ort No. FHWA-RD-74-3

INVESTIGATION OF THE EFFECTIVENESS OF EXISTING GE DESIGN METHODOLOGY IN PROVIDING ADEQUATE UCTURAL RESISTANCE TO SEISMIC DISTURBANCES.

Phase II: Analytical Investigations of the Seismic Response of Long Multiple-Span Highway Bridges

W. Tseng and J. Penzien

January 1974
Final Report

This document is available to the public through the National Technical Information Service, Springfield, Virginia 22151

Prepared for
FEDERAL HIGHWAY ADMINISTRATION
Offices of Research & Development
Washington, D.C. 20590

NOTICE

TECHNICAL REPORT STANDARD TITLE PAGE

1. Report No. FHWA-RD-74-3
2. Government Accession No.
3. Recipient's Catalog No.

4. Title and Subtitle An Investigation of the Effectiveness of Existing Bridge Design Methodology in Providing Adequate Structural Resistance to Seismic Disturbances. Phase II: Analytical Investigations of the Seismic Response of Long Multiple-Span Highway Bridges.

5. Report Date January 1974
6. Performing Organization Code

7. Author(s) W. Tseng and J. Penzien

8. Performing Organization Report No.

9. Performing Organization Name and Address
University of California
Campus Research Office
118 California Hall
Berkeley, California 94720

10. Work Unit No. 35A2-012
11. Contract or Grant No. DOT-FH-11-7798
13. Type of Report and Period Covered Phase II - Final Report

12. Sponsoring Agency Name and Address
Office of Research and Development
Federal Highway Administration
U. S. Department of Transportation
Washington, D. C. 20590

u.s.

14. Sponsoring Agency Code S0222

15. Supplementary Notes

FHWA Contract Manager: J. D. Cooper (HRS-11)

16. Abstract

This report is the second in a series to result from the investigation, "An Investigation of the Effectiveness of Existing Bridge Design Methodology in Providing Adequate Structural Resistance to Seismic Disturbances", sponsored by the U. S. Department of Transportation, Federal Highway Administration. Descriptions are given to the analytical investigations of the seismic response of long, multiple-span, highway bridge structures of the type which suffered heavy damages during the San Fernando earthquake of February 9, 1971.

Linear and nonlinear mathematical modeling of this type of bridge structural system is presented. A three-dimensional elasto-plastic flexural column model suitable for modeling the coupled inelastic behavior of reinforced concrete bridge columns is described in detail. A nonlinear mathematical model for simulating the nonlinear discontinuous behavior of bridge expansion joints is also presented. Then, appropriate linear and nonlinear analytical procedures are described for determining the seismic response of this type of bridge structure.

Nonlinear seismic responses are presented for three prototype long, multiple-span, reinforced concrete highway overcrossing structures. Parameter studies carried out on these bridges are described, and the analytical results are presented, discussed, and correlated with the apparent prototype behavior observed during the San Fernando earthquake.

Finally, based on the analytical seismic responses presented, general conclusions and recommendations related to the fundamental seismic design methodology of long, multiple-span highway bridges are deduced and summarized.

This report is the second in a series. The others in the series are:

Phase No.	FHWA No.	Short Title	NTIS(PB) No. (if available)
I	73-13	Literature Survey	(not yet available)

17. Key Words Earthquake, Bridge Seismic Analysis, Structural Analysis, Seismic Design Criteria,

18. Distribution Statement No restrictions. This document is available through the National Technical Information Service, Springfield, Virginia 22151

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	329	

Form DOT F 1700.7 (8-69)

PREFACE

The investigation with interpretation as described in this report was sponsored by the U. S. Department of Transportation, Federal Highway Administration, under Contract No. DOT-FH-11-7798 covering the period July 1, 1971 through September 30, 1974.

The general investigation called for in this contract is under the supervision and technical responsibility of Professors R. W. Clough, W. G. Godden, and J. Penzien. Professor Penzien acts as principal investigator.

