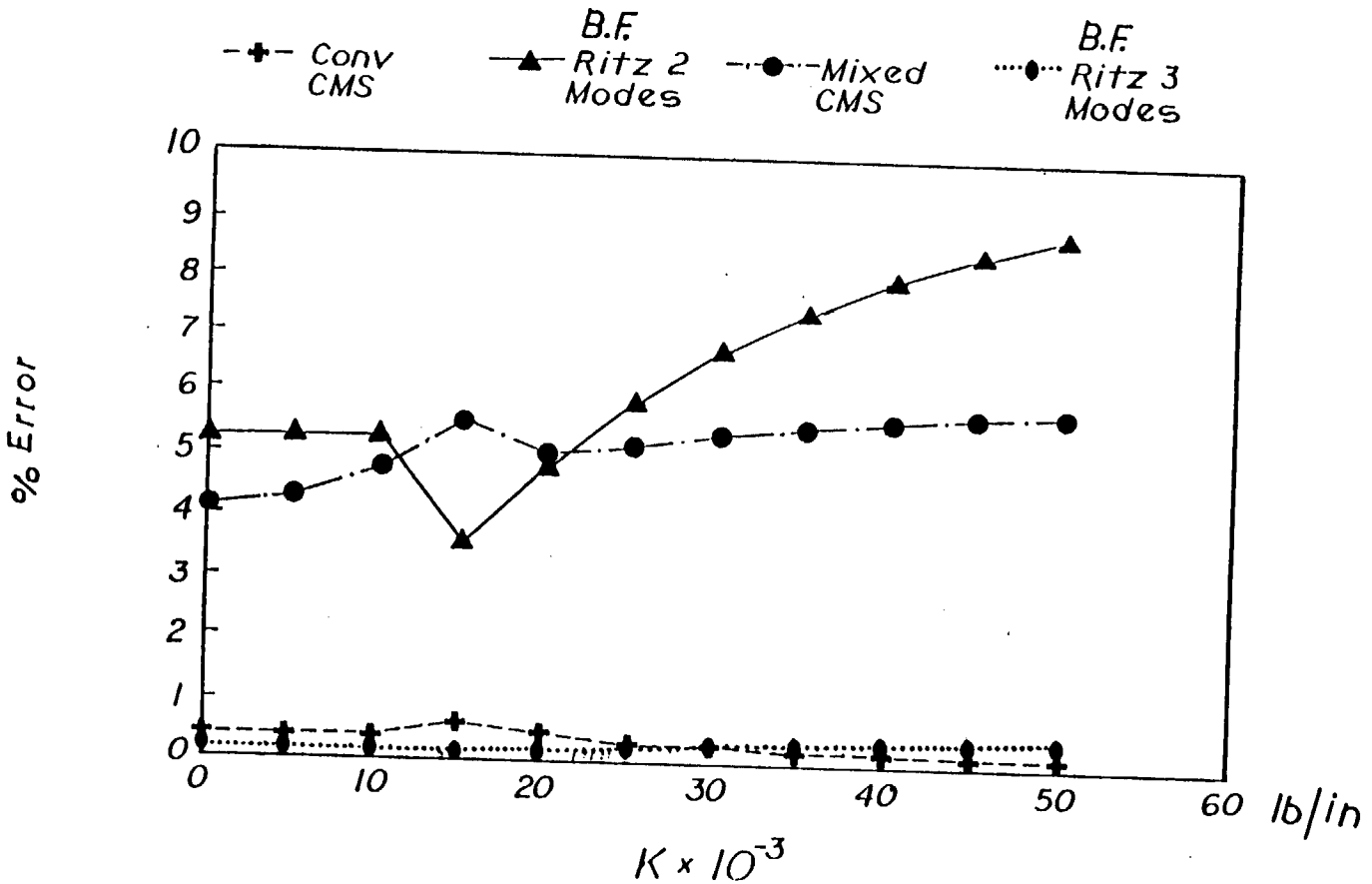


Figure (4.5) - Variation of Error in ω_4 with K for Fixed Methods

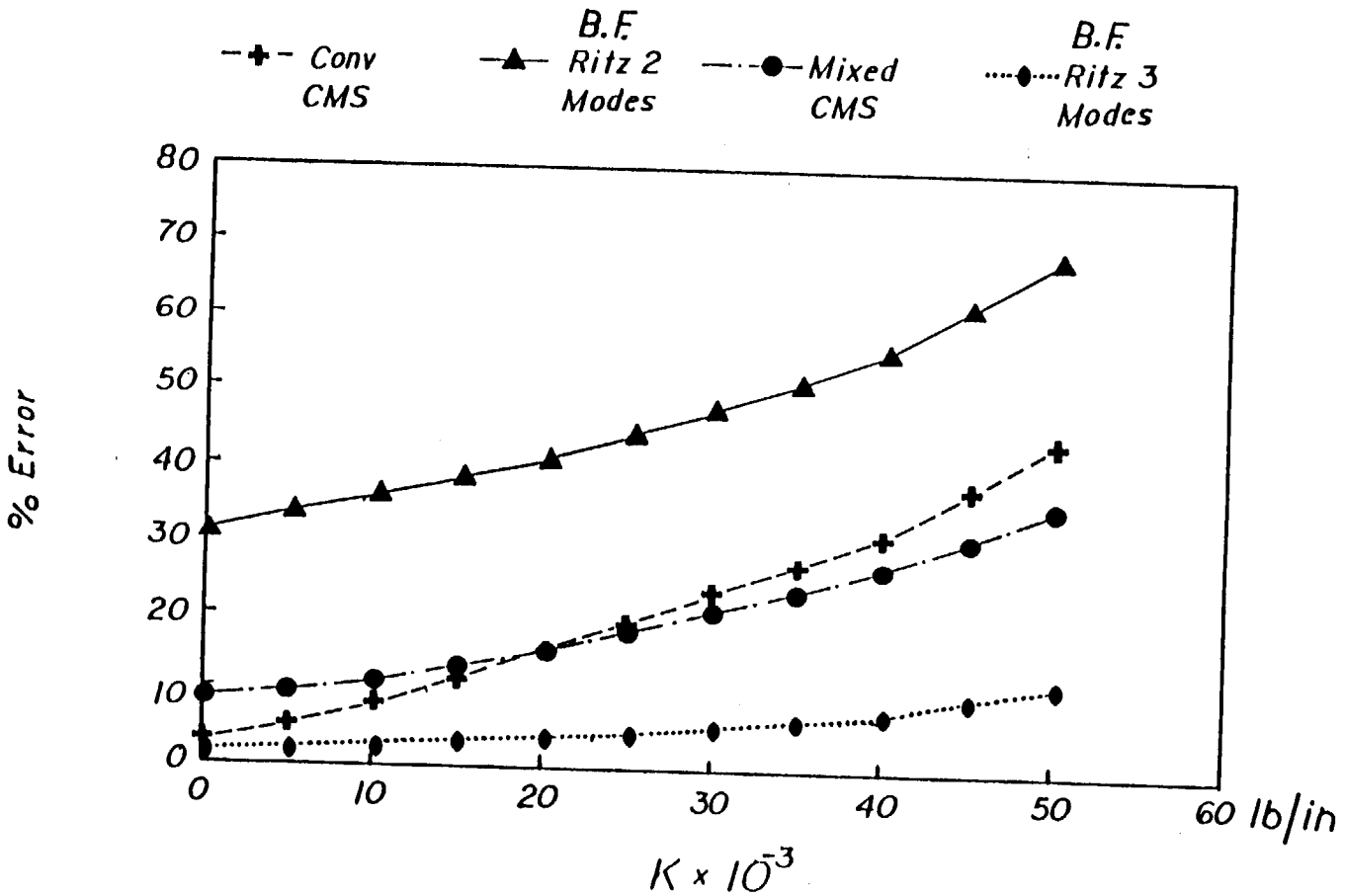


Figure (4.6) - Variation of Error in ω_s with K for Fixed Methods

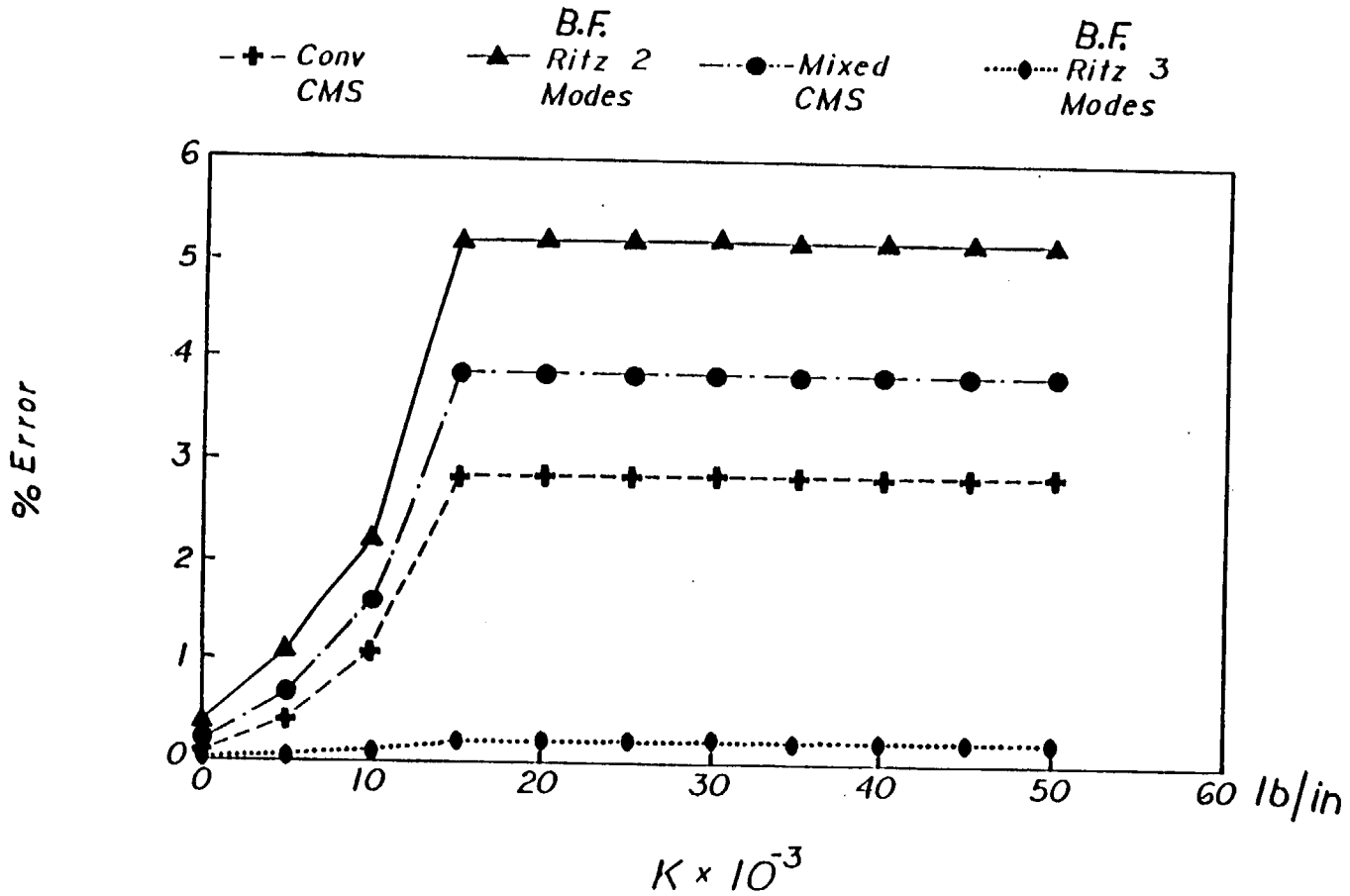


Figure (4.7) - Variation of Error in ω_3 with $\dot{K}$ for Free Methods

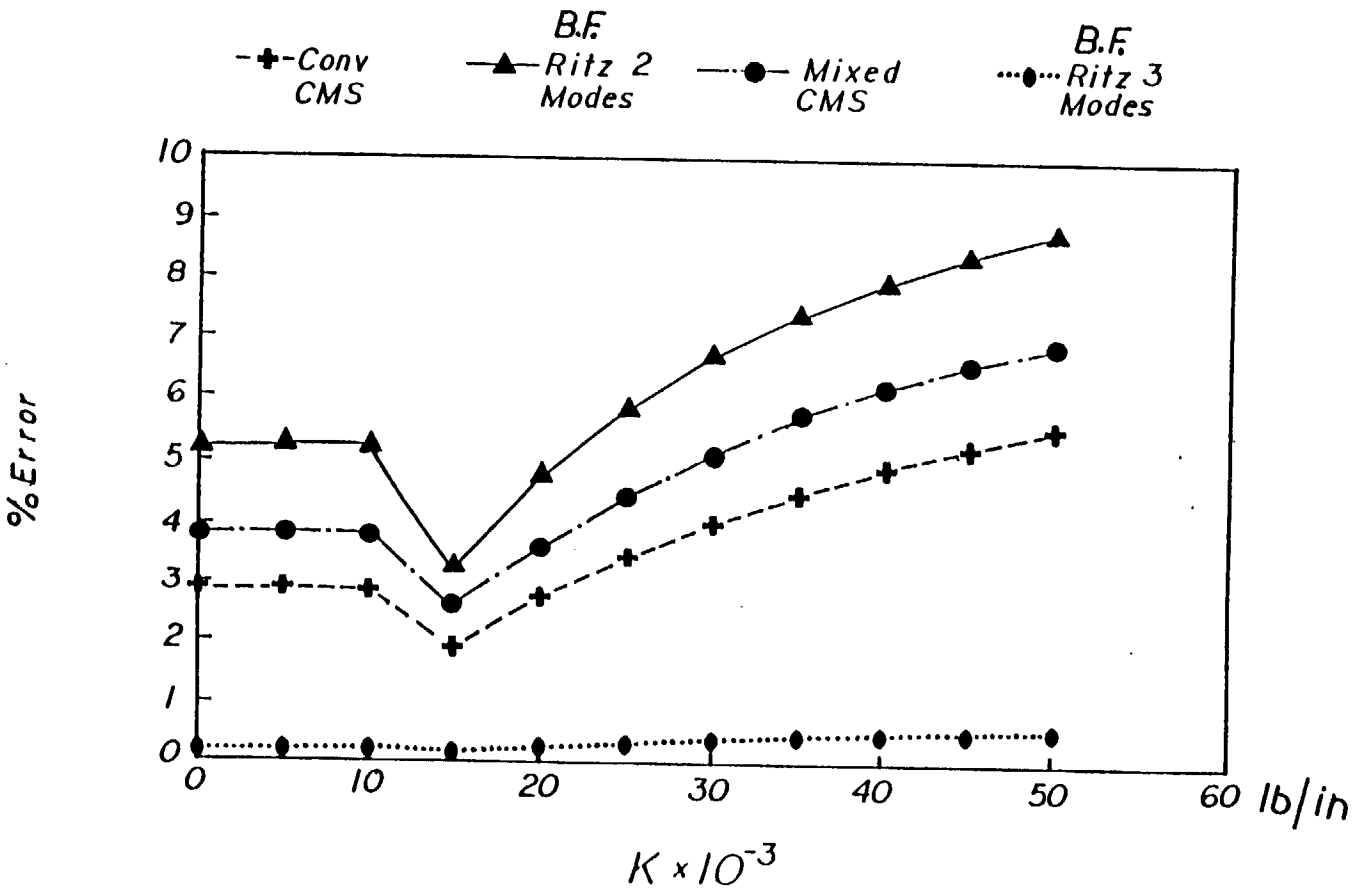


Figure (4.8) - Variation of Error in ω_4 with K for Free Methods

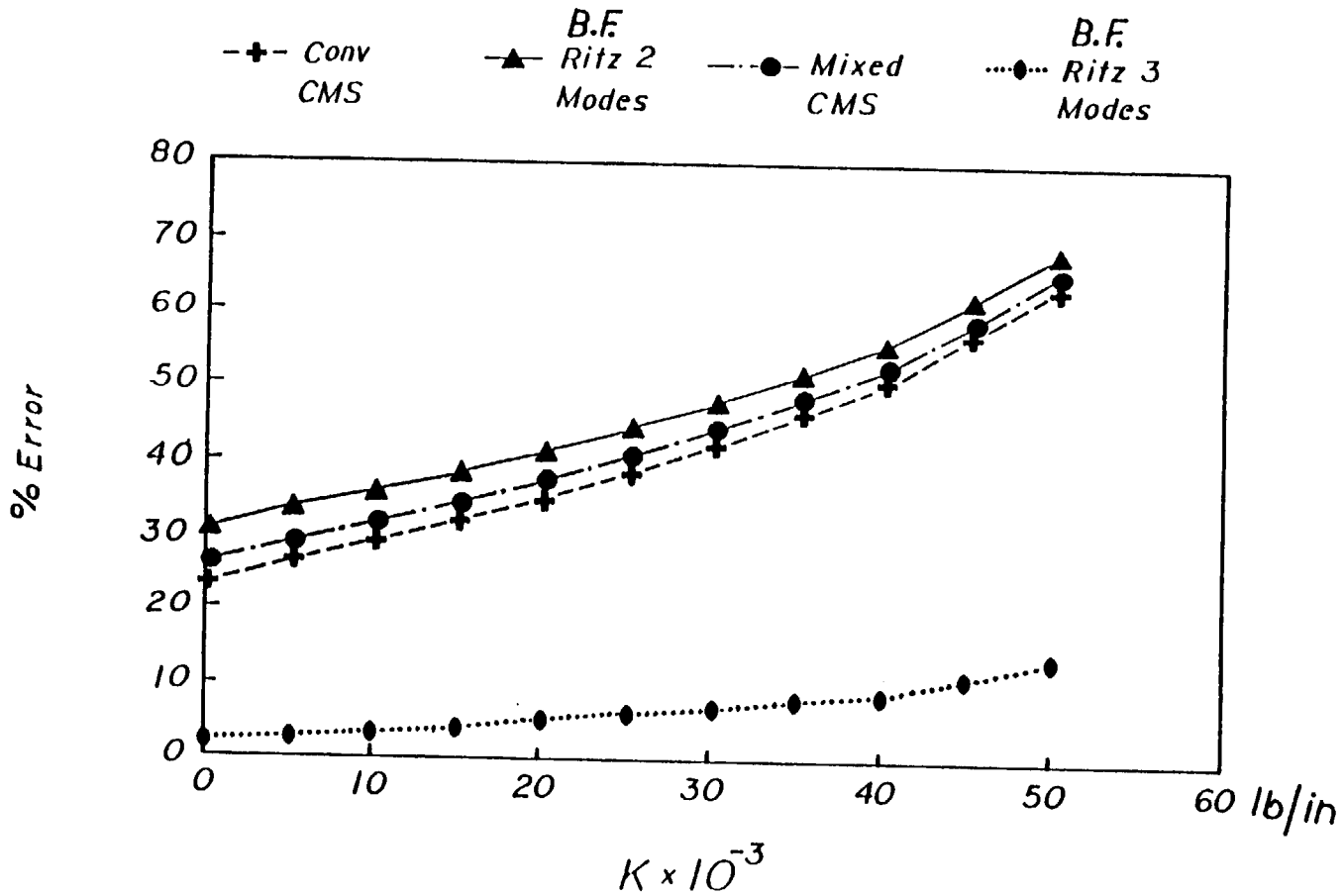
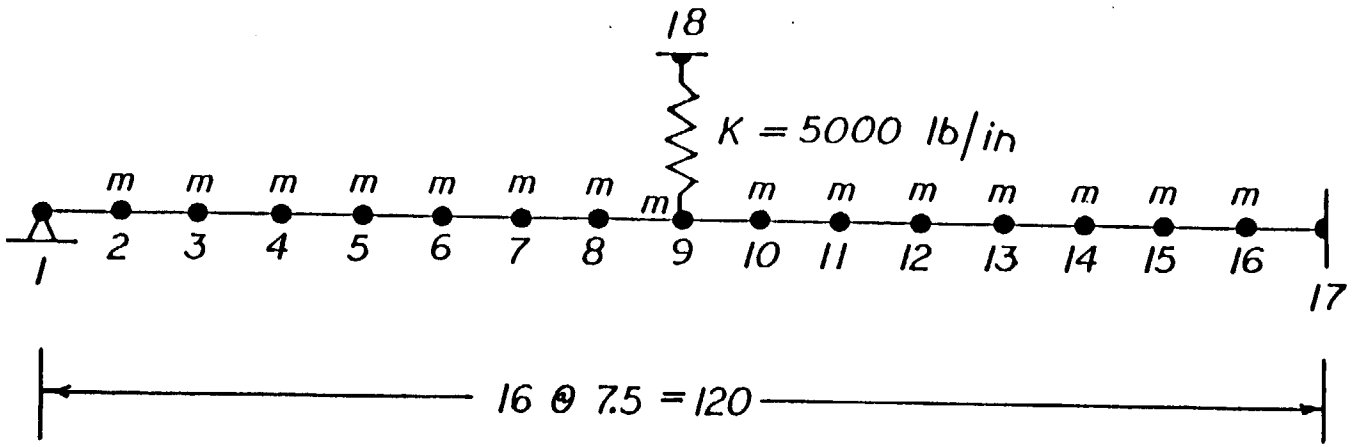


Figure (4.9) - Variation of Error in ω_s with K for Free Methods



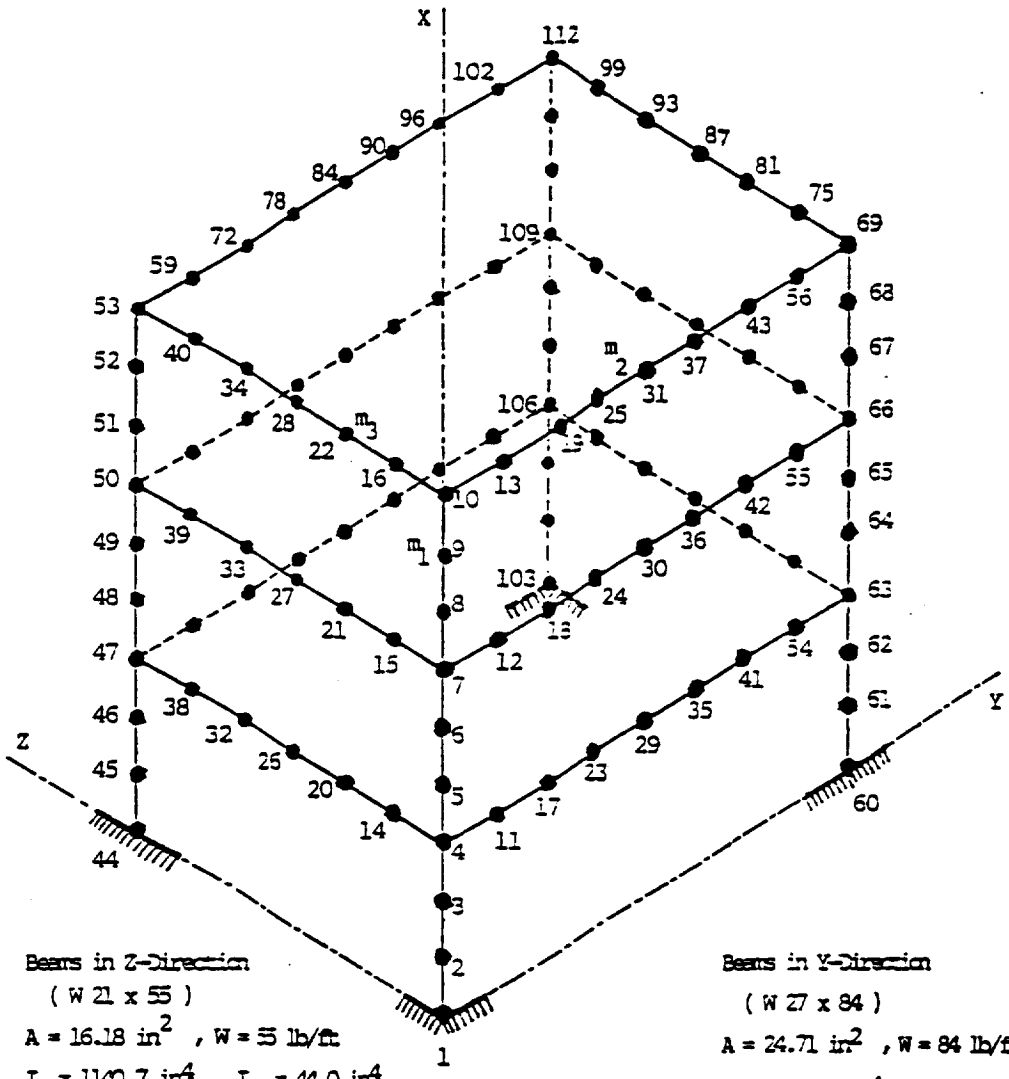
$m = .01554\ lb\text{-}sec^2/in$

$A = 4.0\ in^2$

$I = 10.0\ in^4$

$E = 208000\ lb/in^2$

Figure (4.10) - Sample Problem Two



Beams in Z-Direction
(W 21 x 55)