ACKNOWLEDGEMENT

The authors wish to express their sincere appreciation to the California State Division of Highways, Department of Public Works, for providing the engineering data of the bridge structures studied in this investigation. Special thanks are due to Mr. Roy A. Imbsen, Associate Bridge Engineer, Bridge Department, State of California, for his cooperation and assistance in preparing computer program input data and providing structural diagrams. This assistance greatly facilitated the progress of the investigation reported herein.

TABLE OF CONTENTS

TABLE OF CONTENTS (cont.)

LIST OF TABLES

LIST OF TABLES (cont.)

LIST OF FIGURES

LIST OF FIGURES (cont.)

LIST OF FIGURES (cont.)

LIST OF FIGURES (cont.)

LIST OF FIGURES (cont.)

LIST OF FIGURES (cont.)

I INTRODUCTION

In the past, numerous highway bridges have suffered extensive damages due to strong motion earthquakes [1,2]*. The older bridges, consisting of single or multiple simple truss or girder spans supported on massive piers and abutments, were particularly vulnerable to the action of strong ground motions. Seismic damages were most commonly caused by foundation failures resulting from excessive ground deformation and/or loss of stability and bearing capacity of the foundation soils. As a direct result, the sub-structures often tilted, settled, slid, or even overturned; thus severe cracking or complete failure was often experienced. These large support displacements also caused relative shifting of and damage to the super-structures, induced failures within the bearing supports, and even caused spans to fall off their supports. It is significant to note that very little damage occurred to these older structures as a direct result of structural vibration effects.

Certain types of modern highway bridges may, on the other hand, be quite susceptible to damage from strong ground vibration effects. This fact became very evident during the San Fernando earthquake of February 9, 1971, when numerous reinforced concrete highway bridges suffered severe damages [3,4]. These structures generally fall into two classifications (1) long, multiple-span bridges, and (2) short, single or multiple span bridges.

The long, multiple-span bridges usually consist of either straight or curved continuous reinforced concrete box girder decks supported on reinforced concrete columns and diaphragm abutments. The decks which are continuous over many spans are provided with intermediate expansion joints

---

* Numbers in brackets refer to Bibliography numbers.

having restrainer bars tieing adjacent spans together. Although the overall bridge system in this case is an integrated space frame, significant structural discontinuities result from the presence of these joints. The lateral rigidity of the overall continuous deck system is therefore greatly reduced. Consequently, the lateral resistance of the total bridge structure is provided mainly by its supporting columns. These columns are often of various lengths consistent with the ground surface profile. Structure-foundation interaction effects may be relatively small for the taller bridges but can be quite large for the shorter bridges in which case they must be included in the analysis. In any case, the dynamic response of abutment backfills have negligible effect on the overall dynamic response of these bridges.

During a severe earthquake, long multiple-span bridge structures of this type can develop large amplitude oscillations. These oscillations may cause large cyclic inelastic deformations, of a coupled form, to develop in the columns. Also, cyclic slippage of the Coulomb type can take place in the expansion joints causing multiple impacts and separations to occur. These separations may be sufficiently large to cause yielding in the tension restrainer bars and, as experienced in San Fernando, may even cause deck spans to fall off their supports. While yielding of the restrainer bars can be tolerated, collapse of the bridge system is unacceptable.

From the above description, it is quite apparent that the dynamic response characteristics of long multiple-span reinforced concrete bridge structures are very complex due to the nonlinear, discontinuous behavior in the expansion joints under combined loading conditions. Because these characteristics are unique, one cannot compare the seismic response of this type of structure directly with the dynamic response of buildings under similar excitation conditions. Therefore, it is essential that the dynamic response and failure characteristics of these bridges be investigated separately by analytical and experimental means and that the results of these investigations be fully correlated with field evidence.

Turning our attention now to the second class of bridge structures mentioned above, i.e., short, single or multiple span bridges, it is immediately apparent that structure-foundation interaction effects greatly influence the seismic response characteristics of this class of structure. Because they lack intermediate expansion joints, the deck responds essentially as a rigid body transferring its seismic loads through the supporting columns and abutments to the foundation.