$A = 16.18\ in^2$, $W = 55\ lb/ft$
$I_{zz} = 1140.7\ in^4$, $I_{yy} = 44.0\ in^4$
$J = 1.1263\ in^4$
$m_3 = 0.5694\ lb\text{-}sec^2/in$

Beams in Y-Direction
(W 27 x 84)
$A = 24.71\ in^2$, $W = 84\ lb/ft$
$I_{zz} = 2224.8\ in^4$, $I_{yy} = 95.7\ in^4$
$J = 2.5497\ in^4$
$m_2 = 0.3696\ lb\text{-}sec^2/in$

Columns (W 14 x 127)
$A = 37.33\ in^2$, $W = 127\ lb/ft$
$I_{zz} = 1476.7\ in^4$, $I_{yy} = 527.6\ in^4$
$J = 10.6898\ in^4$
$m_1 = 1.3147\ lb\text{-}sec^2/in$

$E_{all} = 30 \times 10^6\ lb/in^2$

Figure (4.11) - Sample Problem Three

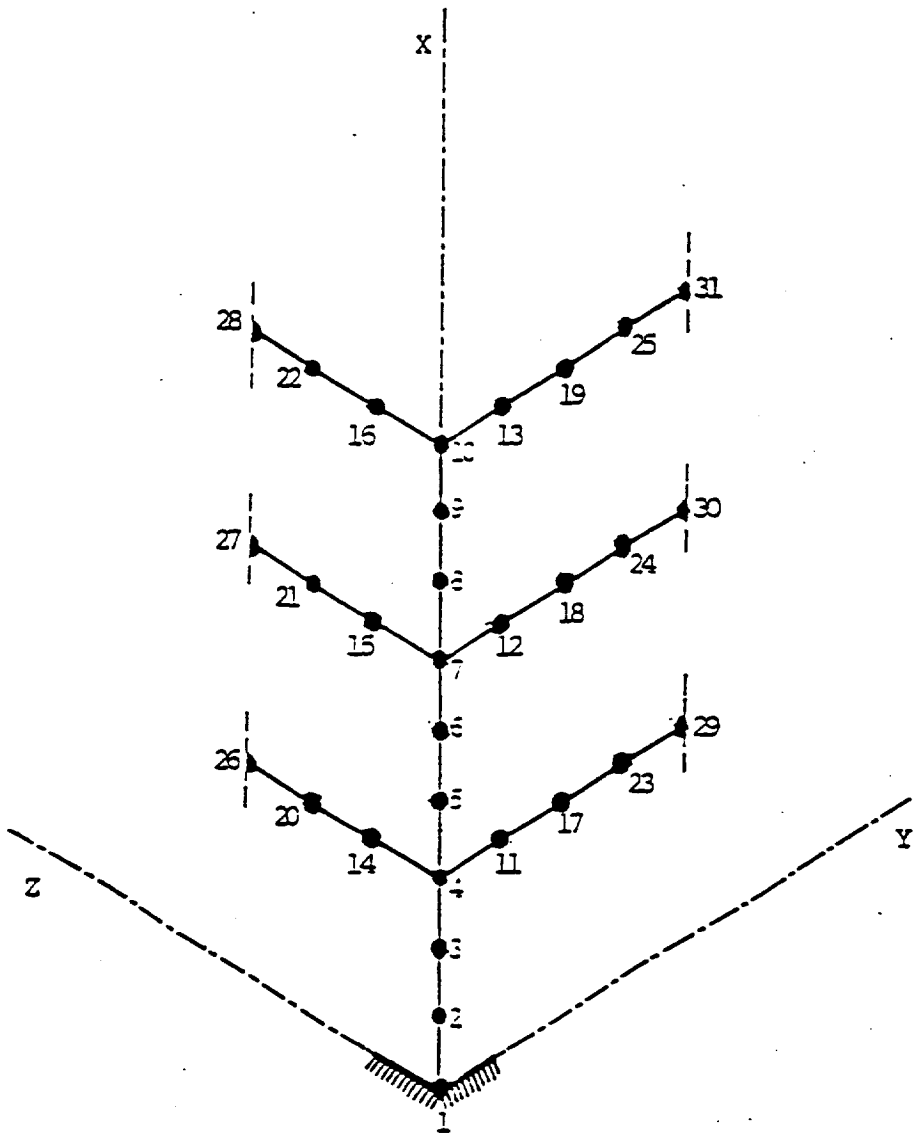


Figure (4.12) - Component Number One for Sample Problem Three

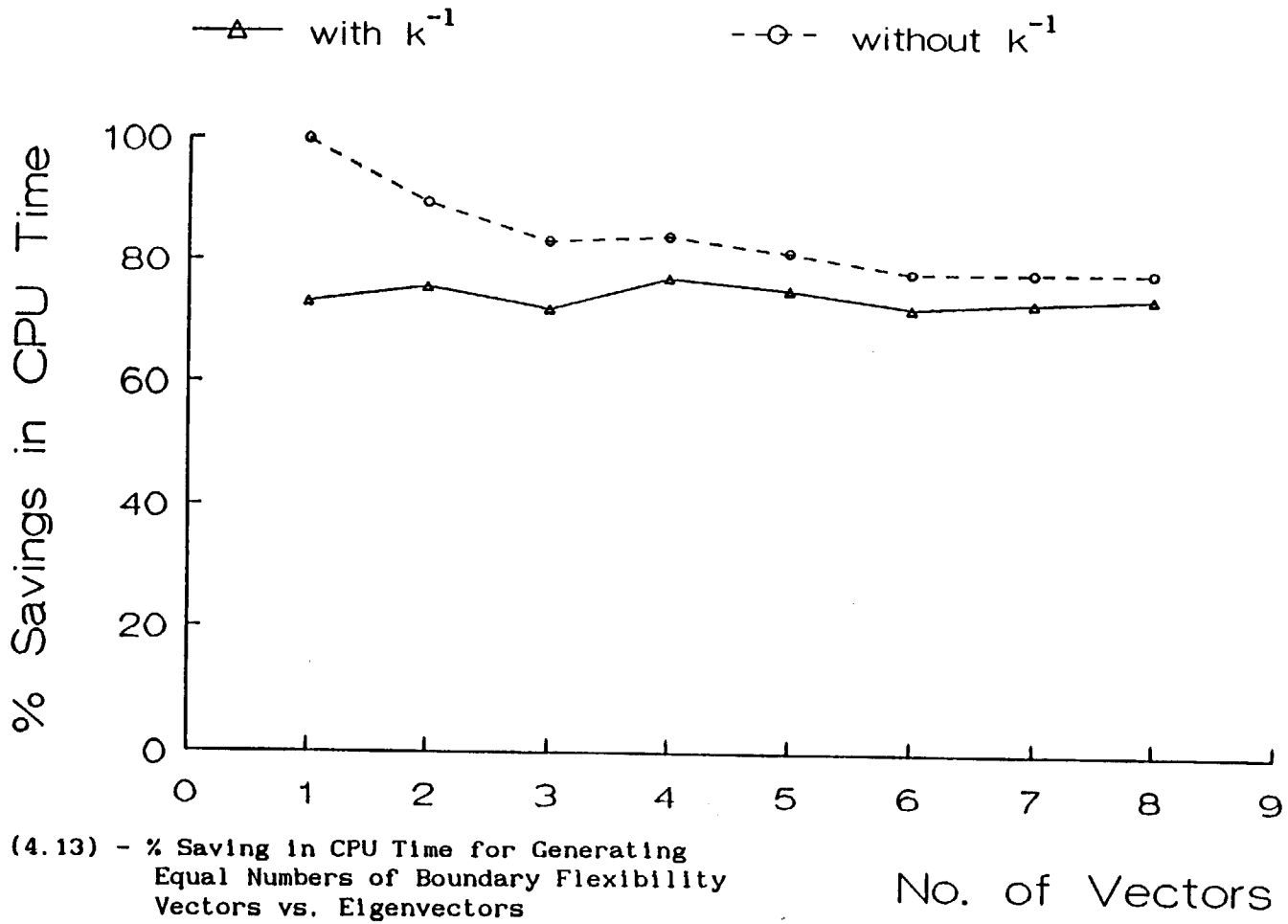


Figure (4.13) - % Saving in CPU Time for Generating Equal Numbers of Boundary Flexibility Vectors vs. Eigenvectors