The seismic response of abutment backfills may be in phase with the response of the bridge deck in such a manner that large dynamic active pressures develop which are additive to the seismic forces in the deck, thus greatly amplifying the dynamic response of the complete structural system. If the bridge should be skewed, as is often the case, the active forces on abutment can develop large twisting moments about a vertical axis through the elastic center of the bridge. These moments may become sufficiently large to cause failures of columns and abutments [1,2,3].

Obviously, any seismic analysis of this type of structure must consider foundation and abutment backfill soils as part of the complete structural system, if realistic results are to be obtained.

As previously mentioned, the 1971 San Fernando earthquake caused heavy damages to both classes of modern reinforced concrete bridges described above which were designed by the tradiational elastic approach, i.e., by the equivalent static seismic coefficient method similar to that previously adopted for buildings [5]. Because of this experience, it was immediately apparent that the seismic requirements used were inadequate and that the design methodology should be critically examined. Action was quickly taken following the earthquake to correct certain design deficiencies [6]; however, the basic static approach to design still remains in effect.

Recognizing the urgent need for both theoretical and experimental research related directly to seismic effects on bridge structures, a three-year investigation entitled "An Investigation Of The Effectiveness of Existing Bridge Design Methodology in Providing Adequate Structural Resistance to Seismic Disturbances" was initiated in 1971 within the Earthquake Engineering Research Center, University of California, Berkeley, under the sponsorship of the U. S. Department of Transportation, Federal Highway Administration. This investigation consists of the following five phases:

(1) A thorough review of the world's literature on seismic effects on highway bridge structures including damages to bridges during the San Fernando earthquake of February 9, 1971.

(2) An analytical investigation of the dynamic response of long, multiple-span, highway overcrossings of the type which suffered heavy damages during the 1971 San Fernando earthquake.

(3) An analytical investigation of the dynamic response of short, single and multiple span, highway overcrossings of the type which suffered heavy damages during the 1971 San Fernando earthquake.

(4) Detailed model experiments on a shaking table to provide dynamic response data similar to prototype behavior which can be used to verify the validity of theoretical response predictions.

(5) A thorough comparison of dynamic response obtained from analyses, experiments, and field experience followed by the preparation of recommendations for changes in seismic design specifications and methodology as necessary to provide adequate protection against future earthquakes.

The final report covering Phase 1 has recently been published [1,2] and the present report is the final report covering Phase 2.

The primary objectives of the investigation in Phase 2 are to develop suitable analytical procedures for linear and nonlinear three-dimensional earthquake response analysis of long, multiple-span, modern highway bridges of the type which suffered heavy damages during the 1971 San Fernando earthquake and to identify and investigate the important parameters which significantly affect the response of this type of bridge system.

To achieve these objectives, linear and nonlinear mathematical models suitable for modelling the dynamic characteristics of this type of bridge have been defined and analytical procedures and computer programs have been developed for determining the seismic response of the complete bridge system to arbitrary, but prescribed, earthquake excitations.

Linear and nonlinear seismic analyses have been performed on three typical major overcrossings using earthquake ground motions of several intensities. These bridges are (1) the South Connector Overcrossing of the Golden State freeway and Antelope Valley freeway interchange (5/14 South Connector Overcrossing), (2) the proposed curved Figueroa Street Undercrossing Connector in the Los Angeles area, and (3) a straightened version of the Figueroa Street Undercrossing Connector. Various parameter studies have been carried out for these bridges and the results are discussed and correlated with the apparent behavior of this type of structure during the San Fernando earthquake. In particular, the causes of collapse of the 5/14 South Connector Overcrossing are identified and examined.

Finally, based on the analytical results obtained, some general conclusions and recommendations related to fundamental seismic design methodology have been formulated.

Chapter 2 of this report describes the linear and nonlinear mathematical modelling of typical, long, multiple-span, modern highway overcrossings; Chapter 3 presents in detail a three-dimensional elasto-plastic flexural column model suitable for representing the coupled yielding behavior of columns under large cyclic deformations; Chapter 4 defines an appropriate nonlinear mathematical model for expansion faults which simulate the coupled effects of impact slippage with Coulomb friction and separation under the action of elasto-plastic restrainer bar forces; Chapter 5 and 6 describe the analytical procedures and computer programs; Chapter 7 presents the numerical results of the investigation; Chapter 8 gives a discussion of the numerical results, and Chapter 9 summarizes the conclusions and recommendations.

II MATHEMATICAL MODELLING OF LONG MULTIPLE-SPAN BRIDGES

A. STRUCTURAL SYSTEM

The structural system of this type of bridge consists of a multiple-span continuous box girder deck supported on and rigidly connected to reinforced concrete columns and diaphragm abutments. The deck may be straight or curved in both horizontal and elevation views and is supported at discrete locations along its longitudinal axis by either centrally positioned columns or by transverse rows of multiple columns. Intermediate expansion joints divide the deck into several segments. Figure 1 shows the structural lay-out of the 5/14 South Connector Overcrossing which is typical of this type of bridge.

The entire structural system exhibits characteristics of a continuous space frame. Its dynamic response to earthquake excitations is of lower mode type; hence, a mathematical model of discrete form can be used to approximate the continuous system. This form of modelling leads to a system having a finite number of degrees of freedom.

Following the standard finite element procedure, these degrees of freedom are chosen as the nodal displacements of the discrete finite element model. For a three dimensional model, each nodal point usually has 6 degrees of freedom, i.e., 3 translation components and 3 rotation components. Internal constraints may however reduce this number at some nodal points.

For purposes of dynamic analysis, the stiffness, mass, and damping properties of each finite element must be realistically defined and the characteristics of each expansion joint must be properly modelled. These features of the overall system are described in detail in the subsequent sections of this report.

B. STIFFNESS IDEALIZATION

The finite element idealization of the complete bridge system results in a stiffness matrix which is an assemblage of the generalized stiffness matrices for individual elements, i.e., in symbolic form:

$$\underline{K} = \sum_{i=1}^{N} \underline{k}_i \tag{1}$$

where $\underline{K}$ is the total stiffness matrix for the entire bridge, $\underline{k}_i$ is the stiffness matrix for element i, and N is the total number of elements in the system.

For small amplitude response, the bridge system may be modelled by a set of linear elastic elements; however, when subjected to high amplitude response as occurs during severe earthquakes, certain critical regions of the structure may undergo large cyclic inelastic deformations. Therefore, nonlinear finite elements must be chosen for the mathematical model which have realistic nonlinear hysteretic force-deformation characteristics. The stiffnesses of these elements are time dependent and are functions of element deformations and deformation histories. Usually, they are linearized for analysis in a piecewise fashion using tangent stiffnesses at discrete times. Thus, the total stiffness matrix for the entire structure may be written in the symbolic form:

$$\underline{K}_t = \sum_{i=1}^{N} \underline{k}_{ti} \tag{2}$$

where $\underline{K}_t$ is the total stiffness matrix at time t and $\underline{k}_{ti}$ is the stiffness matrix for element i at time t. It should be noted that all nonlinear elements referred to in this report account for material nonlinearities only. Nonlinearities arising from large geometry changes are not included as they are negligible for the levels of response considered acceptable in these structural systems.

1. Decks - The decks of most modern reinforced concrete highway bridges are of the curved box girder type as shown in Fig. 2a. The performance of these girders under static loadings has been investigated extensively [7 - 10]. The analytical methods employed used various types of elements for the deck structure such as simple

beam elements, folded plate elements, and quadralateral plate elements [7]. It has been shown that when determining internal stress distributions in constituent flanges and webs of box girders under localized loadings that rather elaborate methods of analysis, such as finite segment analysis or finite element analysis, must be used. However, when the external loadings are rather uniformly distributed and when only resultant forces on transverse cross-sections, i.e., 3 components of force and 3 components of moment, are required, a simple beam analysis is usually sufficient to yield accurate results. Thus, the mathematical model used herein for seismic analysis represents the deck by a series of either straight or curved beam elements as shown in Fig. 2b.