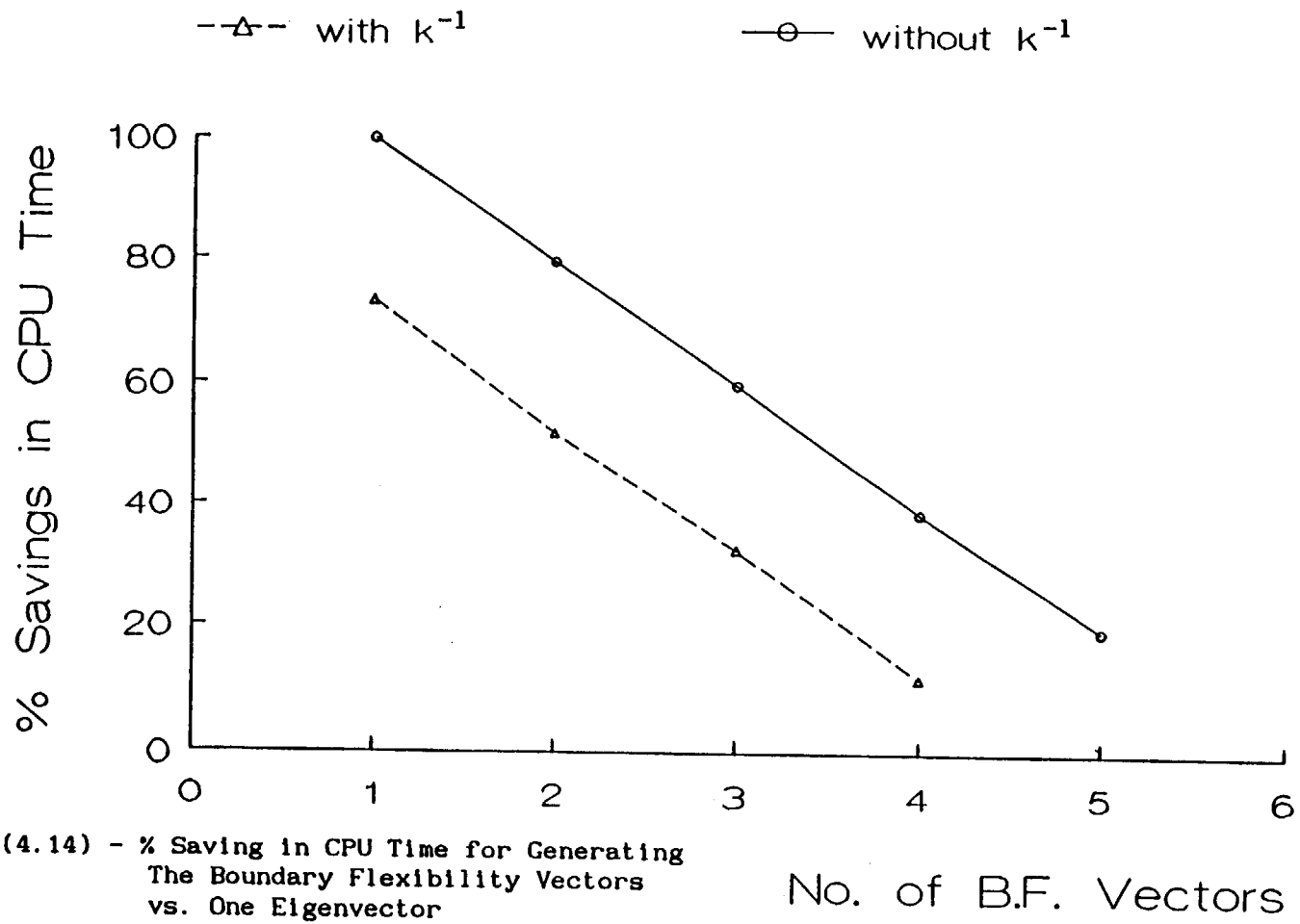


Figure (4.14) - % Saving in CPU Time for Generating The Boundary Flexibility Vectors vs. One Eigenvector

Method	ω_1	ω_2	ω_3	ω_4	ω_5
1	0.05853	0.02541	0.63544	2.39934	28.0691
2	4.24369	4.23788	1.73601	2.80158	31.4793
3	0.05878	0.02421	0.03567	0.14458	2.64257

Table (4.1) - % Error of Natural Frequencies For Sample Problem Two

Natural Frequency (rad/sec)	Method		
	FEM	Conven. CMS	B.F. CMS
ω_1	24.5962	24.6043	24.6025
ω_2	29.6600	29.6670	29.6691
ω_3	30.2068	30.2142	30.2163
ω_4	30.8600	30.8678	30.8698
ω_5	32.8940	32.9237	32.9275
ω_6	37.3674	37.3866	37.3686
ω_7	38.6329	38.6544	38.6339
ω_8	39.3779	39.4619	39.4755
ω_9	44.9766	45.0620	45.0627
ω_{10}	52.0542	52.1692	52.1414
ω_{11}	64.4975	64.9022	65.1329
ω_{12}	65.8761	66.2917	66.5366

Table (4.2) - Comparison of Natural Frequencies for Sample Problem Three

Chapter (5)

Forced Vibration Response by The Boundary Flexibility Vector Method of CMS

5.1) Introduction:

This chapter presents the formulation of the boundary flexibility method of CMS, for the forced vibration problem. It also presents a derivation indicating that the load-dependent Ritz vectors, used by Wilson in his formulation for fixed-interface CMS, yield a flexibility representation higher than the corresponding finite-element model, for the internal D.O.F., where external loads are applied to generate the vectors. It is further shown in this chapter, how the load-dependent Ritz vectors can be generated in a way such that the flexibility representation of fixed-interface components is equal to the corresponding finite-element model. Finally, a numerical example is solved by several methods and for different loading conditions. The results of the boundary flexibility method is compared to that obtained by the other methods.

5.2) The Application of The Boundary Flexibility Method of CMS in Forced Vibration Problems:

The boundary flexibility method of CMS, presented in chapter three, can be applied to forced vibration problems. For components having no external loads applied, the formulation will be similar to that presented in chapter three, for fixed and free-interface components. For components where external loads

are applied, the following two cases are considered:

1) If the external load is applied at an interface D.O.F., the formulation will be exactly the same as to what was presented in chapter three, for fixed and free-interface components.

2) If the external load is applied at an internal D.O.F., then the external load vector is transformed to the mixed physical and generalized coordinate system as follows:

Consider the equation of motion for a fixed or free-interface component:

$$[m]\{\ddot{u}\} + [c]\{\dot{u}\} + [k]\{u\} = \begin{Bmatrix} P_c \\ f_1 \end{Bmatrix} \tag{5.1}$$

where $\{P_c\}$ = subvector of interface internal forces between components.

$\{f_1\}$ = subvector of external loads applied at one or more internal D.O.F.

$[m]$, $[k]$, $[c]$, $\{\ddot{u}\}$, $\{\dot{u}\}$ and $\{u\}$ are as before.

The equation of motion is transformed to the mixed physical and generalized coordinate system as follows:

$$[T]^T [m][T]\{\ddot{u}\} + [T]^T [c][T]\{\dot{u}\} + [T]^T [k][T]\{u\} = [T]^T \begin{Bmatrix} P_c \\ f_1 \end{Bmatrix} \tag{5.2}$$

where $[T]$ = transformation matrix given by either of equations (3.11), (3.14) or (3.36).

It should be noted that the internal force subvector, $\{P_c\}$, will be unaffected by any of the transformations given by equations

(3.11), (3.14) or (3.36). While the external load subvector is transformed from the physical coordinate system to a mixed physical and generalized coordinate system.

For free-interface components, the method will be similar to that outlined in chapter three, with every $\{u_c\}$ vector replaced by $\{u_a\}$ and every $\{u_i\}$ vector replaced by $\{u_w\}$, in all the equations , where $\{u_a\}$ is the vector of D.O.F. where external or internal loads are applied and $\{u_w\}$ is the complement of $\{u_a\}$ in $\{u\}$. Thus the load-dependent vectors will be contained in the residual attachment modes and the boundary flexibility vectors.

For fixed-interface components, the formulation has to be modified in order to generate residual attachment modes from the external applied loads. Assume that the external loads are applied at a subvector $\{u_a\}$ of internal D.O.F. Then the total displacement vector is partitioned into

$$\{u\} = \begin{Bmatrix} u_c \\ u_i \end{Bmatrix} = \begin{Bmatrix} u_c \\ u_a \\ u_w \end{Bmatrix} \tag{5.3}$$

where $\{u_a\}$ = subvector of internal D.O.F. where external loads are applied.

$\{u_w\}$ = complement of $\{u_a\}$ in $\{u_i\}$.

In order to obtain good results for the set $\{u_a\}$ where the external loads are applied, the columns of the flexibility matrix corresponding to $\{u_a\}$, in the CMS model, must be complete. In

other words, the flexibility of the D.O.F. in $\{u_a\}$ must be the same in the CMS and FEM models. Similar to free-interface components, the set of constraint modes $[\Phi_c]$ and fixed-interface boundary flexibility vectors $[Q_\ell]$ do not supply the full flexibility for the D.O.F. in $\{u_a\}$. Thus residual attachment modes have to be added to obtain the desired full flexibility. Hence, the contribution of the constraint modes $[\Phi_c]$ and the fixed-interface boundary flexibility vectors $[Q_\ell]$ to the flexibility matrix, must be obtained.

Let [T] be the matrix containing $[\Phi_c]$ and $[Q_\ell]$ in its columns

$$[T] = \begin{bmatrix} [\Phi_c] & [Q_\ell] \end{bmatrix} = \begin{bmatrix} [I_{cc}] & [0]_{c\ell} \\ [\Phi_{ic}] & [Q]_\ell \end{bmatrix} \qquad (5.4)$$

The size of [T] is (n) x (c + ℓ). Where n is the total number of physical D.O.F. in component, c is the number of interface D.O.F. and ℓ is equal to the number of the fixed-interface boundary flexibility vectors. Similar to equation (3.27), the contribution of [T] to the flexibility matrix is given by:

$$[g_k] = [T] \left[[T]^T [k] [T] \right]^{-1} [T]^T \qquad (5.5)$$

where $[g_k]$ = contribution of [T] to the flexibility matrix. Substitute equation (5.4) in equation (5.5) and note that

$[\Phi_{ic}] = - [k_{ii}]^{-1} [k_{ic}]$ and $[k_{ic}]^T = [k_{ci}]$

$$[g_k] = \begin{bmatrix} [I_{cc}] & [0]_{c\ell} \\ -[k_{ii}]^{-1}[k_{ic}] & [Q_\ell] \end{bmatrix} *$$

$$\left[\begin{bmatrix} [I_{cc}] & -[k_{ci}][k_{ii}]^{-1} \\ [0_{\ell c}] & [Q_\ell]^T \end{bmatrix}\begin{bmatrix} [k_{cc}] & [k_{ci}] \\ [k_{ic}] & [k_{ii}] \end{bmatrix}\begin{bmatrix} [I_{cc}] & [0]_{c\ell} \\ -[k_{ii}]^{-1}[k_{ic}] & [Q_\ell] \end{bmatrix}\right]^{-1} *$$

$$\begin{bmatrix} [I_{cc}] & -[k_{ci}][k_{ii}]^{-1} \\ [0_{\ell c}] & [Q_\ell]^T \end{bmatrix} \qquad (5.6)$$

From equation (3.13)

$$[g_k] = \begin{bmatrix} [I_{cc}] & [0]_{c\ell} \\ -[k_{ii}]^{-1}[k_{ic}] & [Q_\ell] \end{bmatrix}\begin{bmatrix} [k_{cc}]-[k_{ci}][k_{ii}]^{-1}[k_{ic}] & [0]_{c\ell} \\ [0_{\ell c}] & [Q_\ell]^T[k_{ii}][Q_\ell] \end{bmatrix}^{-1} *$$

$$\begin{bmatrix} [I_{cc}] & -[k_{ci}][k_{ii}]^{-1} \\ [0_{\ell c}] & [Q_\ell]^T \end{bmatrix} \qquad (5.7)$$

If the fixed-interface boundary flexibility vectors were orthogonalized and normalized with respect to the stiffness matrix, instead of the mass matrix, then, $[Q_\ell]^T[k_{ii}][Q_\ell]=[I]_{\ell\ell}$ and hence,

$$[g_k] = \begin{bmatrix} [I_{cc}] & [0]_{c\ell} \\ -[k_{ii}]^{-1}[k_{ic}] & [Q_\ell] \end{bmatrix} \begin{bmatrix} \left[[k_{cc}]-[k_{ci}][k_{ii}]^{-1}[k_{ic}]\right]^{-1} & [0]_{c\ell} \\ [0_{\ell c}] & [I]_{\ell\ell} \end{bmatrix} *$$

$$\begin{bmatrix} [I_{cc}] & -[k_{ci}][k_{ii}]^{-1} \\ [0_{\ell c}] & [Q_\ell]^T \end{bmatrix}$$

$$= \begin{bmatrix} \left[[k_{cc}]-[k_{ci}][k_{ii}]^{-1}[k_{ic}]\right]^{-1} & [0]_{c\ell} \\ -[k_{ii}]^{-1}[k_{ic}]\left[[k_{cc}]-[k_{ci}][k_{ii}]^{-1}[k_{ic}]\right]^{-1} & [Q_\ell] \end{bmatrix} *$$

$$\begin{bmatrix} [I_{cc}] & -[k_{ci}][k_{ii}]^{-1} \\ [0_{\ell c}] & [Q_\ell]^T \end{bmatrix}$$

$$= \begin{bmatrix} [g_k]_{cc} & [g_k]_{ci} \\ [g_k]_{ic} & [g_k]_{ii} \end{bmatrix} \qquad (5.8)$$

where $[g_k]_{cc} = \left[[k_{cc}]-[k_{ci}][k_{ii}]^{-1}[k_{ic}]\right]^{-1}$

$$[g_k]_{ci} = [g_k]_{ic}^T = -\left[[k_{cc}]-[k_{ci}][k_{ii}]^{-1}[k_{ic}]\right]^{-1}[k_{ci}][k_{ii}]^{-1}$$

$$[g_k]_{ii} = [k_{ii}]^{-1}[k_{ic}]\left[[k_{cc}]-[k_{ci}][k_{ii}]^{-1}[k_{ic}]\right]^{-1}[k_{ci}][k_{ii}]^{-1} + [Q_\ell][Q_\ell]^T$$

Let [g] be the total flexibility matrix of the component. In Appendix B, [g] is derived in terms of the stiffness submatrices of the fixed-interface component. Thus,

$$[g] = \begin{bmatrix} [g_{cc}] & [g_{ci}] \\ [g_{ic}] & [g_{ii}] \end{bmatrix} \tag{5.9}$$

where $[g_{cc}] = \left[[k_{cc}]-[k_{ci}][k_{ii}]^{-1}[k_{ic}]\right]^{-1}$

$$[g_{ci}] = [g_{ic}]^T = -\left[[k_{cc}]-[k_{ci}][k_{ii}]^{-1}[k_{ic}]\right]^{-1}[k_{ci}][k_{ii}]^{-1}$$

$$[g_{ii}] = [k_{ii}]^{-1} + [k_{ii}]^{-1}[k_{ic}]\left[[k_{cc}]-[k_{ci}][k_{ii}]^{-1}[k_{ic}]\right]^{-1}[k_{ci}][k_{ii}]^{-1}$$

Hence the unrepresented flexibility matrix, or the residual flexibility matrix $[g_d]$ is given by:

$$[g_d] = [g] - [g_k] = \begin{bmatrix} [0_{cc}] & [0_{ci}] \\ [0_{ic}] & [k_{ii}]^{-1} - [Q_\ell][Q_\ell]^T \end{bmatrix} \tag{5.10}$$

For fixed-interface components, the set $\{u_c\}$ is considered fixed. Hence the residual flexibility matrix for the fixed-interface component will be $[g_d]_{ii}$

$$[g_d]_{ii} = [k_{ii}]^{-1} - [Q_\ell][Q_\ell]^T \tag{5.11}$$

which is the full flexibility matrix for the fixed component, minus the contribution of the fixed-interface boundary flexibility vectors to the flexibility matrix. The residual flexibility

matrix given by equation (5.11), is used to obtain the residual attachment modes $[\Psi_d]$ for the fixed component. This is achieved by applying unit forces at the subvector $\{u_a\}$ of the physical D.O.F. Hence the residual attachment modes $[\Psi_d]$ are the columns of the residual flexibility matrix $[g_d]_{11}$, corresponding to the D.O.F. in $\{u_a\}$. It should be noted that the final displaced shapes formed by $[\Phi_c]$, $[Q_\ell]$ and $[\Psi_d]$, will provide the full flexibility representation for the D.O.F. in $\{u_a\}$ where the external loads are applied. Thus the prediction of responses of D.O.F. in $\{u_a\}$ by using this formulation, is expected to be better than a formulation which does not use the residual attachment modes. If $\{p_\ell\}$ and $\{p_d\}$ are the generalized coordinates associated with $[Q_\ell]$ and $[\Psi_d]$, the sizes of $\{p_\ell\}$ and $\{p_d\}$ are (ℓ) and (a) respectively, where (ℓ) is the number of boundary flexibility vectors in $[Q_\ell]$, and (a) is the number of D.O.F. in $\{u_a\}$ where the external loads are applied. The transformation of coordinates for the fixed-interface component, for case of external applied loads is given by:

$$\begin{Bmatrix} u_c \\ u_i \end{Bmatrix} = [T] \begin{Bmatrix} u_c \\ p_\ell \\ p_d \end{Bmatrix} = \begin{bmatrix} [I_{cc}] & [0]_{c\ell} & [0_{ca}] \\ [\Phi_{ic}] & [Q_\ell] & [\Psi_d] \end{bmatrix} \begin{Bmatrix} u_c \\ p_\ell \\ p_d \end{Bmatrix} \qquad (5.12)$$

where [T] = transformation matrix of order (n) x ($c + \ell + a$). The transformation matrix [T], is used to transform the property matrices and the load subvector $\{f_i\}$, as in equation (5.2), where

$$\{f_i\} = \begin{Bmatrix} f_a \\ 0_w \end{Bmatrix} \tag{5.13}$$

where $\{f_a\}$ = subvector of external loads applied at internal degrees of freedom $\{u_a\}$.

5.3) A Comment on Wilson's Load-Dependent Ritz Vectors:

As mentioned in chapter two, Wilson et al. [15] applied the load-dependent Ritz vectors to fixed-interface CMS. Only constraint modes were used for components having no external loads applied at internal D.O.F. For components having external loads $\{f_a\}$ applied at a subset $\{u_a\}$ of the internal D.O.F. $\{u_i\}$, load-dependent Ritz vectors were generated from those applied loads. This section proves that the load-dependent Ritz vectors, the subsequent vectors generated from them and the constraint modes, will supply flexibility properties for the D.O.F. in $\{u_a\}$, greater than that of the corresponding finite-element model. Thus higher displacement responses are obtained for the D.O.F. in $\{u_a\}$, by using this model.