The stiffness matrix of a linear elastic straight beam element is found in most modern textbooks on matrix structural analysis [11] and the stiffness matrix of a linearly elastic, circularly curved, beam element can be found in papers by Morris [12] and Tezcan, et.al. [13].

Since a typical box girder deck is extremely stiff and strong in comparison with its supporting columns and abutments, the high amplitude bridge response produced during severe ground shaking will be caused primarily by deformations in the columns, abutments, and expansion joints. The deck will remain elastic and, therefore, can be modelled by linear elastic elements. Nonlinear yielding elements must, however, be used for columns, abutments, and the tension restrainer tie bars used in expansion joints.

2. Columns and Abutments - The structural behavior of columns can be adequately modelled using simple beam elements; see Fig. 2c. Because of large amplitude response, coupled inelastic flexural deformations may occur in these members. Therefore, nonlinear beam elements which realistically characterize the inelastic hysteretic behavior must be used. An appropriate elasto-plastic element under combined loading has been defined for this purpose and described in detail in Chapter 3.

The abutments of this type of bridge are usually of the diaphragm type which act in the manner of shear wall elements. Failures are likely to be of the shear type causing excessive damage. Therefore, such failures should be avoided through proper design. Consequently, elastic behavior is always assumed for abutments in the present analysis.

3. Foundations - Complex mathematical models have been used for structure foundations in previous investigations [14, 15, 16]. However, a simpler and more approximate model has been adopted herein. This model consisting of 3 translational and 3 rotational springs is used to connect the base of each column and abutment to a rigid foundation where the seismic excitation is fully prescribed; see Fig. 2c. For linear analysis, the stiffnesses of these springs can be evaluated using linear elastic half-space theory similar to that used in the reference [15], or methods reported in the literature [17].

For large amplitude response, the foundation soils may undergo inelastic deformations of the hysteretic type. In this case, the six foundation springs should be bilinear hysteretic springs. The elastic stiffnesses of these springs can be obtained using standard methods; however, their yielding stiffnesses and yield values can be established only through extensive experimental studies on the dynamic properties of the foundation soils [15].

A very simple method, which is often used for modelling foundation flexibility, is to assume that the columns and abutments extend below the ground surface to specified depths at which fully flexity conditions are believed to have been reached. The ground excitations are then fully prescribed at these effective depths.

C. MASS IDEALIZATION

The continuous mass of the bridge structural system is modelled in discrete form by lumping element masses at their end nodal points. Since inertia forces are associated with each of the six degrees of freedom at a nodal point, each lumped mass must be assigned appropriate moments of inertia about its own three coordinate axes (x,y,z). It should also be noted that when conducting nonlinear dynamic analyses, the instantaneous stiffness matrix may become singular in which case it is essential that mass moments of inertia be assigned to each rotational degree of freedom. Following this procedure, a diagonal mass matrix $\underline{m}_i$ is established for each element i (i = 1,2, ---,N). The diagonal mass matrix for the complete bridge system can then be assembled and expressed in the symbolic form:

$$\underline{M} = \sum_{i=1}^{N} \underline{m}_i \tag{3}$$

In determining overall dynamic response, this lumped mass method has been found to be quite adequate for analysis purposes [18].

D. DAMPING IDEALIZATION

Velocity dependent damping in the structural system is represented by a generalized damping matrix associated with the finite degrees of freedom permitted in the mathematical model. This matrix can be derived by consistent procedures similar to those used in deriving the stiffness matrix provided the internal damping mechanisms within each element are specified. The structural damping matrix for the complete structural system would then be of the symbolic form:

$$\underline{C} = \sum_{i=1}^{N} \underline{c}_i \tag{4}$$

where $\underline{c}_i$ is the damping matrix for the ith element.