Consider the same notation used in the previous section. Starting only with constraint modes $[\Phi_c]$, as displacement shapes for a component, we can proceed in a similar manner to that of the previous section, to obtain the contribution of the constraint modes to the total flexibility matrix. It was found to be equal to (compare to equation (5.8)):

$$[g_k] = \begin{bmatrix} [g_k]_{cc} & [g_k]_{ci} \\ [g_k]_{ic} & [g_k]_{ii} \end{bmatrix} \tag{5.14}$$

where $[g_k]_{cc} = \left[[k_{cc}]-[k_{ci}][k_{ii}]^{-1}[k_{ic}]\right]^{-1}$

$$[g_k]_{ci} = [g_k]_{ic}^T = -\left[[k_{cc}]-[k_{ci}][k_{ii}]^{-1}[k_{ic}]\right]^{-1}[k_{ci}][k_{ii}]^{-1}$$

$$[g_k]_{ii} = [k_{ii}]^{-1}[k_{ic}]\left[[k_{cc}]-[k_{ci}][k_{ii}]^{-1}[k_{ic}]\right]^{-1}[k_{ci}][k_{ii}]^{-1}$$

If [g] is the total flexibility given by equation (5.9), then the residual flexibility matrix $[g_d]$ (or the unrepresented flexibility), is given by

$$[g_d] = [g] - [g_k] = \begin{bmatrix} [0_{cc}] & [0_{ci}] \\ [0_{ic}] & [k_{ii}]^{-1} \end{bmatrix} \tag{5.15}$$

Which indicates that the constraint modes $[\Phi_k]$ supply full flexibility to the submatrices $[g_{cc}]$, $[g_{ic}]$ and $[g_{ci}]$, and an incomplete flexibility to the submatrix $[g_{ii}]$. Wilson et al. [15] used this $[k_{ii}]^{-1}$ matrix to generate the starting load-dependent vector from the external applied load. Thus the starting vectors will be the columns of the residual flexibility matrix, $[g_d] = [k_{ii}]^{-1}$, corresponding to the D.O.F. in $\{u_a\}$, where the external applied loads act. Any subsequent vectors are generated according to Wilson's algorithm given in chapter two. It should be noted

that the starting vectors, by themselves, will supply the full flexibility representation for the D.O.F. in $\{u_a\}$ and any subsequent generated vectors will add to the flexibility of the D.O.F. in $\{u_a\}$. Thus the resulting component will have an "excess flexibility" for the D.O.F. in $\{u_a\}$ and hence a higher response prediction for the D.O.F. where the external loads are applied.

In order to obtain a component having a more realistic representation of the flexibility, another approach is suggested. This approach is similar to that taken in chapter three for free-interface components, in the boundary flexibility method of CMS. Assume, for simplicity, that the external load is applied only at D.O.F. number j from the set of degrees of freedom in $\{u_i\}$. The static deflected shape obtained by this load is by definition equal to $\{\phi_a^j\}$, which is the attachment mode shape for D.O.F. number j. The $\{\phi_a^j\}$ vector is equal to the column of the flexibility matrix, $[g_d] = [k_{ii}]^{-1}$, corresponding to D.O.F. number j. $\{\phi_a^j\}$ is the starting load-dependent Ritz vector used by Wilson et al. [15]. In this formulation the starting vector will instead be

$$\{x_1^*\} = [k_{ii}]^{-1} [m_{ii}] \{\phi_a^j\} \qquad (5.16)$$

where $\{x_1^*\}$ = starting load-dependent Ritz vector.

It can be seen that $\{x_1^*\}$ is the inertial loading of the static response to the applied external load, and not the response itself, as in Wilson's method. This is exactly similar to that

presented for free-interface components, in the boundary flexibility method of CMS (see section 3.3.1.1.). The subsequent vectors are generated from the $\{x_1^*\}$ vector by using Wilson's algorithm. It should be noted that orthogonalizations and normalizations are carried out with respect to the stiffness matrix $[k_{ii}]$, instead of the mass matrix $[m_{ii}]$. Thus a new simple residual flexibility matrix is obtained (as in the previous section):

$$[g_{d'}] = [g_d] - [X][X]^T$$

$$= [k_{ii}]^{-1} - [X][X]^T \qquad (5.17)$$

where $[g_{d'}]$ = new residual flexibility matrix.

$[X]$ = matrix containing the generated load-dependent Ritz vectors in its columns.

This new residual flexibility matrix is used to generate residual attachment modes $[\Psi_d]$, which will be, in this case, the columns of the $[g_{d'}]$ matrix corresponding to D.O.F. number j. The final transformation of coordinates, in its general form, is given by:

$$\begin{Bmatrix} u_c \\ u_i \end{Bmatrix} = [T] \begin{Bmatrix} u_c \\ p_\ell \\ p_d \end{Bmatrix} = \begin{bmatrix} [I_{cc}] & [0]_{c\ell} & [0_{ca}] \\ [\Phi_{ic}] & [X] & [\Psi_d] \end{bmatrix} \begin{Bmatrix} u_c \\ p_\ell \\ p_d \end{Bmatrix} \qquad (5.18)$$

where $\{p_\ell\}$ = generalized coordinates associated with (ℓ) generated load-dependent vectors in [X].

$\{p_d\}$ = generalized coordinates associated with (a) residual attachment modes $[\Psi_d]$, corresponding to

D.O.F. in $\{u_a\}$ where external loads are applied.

[T] = transformation matrix of order (n) x (c + ℓ + a).

It should be noted that by this approach, we can ensure a full flexibility representation for the D.O.F. in $\{u_a\}$. The reason for this is that the contribution of the generated load-dependent Ritz vectors to the flexibility matrix, was taken into consideration in equation (5.17). The property matrices and the external load vector are transformed, using the [T] matrix as before.

5.4) Sample Problem:

The structure used to illustrate the forced vibration response by the boundary flexibility vector method of CMS, is the one shown in Figure (4.11), in the previous section. The external loading function applied is a step function, shown in Figure (5.1). The method used for integrating the equations of motions, is the Newmark step by step integration scheme (see Bathe [18]), with constants α and δ equal to 0.25 and 0.5 respectively. This method is chosen because it is unconditionally stable for the given values of α and δ. The time step chosen for integration is 0.02 seconds, which is less than the first period of the structure divided by ten. Proportional damping is used (see [16] and [18]) according to the following equation:

$$[C] = \alpha [M] + \beta [K] \qquad (5.19)$$

Where [C], [M] and [K] are damping, mass and stiffness matrices respectively, for the whole system. α and β are constants to be determined from two given damping ratios corresponding to two

unequal natural frequencies (see Appendix C). Two levels of damping are used; a low level corresponding to $\xi_1 = 0.002$ and $\xi_8 = 0.15$ and a higher level corresponding to $\xi_1 = 0.05$ and $\xi_8 = 0.15$. ξ_1 and ξ_8 are the damping ratios for natural frequencies one and eight respectively. For the above two levels of damping, the corresponding proportional constants are ($\alpha_1 = -7.489$, $\beta_1 = 0.0124$) and ($\alpha_2 = -3.584$, $\beta_2 = 0.0099$) respectively (see Appendix C).

Three loading cases are considered, for each damping level. They are as follows:

1) Load acting at node 31 in the global Z direction. Node 31 is an interface D.O.F. in components one and two. This case is solved by three methods; method one is by direct integration of the equations of motion, method two is by conventional CMS by using constraint modes and three fixed-interface normal modes per component and method three is by the boundary flexibility method of CMS by using constraint modes and three fixed-interface boundary flexibility Ritz vectors. For all methods, Z-direction displacement responses are obtained at nodes 13, 31 and 112 (see Figures (5.2), (5.3) and (5.4) for case of low damping, and Figures (5.5), (5.6) and (5.7) for case of high damping). The results obtained are almost the same for the three methods. For all responses, the case of high damping converges to the static solution after few cycles, while the case of low damping keeps oscillating about the static response, and requires a larger

number of response cycles in order to converge to it. For the two damping levels, the maximum and static responses, for each given node, obtained by the three different methods are shown in Tables (5.1) and (5.2).

2) Load acting at node 10 in the global Z-direction. This case is also solved by the above three methods. The Z-direction displacement responses are calculated at nodes 10, 84 and 112 (see Figures (5.8), (5.9) and (5.10) for low damping case and Figures (5.11), (5.12) and (5.13) for high damping case). Also the results obtained are almost the same for all three methods. The maximum and static responses obtained by each method, for both the two damping levels, are presented in Tables (5.3) and (5.4).

3) Load acting at node 16 in the global Y-direction. This case is solved by the above three methods and by another fourth method. The fourth method is the boundary flexibility method of CMS, with two boundary flexibility vectors, one residual attachment mode and constraint modes for component one where the load is applied, and three boundary flexibility vectors and constraint modes for each of the other three components. The Y-direction displacement responses are obtained at nodes 16, 87 and 99 (see Figures (5.14), (5.15) and (5.16) for low damping case and Figures (5.17), (5.18) and (5.19) for high damping case). Tables (5.5) and (5.6) show the maximum and static responses obtained by all methods, for the two damping levels. For nodes 87 and 99, the results obtained by the four methods are almost the same, for the two levels of

damping. For the response at node 16, the results are not the same, as can be seen from Figure (5.14) and Table (5.5) for case of low damping, and Figure (5.17) and Table (5.6) for case of high damping. The results obtained, for the two levels of damping, by the conventional CMS method, indicates the maximum response is about 11% lower and the static response is about 14.5% lower than the direct integration results. Better results are obtained by the boundary flexibility method of CMS, in which three boundary flexibility vectors are used for each component. The results obtained by the boundary flexibility method of CMS, by using one residual attachment mode for component number one, are in the best agreement with the results obtained by the direct integration method.

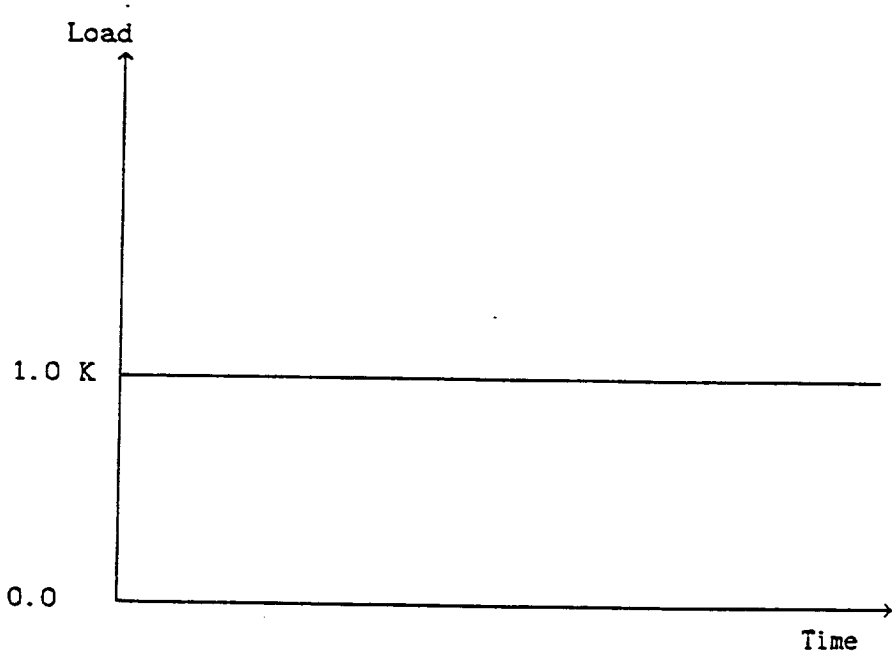


Figure (5.1) - Load for Sample Problem.

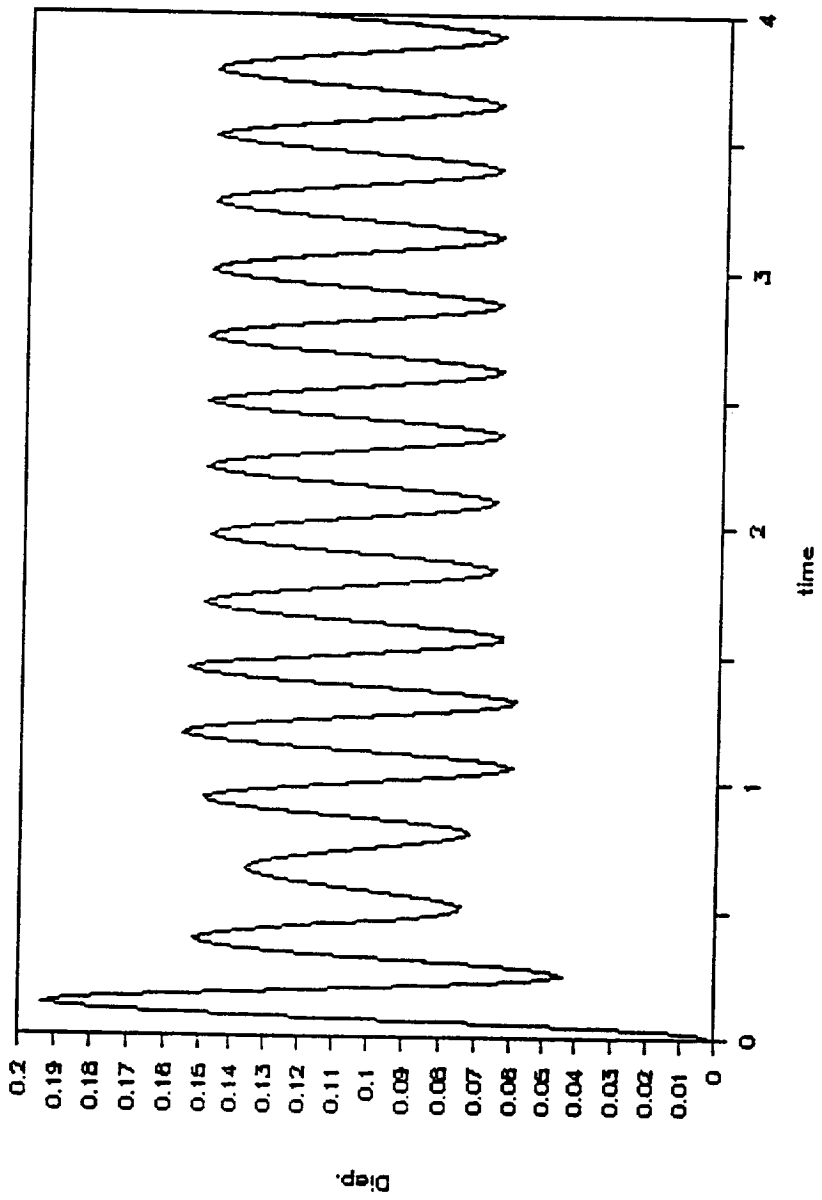


Figure (5.2) - Response at Node 13 in The Z-Direction
for The Case of Load Acting at Node 31
in The Z-Direction (Low Damping)

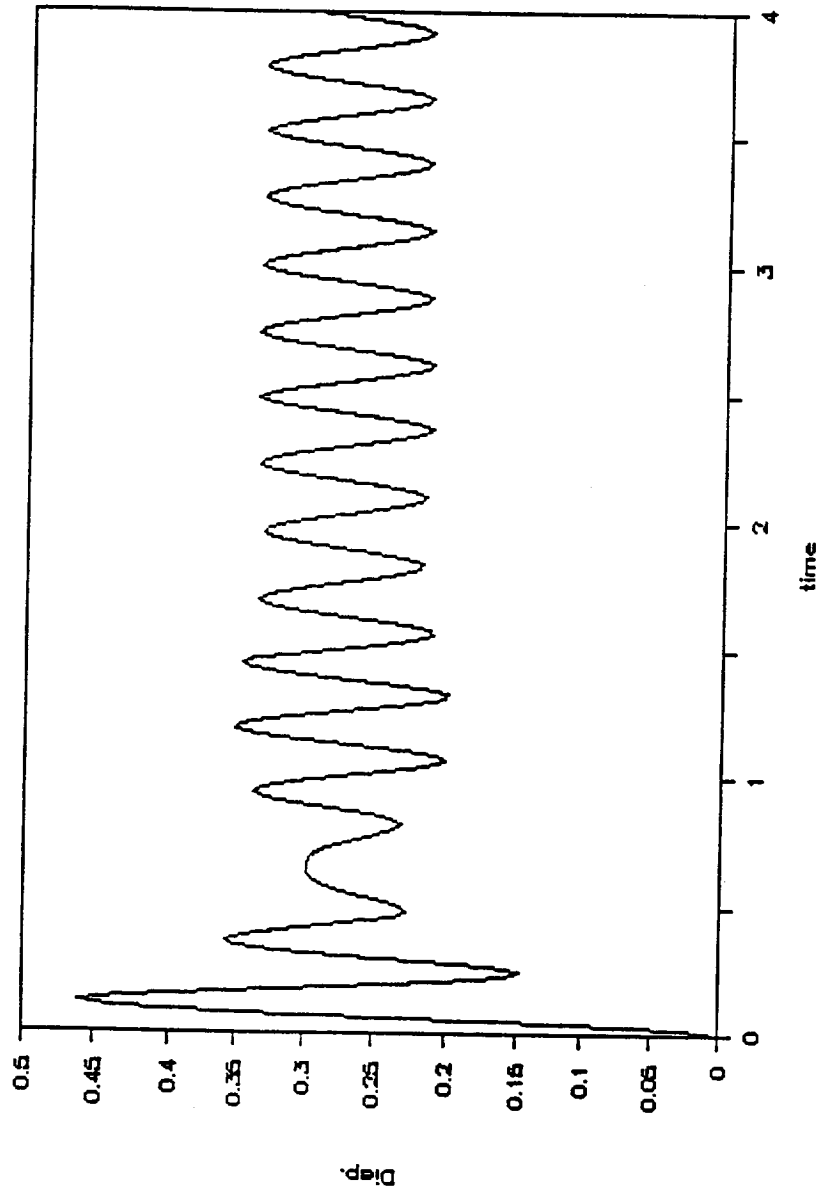


Figure (5.3) - Response at Node 31 In The Z-Direction
for The Case of Load Acting at Node 31
In The Z-Direction (Low Damping)

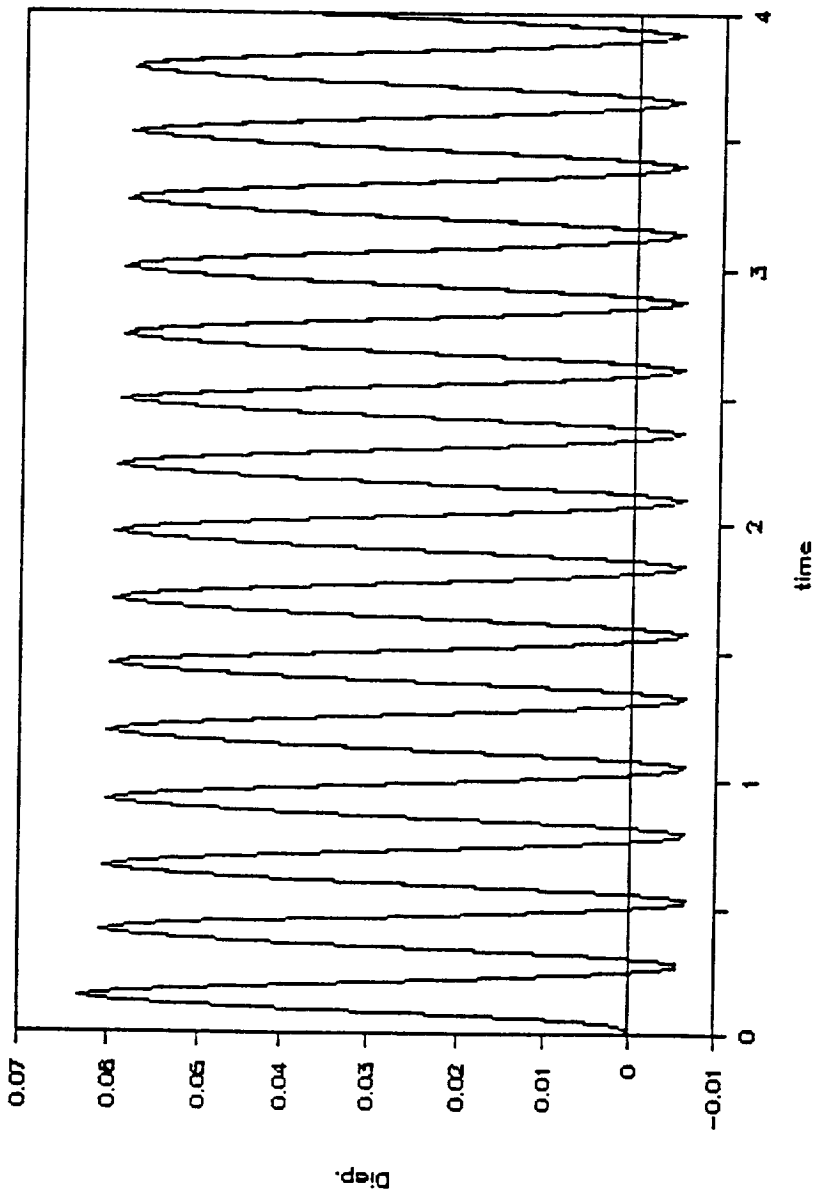


Figure (5.4) - Response at Node 112 in The Z-Direction
for The Case of Load Acting at Node 31
in The Z-Direction (Low Damping)

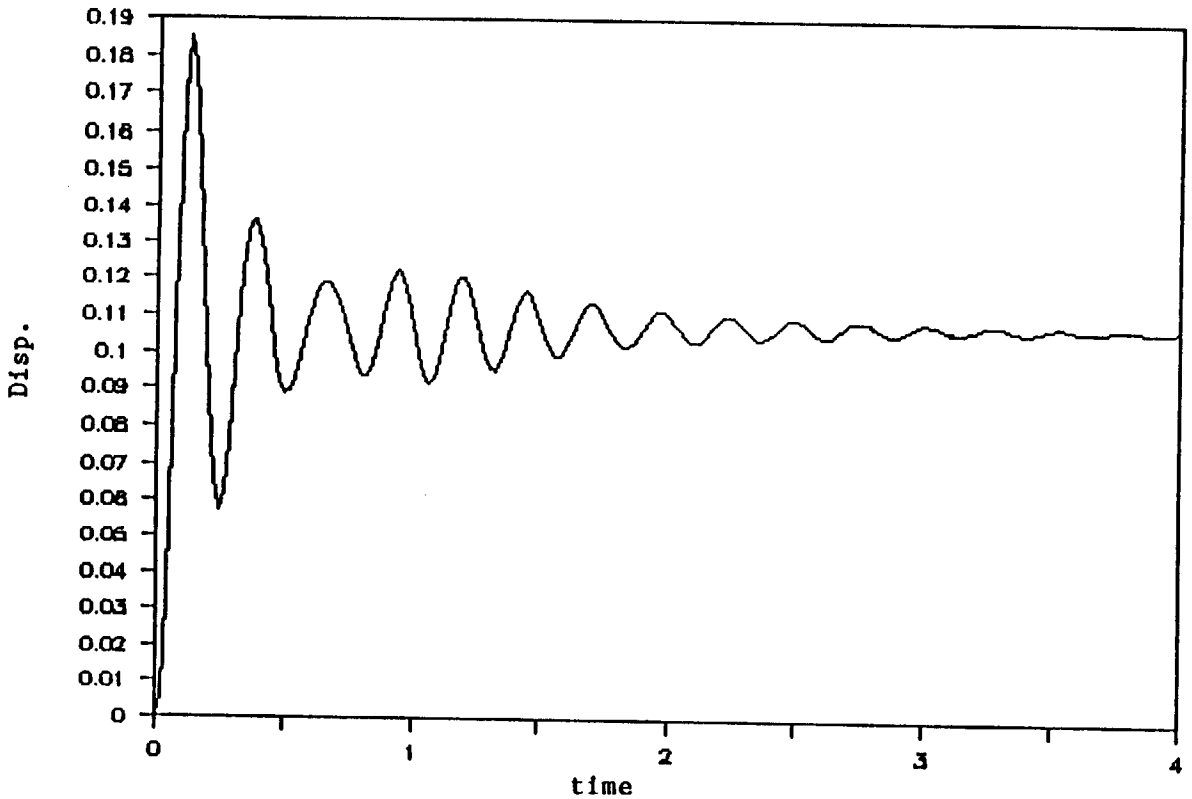


Figure (5.5) - Response at Node 13 in The Z-Direction for The Case of Load Acting at Node 31 in The Z-Direction (High Damping)

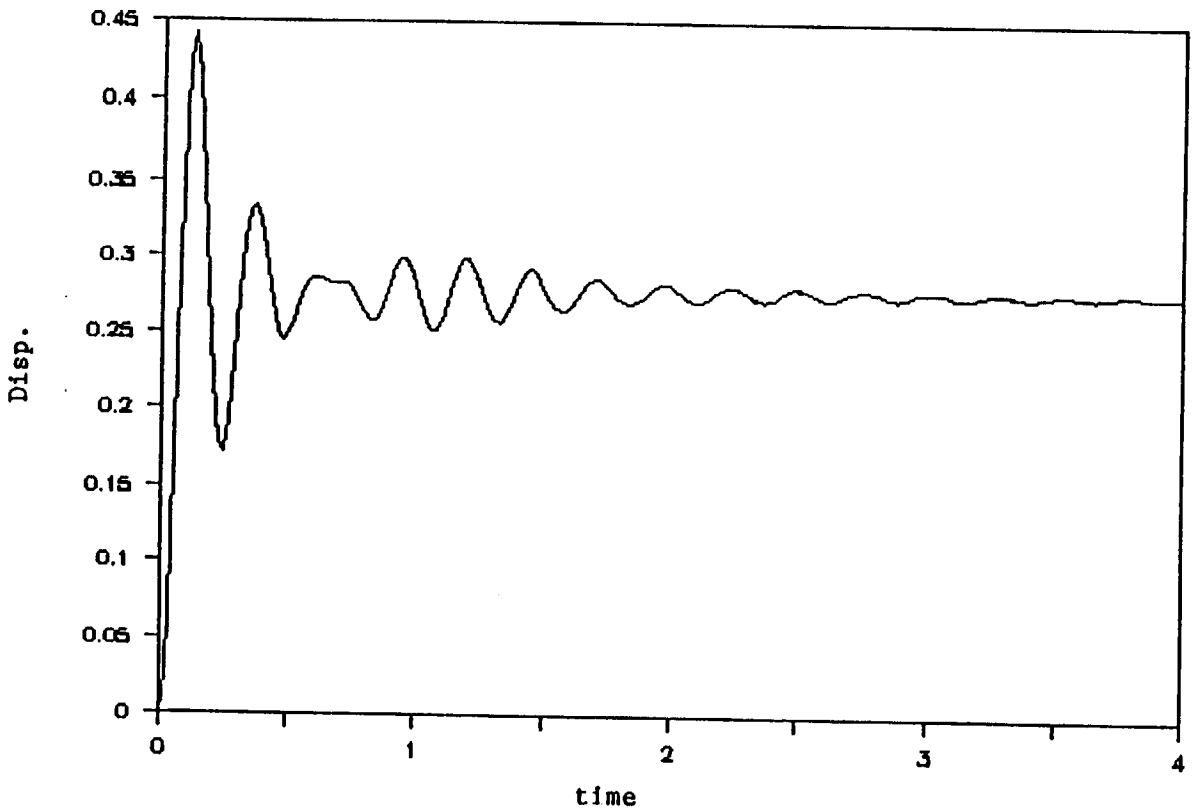


Figure (5.6) - Response at Node 31 in The Z-Direction for The Case of Load Acting at Node 31 in The Z-Direction (High Damping)

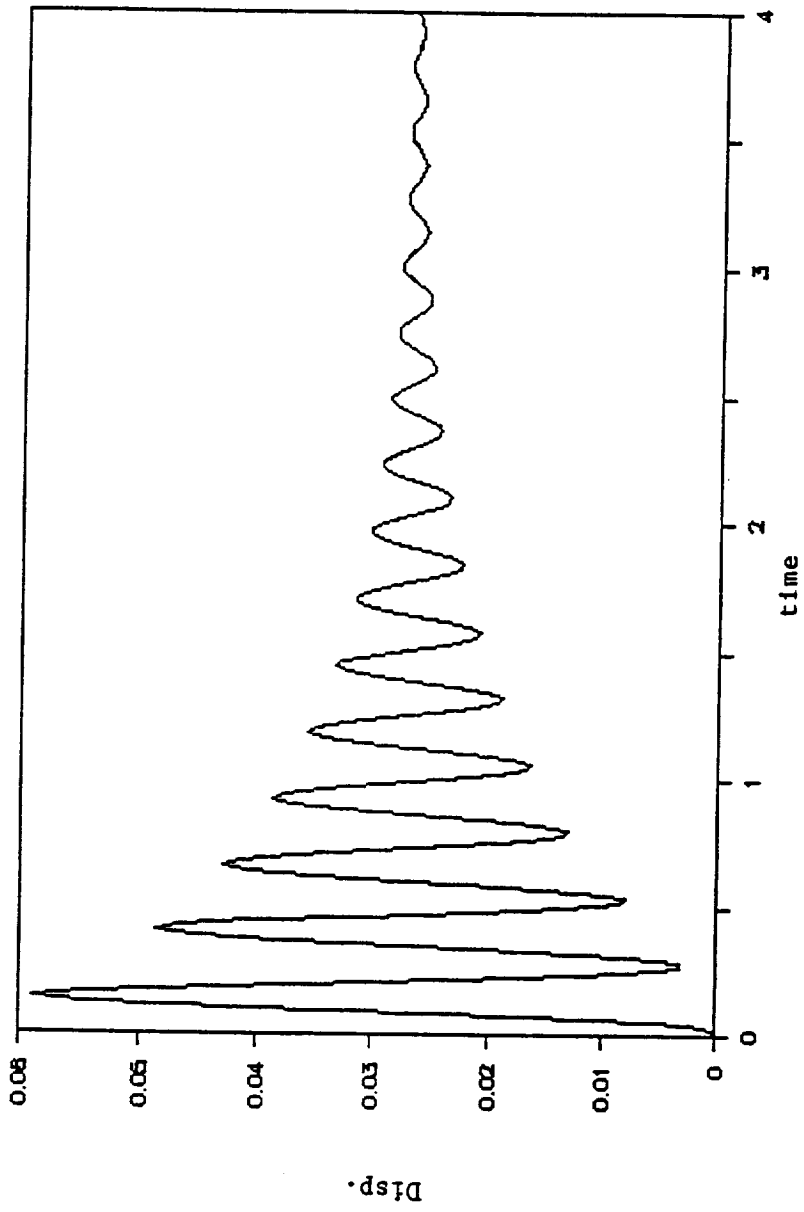


Figure (5.7) - Response at Node 112 in The Z-Direction
for The Case of Load Acting at Node 31
in The Z-Direction (High Damping)

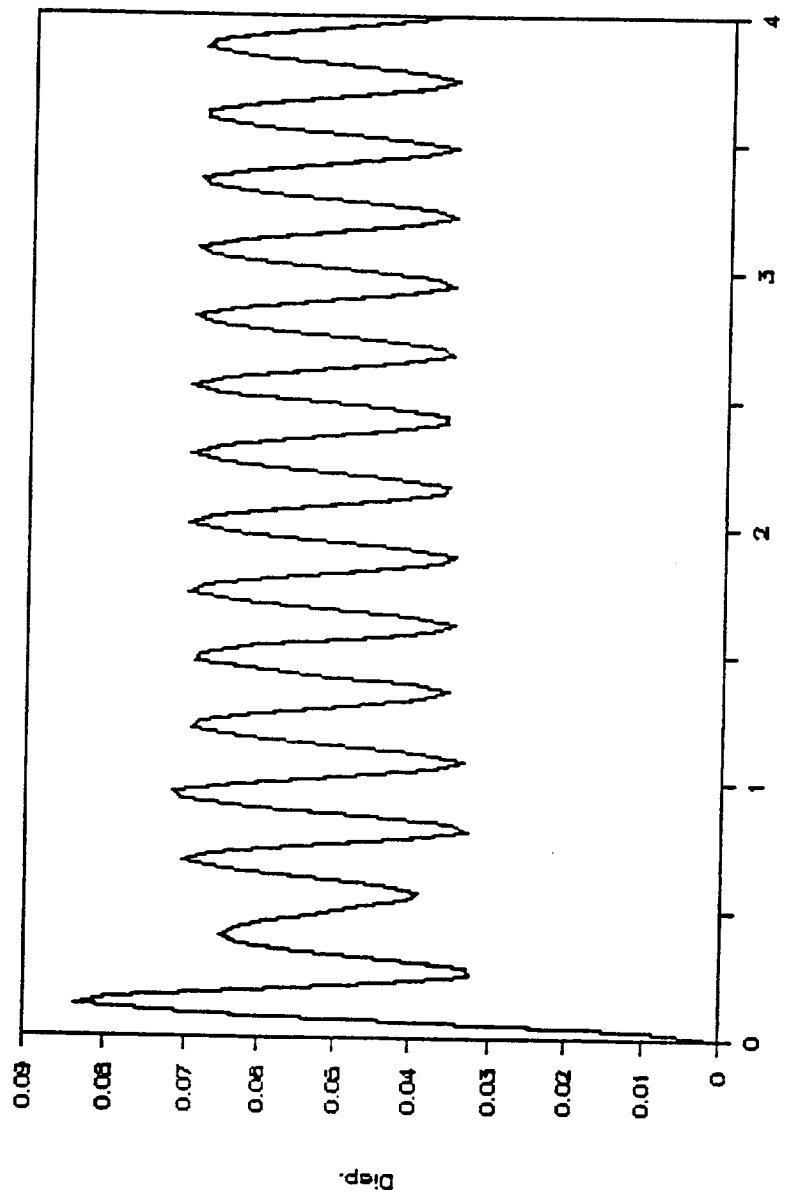


Figure (5.8) - Response at Node 10 in The Z-Direction for The Case of Load Acting at Node 10 in The Z-Direction (Low Damping)

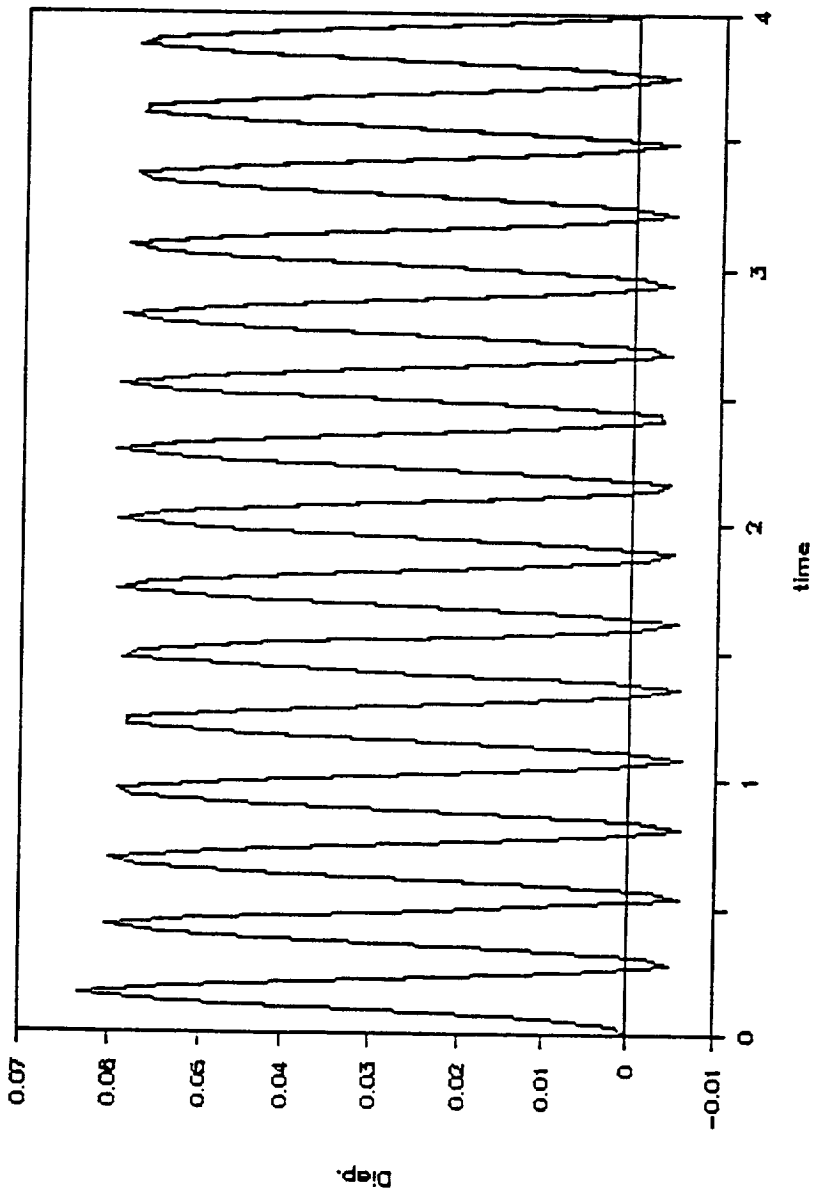


Figure (5.9) - Response at Node 84 In The Z-Direction
for The Case of Load Acting at Node 10
In The Z-Direction (Low Damping)

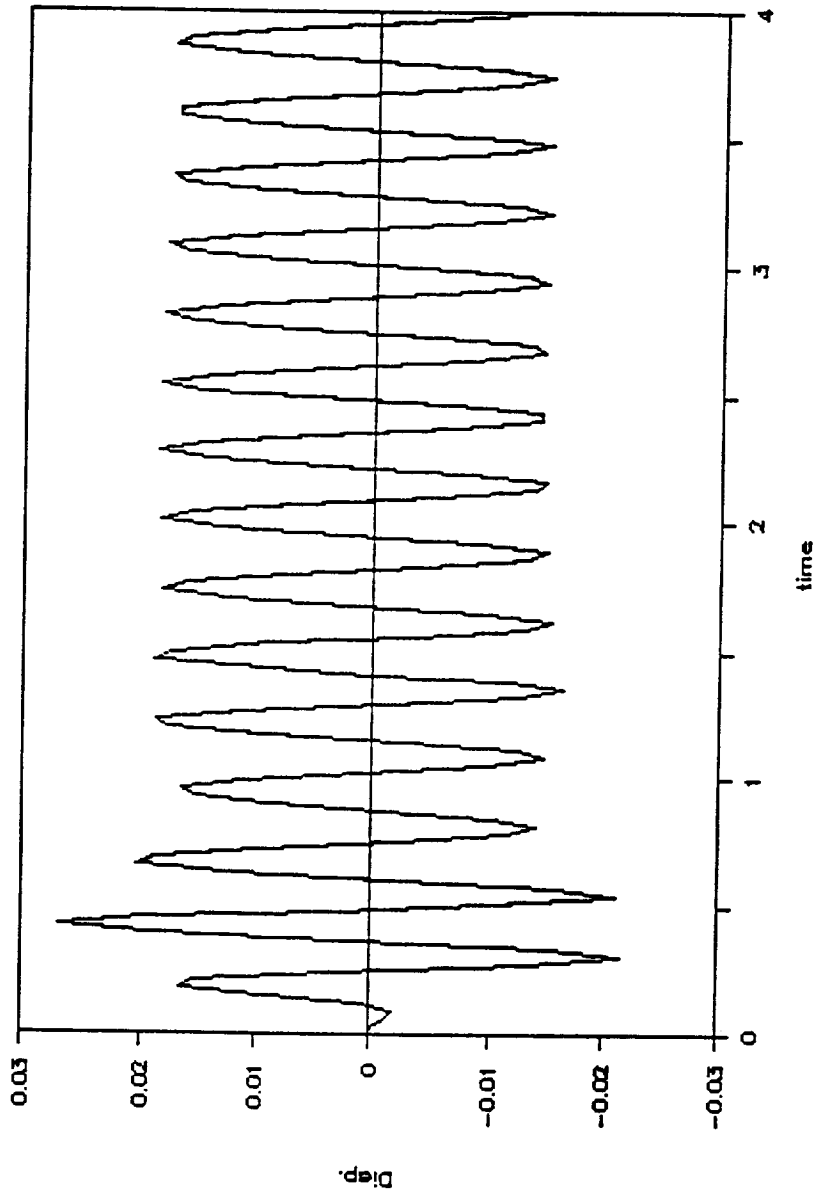


Figure (5.10) - Response at Node 112 in The Z-Direction for The Case of Load Acting at Node 10 in The Z-Direction (Low Damping)

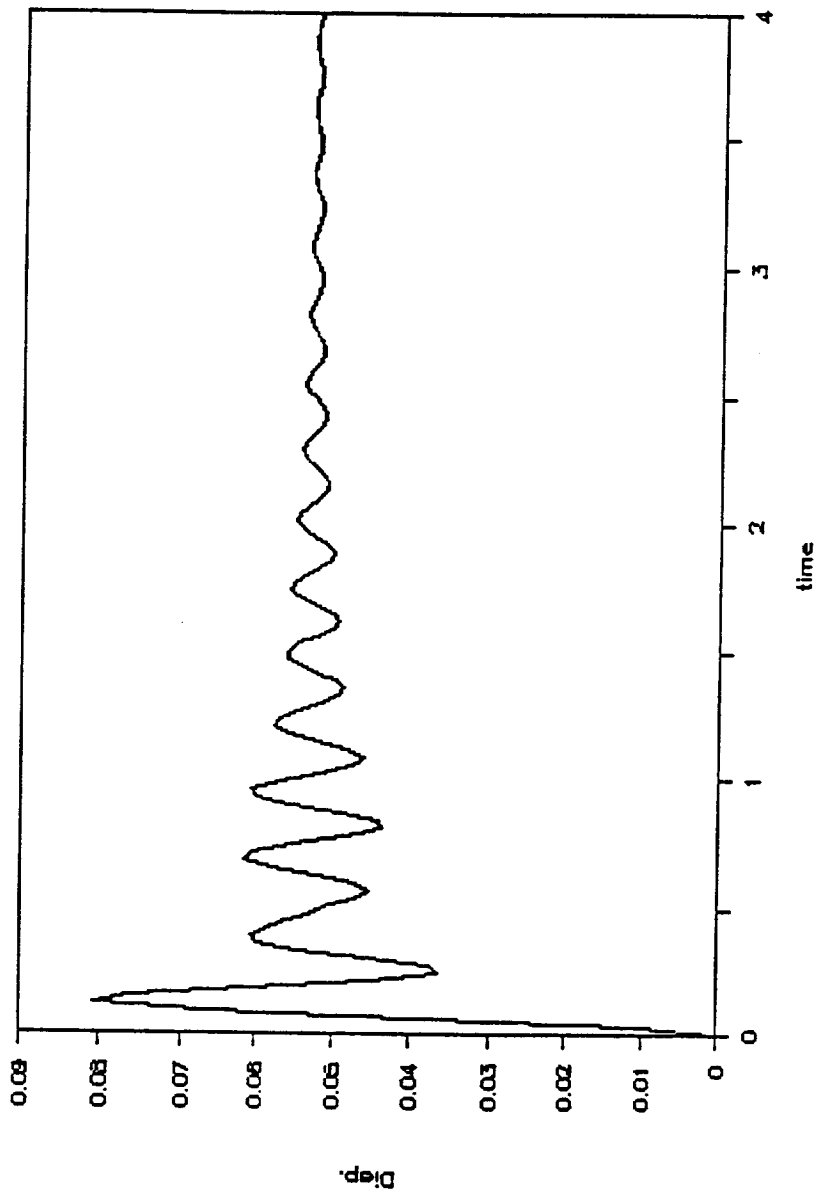


Figure (5.11) - Response at Node 10 In The Z-Direction for The Case of Load Acting at Node 10 In The Z-Direction (High Damping)

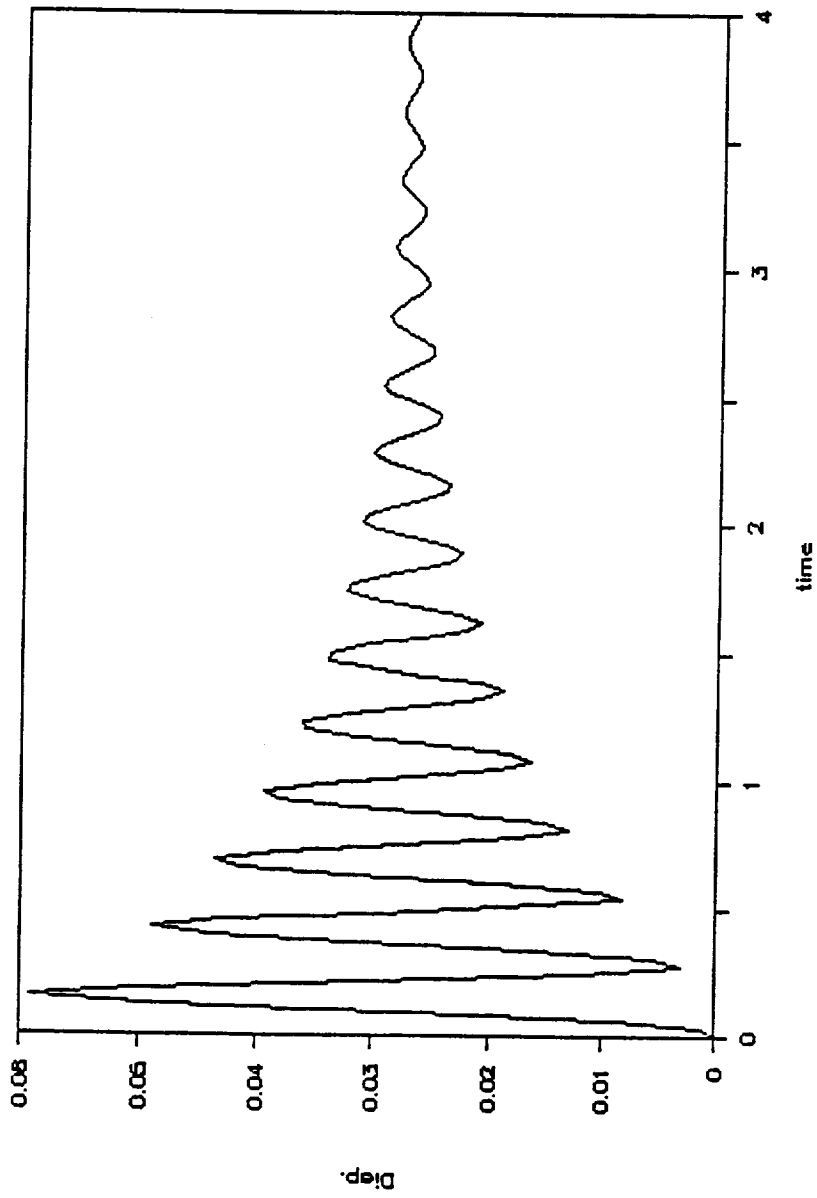


Figure (5.12) - Response at Node 84 In The Z-Direction for The Case of Load Acting at Node 10 In The Z-Direction (High Damping)

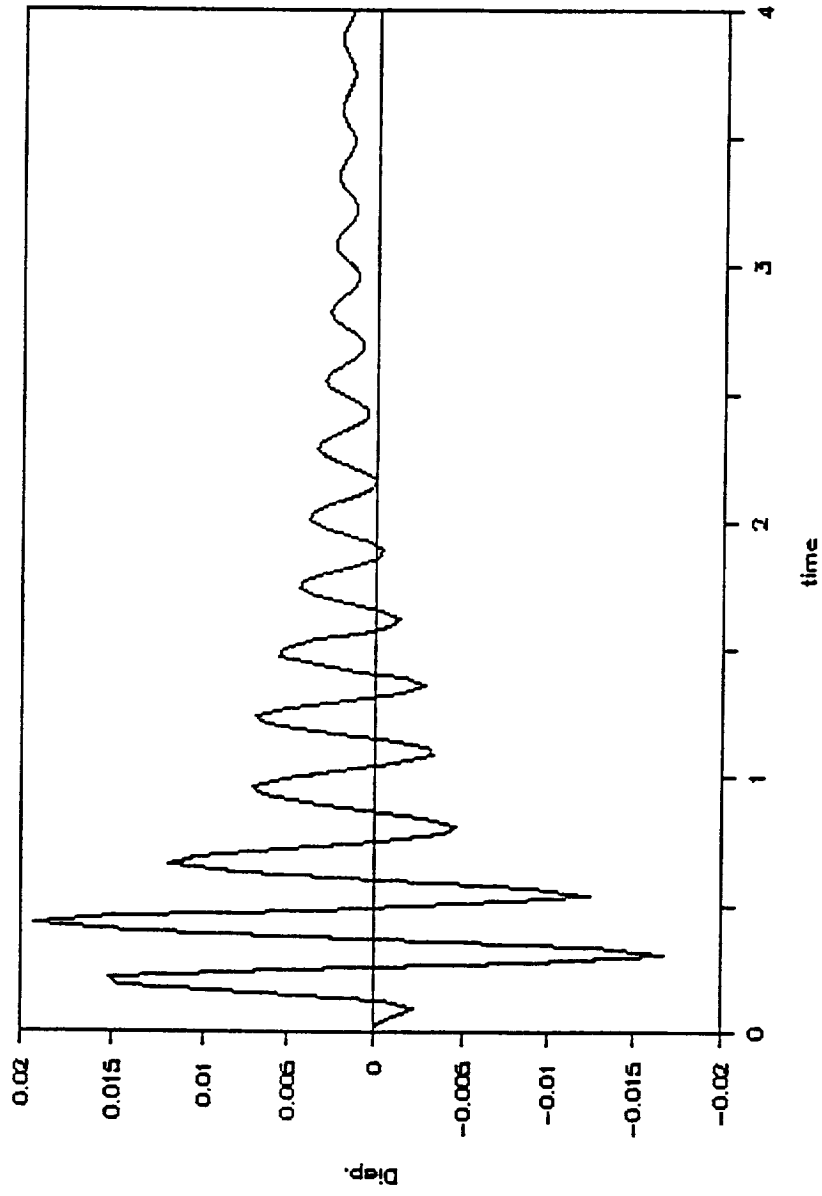


Figure (5.13) - Response at Node 112 in The Z-Direction for The Case of Load Acting at Node 10 in The Z-Direction (High Damping)

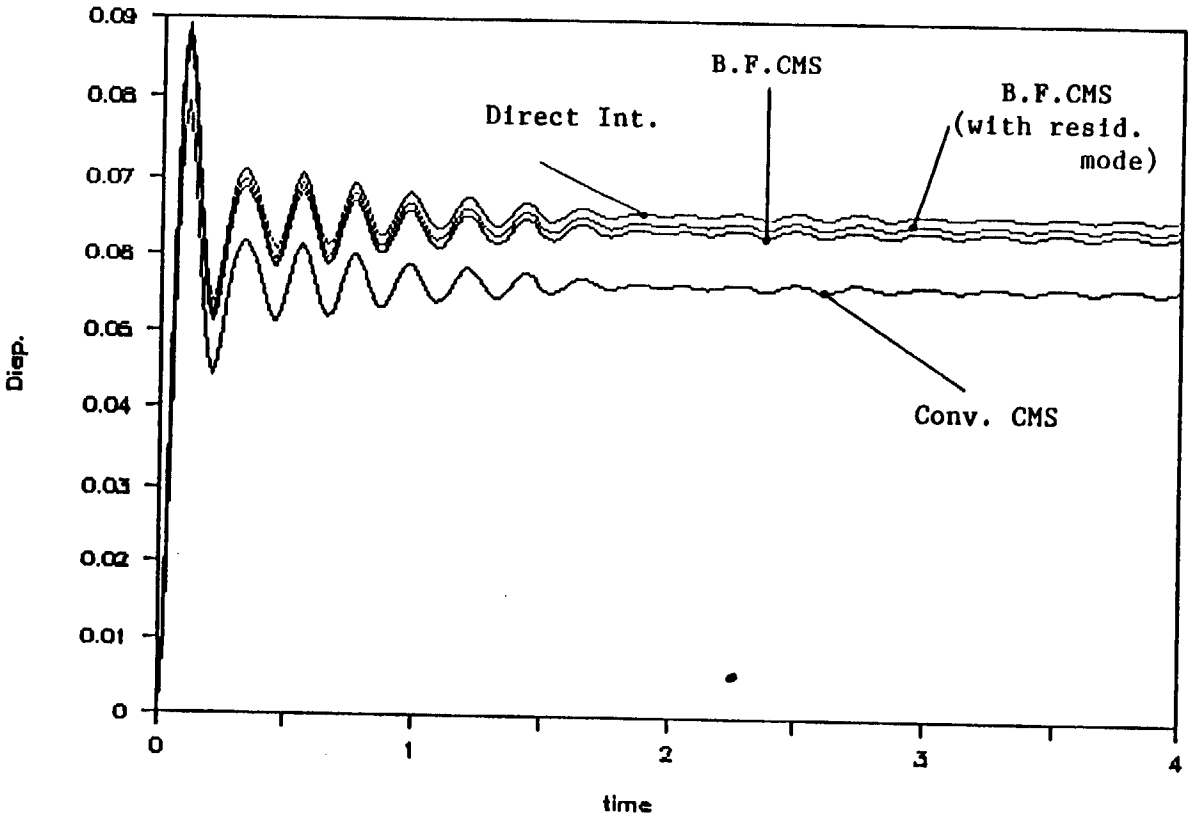


Figure (5.14) - Response at Node 16 in The Y-Direction for The Case of Load Acting at Node 16 in The Y-Direction (Low Damping)

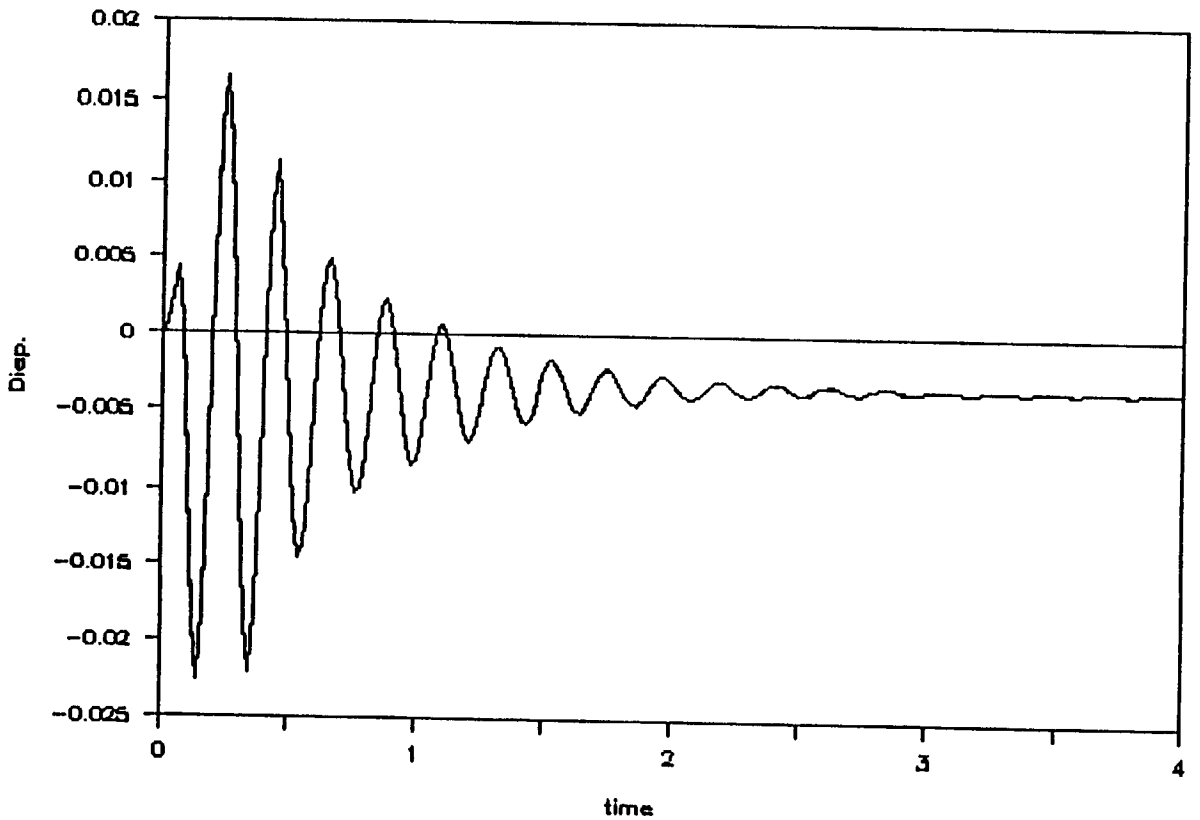


Figure (5.15) - Response at Node 87 in The Y-Direction for The Case of Load Acting at Node 16 in The Y-Direction (Low Damping)

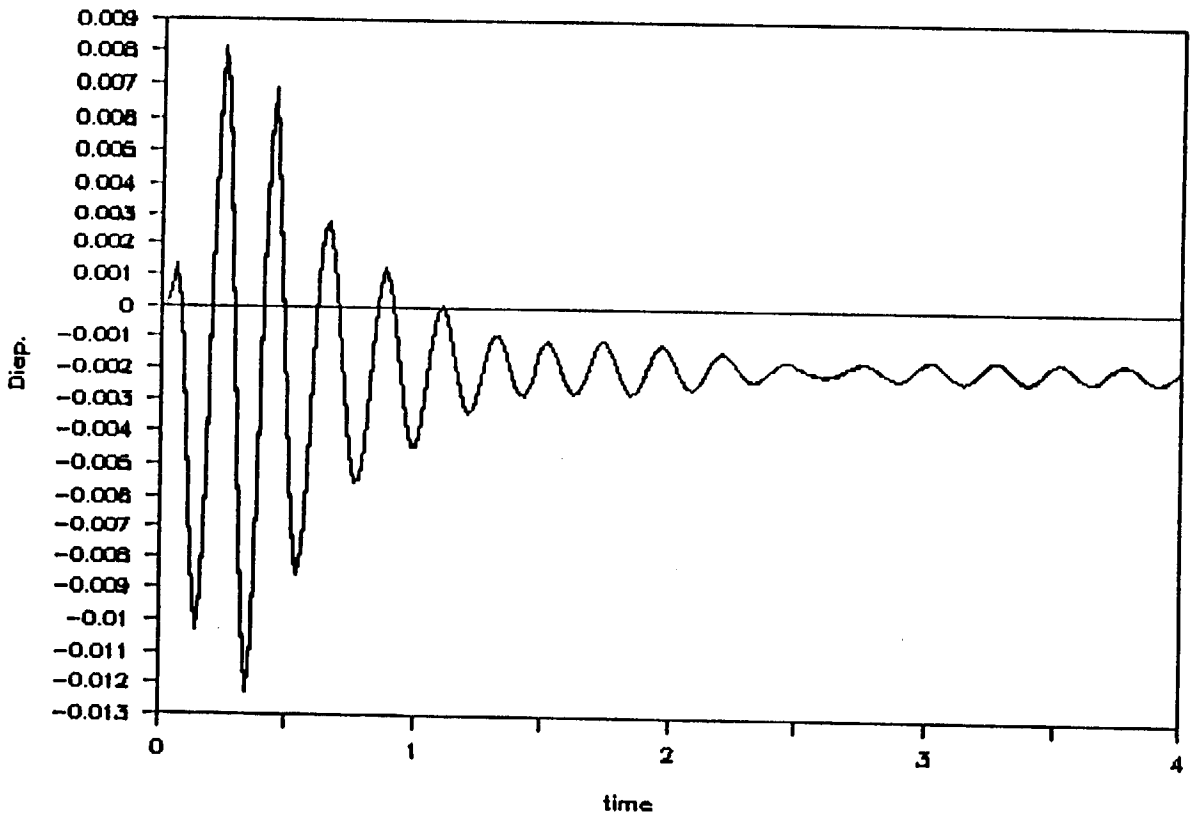


Figure (5.16) - Response at Node 99 in The Y-Direction for The Case of Load Acting at Node 16 in The Y-Direction (Low Damping)

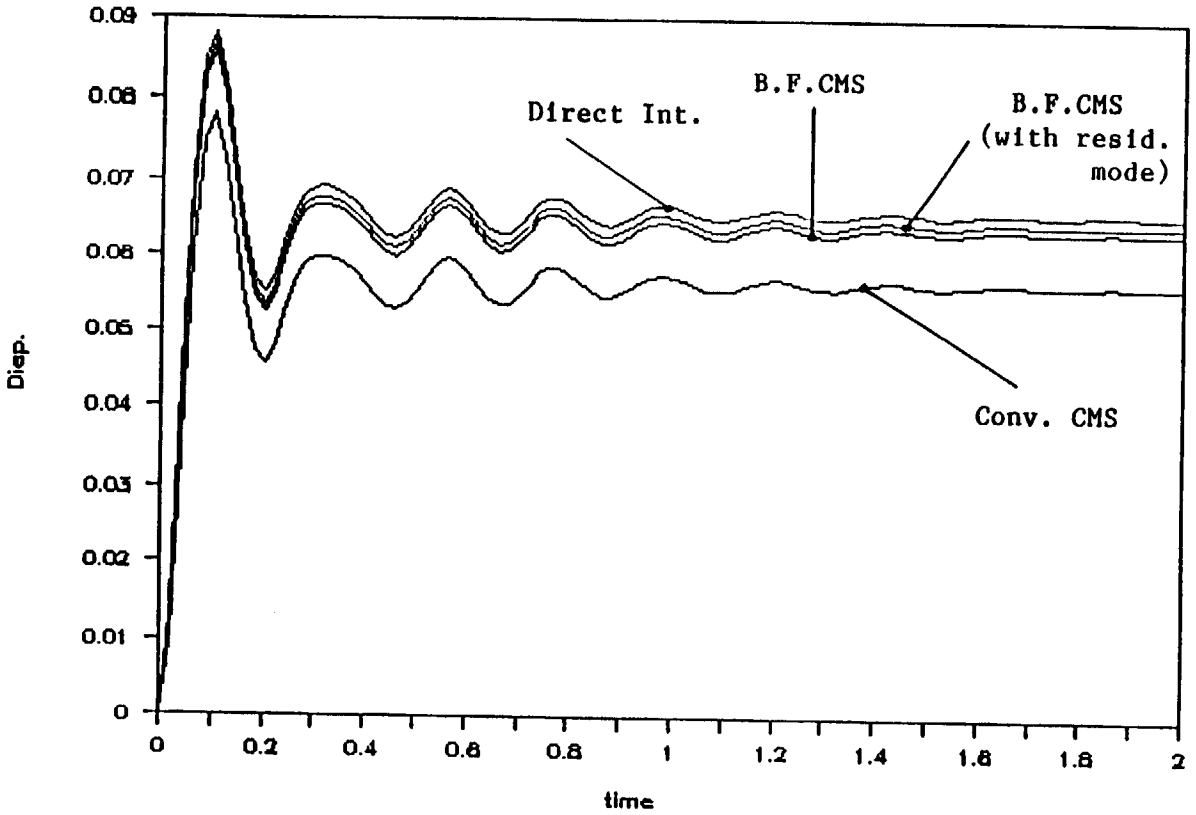


Figure (5.17) - Response at Node 16 in The Y-Direction for The Case of Load Acting at Node 16 in The Y-Direction (High Damping)

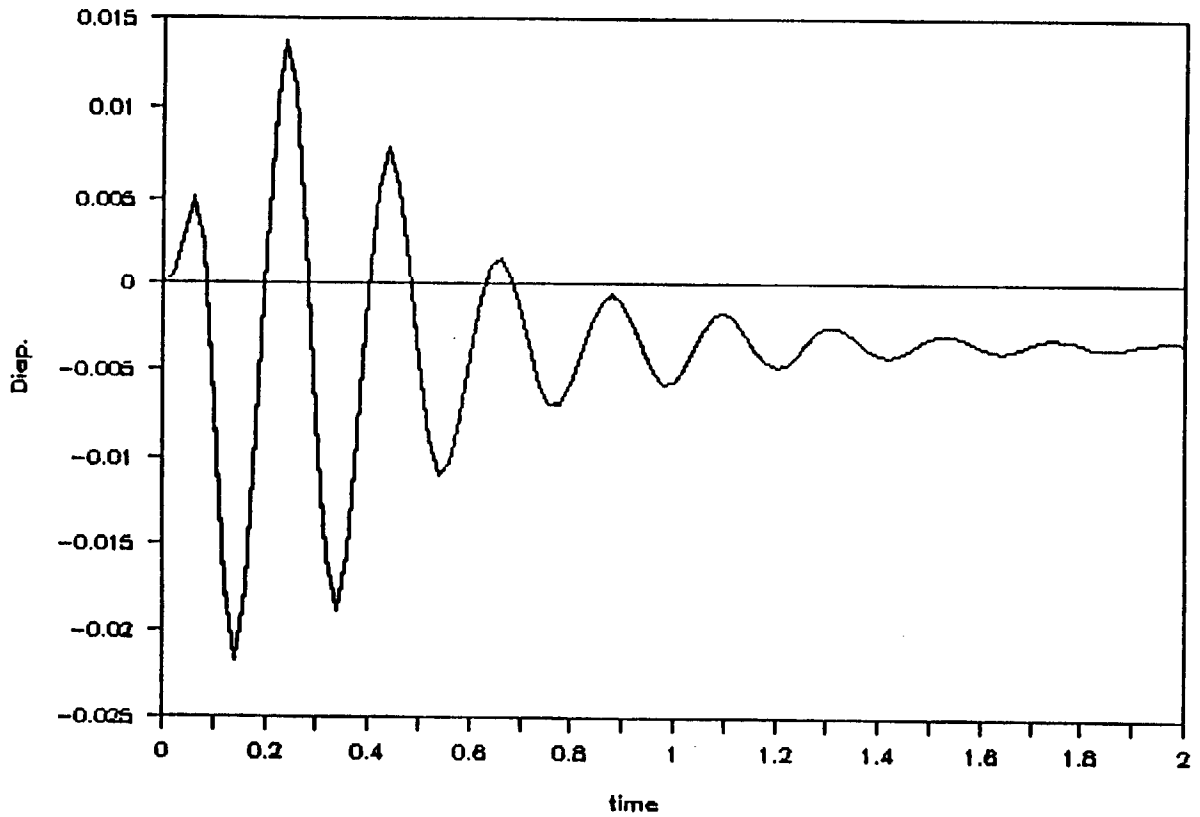


Figure (5.18) - Response at Node 87 in The Y-Direction for The Case of Load Acting at Node 16 in The Y-Direction (High Damping)

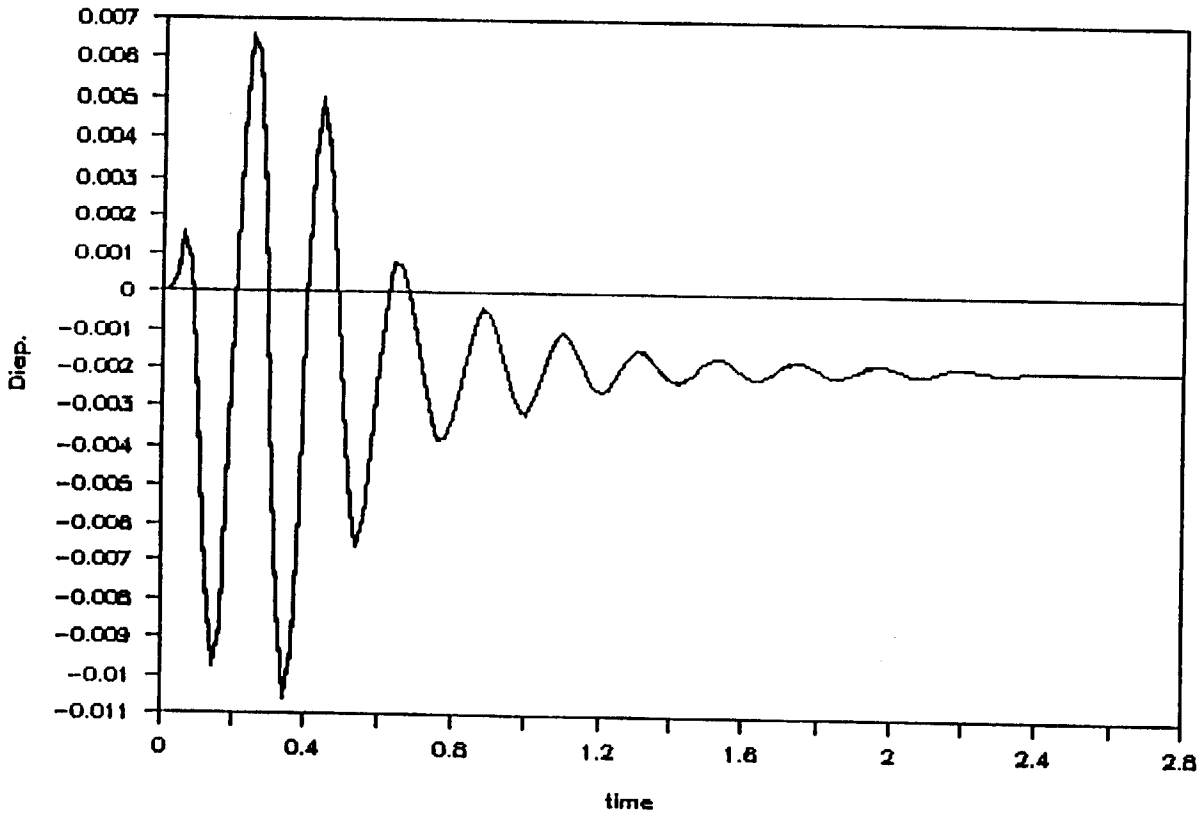


Figure (5.19) - Response at Node 99 in The Y-Direction for The Case of Load Acting at Node 16 in The Y-Direction (High Damping)

Responses	Method		
	Direct Integration	Conven. CMS	B.F. CMS
Z_{13} max.	0.19363	0.19382	0.19355
Z_{13} static	0.10763	0.10763	0.10791
Z_{31} max.	0.45955	0.45959	0.45967
Z_{31} static	0.27631	0.27631	0.27631
Z_{112} max.	0.06321	0.06320	0.06320
Z_{112} static	0.02718	0.02718	0.02718

Table (5.1) - Maximum And Static Responses for Load at Node 31 in Z-Direction With Low Damping.

Responses	Method		
	Direct Integration	Conven. CMS	B.F. CMS
Z_{13} max.	0.18485	0.18503	0.18478
Z_{13} static	0.10699	0.10699	0.10699
Z_{31} max.	0.44217	0.44223	0.44231
Z_{31} static	0.27538	0.27538	0.27538
Z_{112} max.	0.05895	0.05894	0.05894
Z_{112}static	0.02670	0.02670	0.02670

Table (5.2) - Maximum And Static Responses for Load at Node 31 in Z-Direction With High Damping

Responses	Method		
	Direct Integration	Conven. CMS	B.F. CMS
Z_{10} max.	0.08345	0.08313	0.08313
Z_{10} static	0.05265	0.05236	0.05236
Z_{84} max.	0.06310	0.06310	0.06309
Z_{84} static	0.02722	0.02721	0.02721
Z_{112} max.	0.02667	0.02663	0.02663
Z_{112}static	0.00192	0.00192	0.00192

Table (5.3) - Maximum And Static Responses for Load at Node 10 in Z-Direction With Low Damping.

Responses	Method		
	Direct Integration	Conven. CMS	B.F. CMS
Z_{10} max.	0.07661	0.07625	0.07624
Z_{10} static	0.05276	0.05247	0.05246
Z_{84} max.	0.05905	0.05905	0.05905
Z_{84} static	0.02715	0.02715	0.02715
Z_{112} max.	0.01931	0.01928	0.01928
Z_{112}static	0.00184	0.00184	0.00184

Table (5.4) - Maximum And Static Responses for Load at Node 10 in Z-Direction With High Damping.

Responses	Method			
	Direct Integ.	Conven. CMS	B.F. CMS	B.F. CMS With Resid. Mode
Y_{16} max.	0.08908	0.07925	0.08695	0.08781
Y_{16} static	0.06522	0.05583	0.06287	0.06386
Y_{87} max.	-0.02239	-0.02239	-0.02251	-0.02247
Y_{87} static	-0.00353	-0.00353	-0.00353	-0.00353
Y_{99} max.	-0.01219	-0.01212	-0.01231	-0.01229
Y_{99} static	-0.00193	-0.00193	-0.00193	-0.00193

Table (5.5) - Maximum And Static Responses for Load at Node 16 in Y-Direction With Low Damping.

Responses	Method			
	Direct Integ.	Conven. CMS	B.F. CMS	B.F. CMS With Resid. Mode
Y_{16} max.	0.08793	0.07812	0.08578	0.08665
Y_{16} static	0.06555	0.05611	0.06318	0.06417
Y_{87} max.	-0.02161	-0.02159	-0.02172	-0.02168
Y_{87} static	-0.00351	-0.00351	-0.00351	-0.00351
Y_{99} max.	-0.01048	-0.01043	-0.01060	-0.01058
Y_{99} static	-0.00194	-0.00194	-0.00194	-0.00194

Table (5.6) - Maximum And Static Responses For Load at Node 16 in Y-Direction With High Damping.

Chapter (6)

Summary and Conclusions

A new method of component mode synthesis was presented in this work. The new method, namely the boundary flexibility vector of CMS, is based upon a set of static Ritz vectors (boundary flexibility vectors) as generalized displacement shapes for components. The generation of these vectors does not require the solution of the eigenproblem associated with the component, as in the case of conventional methods of CMS. The formulation of the new method for the free vibration problem as well as the forced vibration problem, was presented for both fixed and free-interface components. A comparison of the number of operations required to obtain the boundary flexibility vectors versus normal modes of vibration, was presented. The comparison showed a substantial reduction of the number of required operations by using the boundary flexibility vectors, instead of eigenvectors in CMS.

Numerical examples for the free vibration problem were presented in chapter four. The examples were solved by the boundary flexibility method of CMS and by other different methods. Results indicated that by using the boundary flexibility method, more accurate results could be obtained with a substantial saving in CPU time. The new method was applied to a substantially large structure, in chapter four. The results obtained by the boundary flexibility method of CMS, were in good agreement with finite-element and conventional methods of CMS. The saving in CPU

time gained by using the boundary flexibility vectors, instead of eigenvectors, was computed for this problem. An average saving of CPU time between 75% and 80% could be attained, for generating the same number of vectors.

A numerical example was presented for the forced vibration problem. The results obtained by the boundary flexibility method of CMS, were in good agreement with the results obtained from the direct integration of the equations of motion. In all cases considered, the results obtained by the boundary flexibility method of CMS were almost the same and sometimes superior to those obtained by conventional methods of CMS.

It was proved by this work, that the boundary flexibility method of CMS could be applied to free and forced vibration problems. It was shown that accurate results could be obtained by applying the method, with a substantial saving in CPU time compared to conventional methods of CMS.

A general outline for continuation of this work may be the following:

1- Develop a criteria to specify which of the constraint or attachment modes, are to be chosen to generate the boundary flexibility vectors. This criteria should give an indication of how much a constraint or an attachment mode will contribute to the response.

2- Investigate methods for refining the generated boundary flexibility vectors. One idea suggested is to investigate the

generation of the vectors from actual interface stiffness properties between components, instead of assuming the interface either fixed or free.

References

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

2) Abdallah, A.A., "Generalized Dynamic substructuring Utilizing Mixed Finite Element and Modal Synthesis Techniques and Development of Computer Program DYSTAN 1987", M.S. Thesis, Department of Civil Engineering, Case Western Reserve University, Cleveland, Ohio, January 1988.

3) Craig, R.R. Jr., and Bampton M.C.C., "Coupling of Substructures for Dynamic Analysis", *AIAA Journal*, Vol. 6, No. 7, July 1968, pp. 1313-1319

4) Craig, R.R. Jr., *"Structural Dynamics"*, John Wiley and Sons, New York, 1981.

5) MacNeal, R.H., "A Hybrid Method of Component Mode Synthesis", *Computers & Structures*, Vol. 1, 1971, pp. 581-601.

6) Rubin, S., "Improved Component-Mode Representation for Structural Dynamic Analysis", *AIAA Journal*, Vol. 13, No. 8, August 1975, pp. 995-1006.

7) Martinez, D.R., and Gregory, D.L., "A Comparison of Free Component Mode Synthesis Techniques Using MSC/NASTRAN", *SAND 83-0025*, June 1984.

8) Goldman, R.L., "Vibration Analysis by Dynamic Partitioning", *AIAA Journal*, Vol. 7, No. 6, June 1969, pp. 1152-1154.

9) Hou, S.N., "Review of a Modal Synthesis Technique and a New Approach", *Shock & Vibration Bulletin*, No. 40, Part 4, 1969, pp. 25-39.

10) Dowell, E.H., "Free Vibrations of an Arbitrary Structure in Terms of Component Modes", *Journal of Applied Mechanics*, Vol. 39, No. 3, September 1972, pp. 727-732.

11) Meirovitch, L., *"Computational Methods in Structural Dynamics"*, Sijthoff & Noordhoff International Publishers, The Netherlands, 1980.

12) Wilson, E.L., Yuan, M.-W., and Dickens, J.M., "Dynamic Analysis by Direct Superposition of Ritz Vectors", *Earthquake Engineering and Structural Dynamics* , Vol. 10, No. 6, November 1982, pp. 813-823.

13) Nour-Omid, B., and Clough, R.W., "Dynamic Analysis of Structures Using Lanczos Co-ordinates", *Earthquake Engineering and Structural Dynamics*, Vol. 12, No. 4, July 1984, pp. 565-577.

14) Arnold, R.R., Citerley, R.L., Chargin, M., and Galant, D., "Application of Ritz Vectors for Dynamic Analysis of Large Structures", *Computers & Structures*, Vol. 21, No. 3, 1985, pp. 461-467.

15) Wilson, E.L., and Bayo, E.P., "Use of Special Ritz Vectors in Dynamic Substructure Analysis", *ASCE Journal of Structural Engineering*, Vol. 112, No. 8, August 1986, pp. 1944-1954.

16) Clough, R.W., and Penzien, J., *"Dynamics of Structures"*, McGraw-Hill, New York 1975.

17) Hintz, R.M., " Analytical Methods in Component Modal Synthesis", *AIAA Journal*, Vol. 13, No. 8, August 1975, pp. 1007-1016.

18) Bathe, K.-J., *"Finite Element Procedures in Engineering Analysis"*, Prentice-Hall, New Jersey, 1982.

19) Burden, R.L., and Faires, J.D., *"Numerical Analysis"*, Prindle Weber & Schmidt Publishers, Boston, 1981.

Appendix A

Comparison of The Number of Operations For Generating The Boundary Flexibility Vectors Versus Eigenvectors

The number of operations are derived using some basic criteria in reference [19].

A.1) Extraction of k Eigenpairs:

For the deteminant search and inverse iteration methods, see Bathe [18]

Assume: n = total number of degrees of freedom

k = number of required eigenpairs

y_i = number of iterations in determinant search method required to obtain eigenvalue i

x_i = number of inverse iterations required to obtain eigenvector i

$[K]$= stiffness matrix

$[M]$= mass matrix

$\{Y_i\}$= eigenvector number i

μ = amount of shift applied to stiffness matrix

The following steps are required to obtain each eigenpair, thus they are repeated k times.

Step	Operation	No. of Mult. or Divisions	No. of Add. or Subtractions
1	Apply shift $[K] = [K] - \mu [M]$	n^2	n^2
	Triangularize $[K]$	$\frac{2n^3 + 3n^2 - 5n}{6}$	$\frac{n^3 - n}{3}$
	Total for step 1	$\frac{2n^3 + 9n^2 - 5n}{6}$	$\frac{n^3 + 3n^2 - n}{3}$

Step	Operation	No. of Mult. or Divisions	No. of Add. or Subtractions
	Step 1 is repeated from $j = 1 \ldots . y_i$		
2	Obtain $[K]_i^{-1}$	$\frac{4n^3 - n}{3}$	$\frac{8n^3 - 9n^2 + n}{6}$
3	Compute $\{X\}_{j+1} = [K]_i^{-1}\{Y_j\}$	n^2	$n^2 - n$
	Compute $\{Y\}_{j+1} = [M]\ \{X\}_{j+1}$	n^2	$n^2 - n$
	Compute $\rho_{j+1} = \frac{\{X\}_{j+1}^T\ \{Y\}_j}{\{X\}_{j+1}^T\ \{Y\}_{j+1}}$	$2n + 1$	$2n - 2$
	Compute $\{Y\}_{j+1} = \frac{\{Y\}_{j+1}}{\left[\{X\}_j^T \{Y\}_{j+1}\right]^{1/2}}$	$2n$	$n - 1$
	Total for step 3	$2n^2 + 4n + 1$	$2n^2 + n - 3$
	Step 3 is repeated from $j = 1 \ldots\ldots x_i$		

Then, total number of multiplications and divisions =

$$\frac{k}{3}(4n^3-n) + \sum_{i=1}^{k}\left[\frac{y_i}{6}(2n^3+9n^2-5n) + x_i(2n^2+4n+1)\right] \quad (A.1)$$

Total number of additions and subtractions =

$$\frac{k}{6}(8n^3-9n^2+n) + \sum_{i=1}^{k}\left[\frac{y_i}{3}(n^3+3n^2-n) + x_i(2n^2+n-3)\right] \quad (A.2)$$

A.2) Generation of ℓ Boundary Flexibility Vectors:

Assume $\{q_i\}$ = boundary flexibility vector number i

Step	Operation	No. of Mult. or Divisions	No. of Add. or Subtractions
1	Obtain $[K]^{-1}$	$\frac{4n^3 - n}{3}$	$\frac{8n^3 - 9n^2 + n}{6}$
2	Compute $\{X_i\} = [M]\{q_i\}$	n^2	$n^2 - n$
	Compute $\{q^*_{i+1}\} = [K]^{-1}\{X_i\}$	n^2	$n^2 - n$
	Normalizing $GM = \{q^{**}_{i+1}\}^T [M] \{q^{**}_{i+1}\}$	$n^2 + n$	$n^2 - 1$
	$\{q_{i+1}\} = \frac{1}{GM} \{q^{**}_{i+1}\}$	n	——
	Total for step 2	$3n^2 + 2n$	$3n^2 - 2n - 1$
	Step 2 is repeated for i = 1..... ℓ		
3	Orthogonalizing Compute $c_j = \{q^*_i\}^T [M]\{q_j\}$	$n^2 + n$	$n^2 - 1$
	Compute $\{q^{**}_i\} = \{q^*_i\} - c_j\{q_j\}$	n	n
	Total for step 3 Where j = 1.....i-1 Step 3 is repeated for i = 2 ..... ℓ	$n^2 + 2n$	$n^2 + n - 1$

Total number of repetitions for step number 3 = $(1+2+3....(\ell-1))$

$$= \sum_{k=1}^{\ell-1} k$$

$$= \frac{\ell\,(\ell-1)}{2}$$

Hence total number of multiplications and divisions =

$$\left(\frac{4}{3}\right) n^3 + \left(\frac{\ell^2 + 5\ell}{2}\right) n^2 + \left(\frac{3\ell^2 + 3\ell - 1}{3}\right) n \qquad (A.3)$$

Total number of additions and subtractions =

$$\left(\frac{4}{3}\right) n^3 + \left(\frac{\ell^2 + 5\ell - 3}{2}\right) n^2 + \left(\frac{3\ell^2 - 15\ell + 1}{6}\right) n - \left(\frac{\ell^2 + \ell}{2}\right) \qquad (A.4)$$

Appendix B

Flexibility Matrix of Components

In Terms of Stiffness Submatrices

$$\text{Assume stiffness matrix } [k] = \begin{bmatrix} [k_{cc}] & [k_{ci}] \\ [k_{ic}] & [k_{ii}] \end{bmatrix} \tag{B.1}$$

$$\text{flexibility matrix } [g] = \begin{bmatrix} [g_{cc}] & [g_{ci}] \\ [g_{ic}] & [g_{ii}] \end{bmatrix} \tag{B.2}$$

Since $[g] = [k]^{-1}$, then

$$\begin{bmatrix} [k_{cc}] & [k_{ci}] \\ [k_{ic}] & [k_{ii}] \end{bmatrix} \begin{bmatrix} [g_{cc}] & [g_{ci}] \\ [g_{ic}] & [g_{ii}] \end{bmatrix} = \begin{bmatrix} [I_{cc}] & [0_{ci}] \\ [0_{ic}] & [I_{ii}] \end{bmatrix} \tag{B.3}$$

where $[I_{cc}]$ = identity matrix of order c

where $[I_{ii}]$ = identity matrix of order i

Equation (B.3) is reduced to the following four matrix equations

$$[k_{cc}][g_{cc}] + [k_{ci}][g_{ic}] = [I_{cc}] \tag{B.4}$$

$$[k_{cc}][g_{ci}] + [k_{ci}][g_{ii}] = [0_{ci}] \tag{B.5}$$

$$[k_{ic}][g_{cc}] + [k_{ii}][g_{ic}] = [0_{ic}] \tag{B.6}$$

$$[k_{ic}][g_{ci}] + [k_{ii}][g_{ii}] = [I_{ii}] \tag{B.7}$$

From equation (B.6)

$$[g_{ic}] = -[k_{ii}]^{-1}[k_{ic}][g_{cc}] \tag{B.8}$$

Substitute in equation (B.4), then

$$[g_{cc}] = \left[[k_{cc}]-[k_{ci}][k_{ii}]^{-1}[k_{ic}]\right]^{-1} \tag{B.9}$$

Hence

$$[g_{ic}] = -[k_{ii}]^{-1}[k_{ic}]\left[[k_{cc}]-[k_{ci}][k_{ii}]^{-1}[k_{ic}]\right]^{-1} \qquad (B.10)$$

Similarly, by solving equations (B.5) and (B.7) simultaneously

$$[g_{ci}] = [g_{ic}]^T = -\left[[k_{cc}]-[k_{ci}][k_{ii}]^{-1}[k_{ic}]\right]^{-1}[k_{ci}][k_{ii}]^{-1} \qquad (B.11)$$

$$[g_{ii}] = [k_{ii}]^{-1} + [k_{ii}]^{-1}[k_{ic}]\left[[k_{cc}]-[k_{ci}][k_{ii}]^{-1}[k_{ic}]\right]^{-1}[k_{ci}][k_{ii}]^{-1} \qquad (B.12)$$

Appendix C

Proportional Damping Matrix

Proportional damping matrix [C], (see Bathe [18] and Clough and Penzien [16]), is given by

$$[C] = \alpha\,[M] + \beta\,[K] \qquad (C.1)$$

where [M] = mass matrix

[K] = stiffness matrix

α and β are constants to be determined from two given damping ratios corresponding to two unequal frequencies of vibrations.

Assume mode shape $\{\phi_i\}$ has a frequency ω_i. Pre and post multiply both sides of equation (C.1) by $\{\phi_i^T\}$ and $\{\phi_i\}$ respectively

$$2\,\omega_i\xi_i = \{\phi_i^T\}\,(\alpha\,[M] + \beta\,[K])\,\{\phi_i\}$$

$$2\,\omega_i\xi_i = \alpha + \beta\,\omega_i^2 \qquad (C.2)$$

where ξ_i = damping ratio of mode shape i.

By substituting, in equation (C.2), any two unequal frequencies and their corresponding assumed damping ratios, α and β can be calculated from the resulting two simultaneous equations. The damping ratio of any other natural frequency is obtained from:

$$\xi_i = \frac{\alpha + \beta\,\omega_i^2}{2\,\omega_i} \qquad (C.3)$$