In practice, however, it is very difficult to establish the basic characteristics of damping in the individual elements. Therefore, it is customary to assume that the damping forces of each element consists of one set which are proportional to the velocities of each mass point and a set which are proportional to the rates of deformation in the element [18]. Thus, the element damping matrix can be represented as:

$$\underline{c}_i = \alpha_i \underline{m}_i + \beta_i \underline{k}_i \tag{5}$$

where α_i and β_i are the scalar proportionality constants for the ith element. If the entire structure is made of the same basic material, it is reasonable to assume similar damping characteristics for each element in the system, i.e., α_i and β_i are independent of i in which case, combining Eqs. (4) and (5), the structural damping matrix becomes:

$$\underset{\sim}{C} = \alpha \underset{\sim}{M} + \beta \underset{\sim}{K} \tag{6}$$

where

$$\alpha = \alpha_i \ ; \ \beta = \beta_i \qquad (i=1,2,---,N) \tag{7}$$

In the case of linear response, it can be shown that the two generalized parameters α and β can be determined after assigning damping ratios to the first two natural modes of vibration [19].

For nonlinear dynamic analysis, the viscous damping properties of the structure are much more difficult to access. However, it is reasonable to make assumptions similar to the elastic case, i.e., to assert one set of damping forces proportional to the nodal velocities, as assumed in the elastic case, and a second set which are proportional to the rates of elastic deformation only. In this case, the structural damping matrix takes the form:

$$\underset{\sim}{C}_t = \alpha \underset{\sim}{M} + \beta \underset{\sim}{K}_t \tag{8}$$

where α and β are the constants given in Eq. (7).

E. EXPANSION JOINT IDEALIZATION

A typical expansion joint is shown in Figs. 3a and 3b. The bridge deck on one side of the expansion joint rests on a ledge of the deck on the opposite side. A number of transverse shear keys exist on the support ledge which prevent relative transverse motion of the deck within the joint. Longitudinal restrainer bars are attached across the expansion joint tieing together the end diaphragms of the adjacent deck spans. These tie bars are placed in such a way that they carry only tensile forces after a small tie gap is closed by joint separation. Another small gap is provided in the joint to allow for thermal expansion of the deck spans. Vertical restrainers are installed in the expansion joint to prevent lifting of one span off the other along the support ledge.

Referring to Figs. 3a and 3b, the expansion joint characteristics may be summarized as follows:

(1) Relative translation of the deck spans along the longitudinal x axis cause impacts when the gap between adjacent spans is closed, slippage on the support ledge of the Coulomb type, and extensions of the longitudinal restrainer bars when joint separations exceed tie bar gaps.

(2) Relative translation of the spans along the transverse y axis is prevented by the shear keys.

(3) Relative translation of the spans along the vertical z axis is constrained by vertical restrainer bars.

(4) Relative rotation of the spans about the x axis is constrained by the vertical restrainer bars.

(5) Relative rotation of the spans about the y axis is freely permitted.

(6) Relative rotation of the spans about the z axis is coupled to the relative translation of spans along the x axis and may be partially or fully constrained by a combination of closure of the gap between decks, actions of the longitudinal restrainer bars, and the presence of Coulomb friction.

Because of these characteristics, the expansion joint behaves as a nonlinear discontinuous element causing different degrees of constraint during the motion of the structure. Therefore, the total number of degrees of freedom in the complete system changes from time to time during the response causing complexities to arise in the nonlinear dynamic analysis procedures.

Two limiting cases of expansion joint behavior, providing minimum and maximum constraint, are assumed for linear dynamic analysis. These two linear cases are achieved by neglecting the effects of impact and longitudinal restrainer bar constraints and by assuming either zero or infinite Coulomb friction. The infinite friction case prevents relative translation of deck spans along the longitudinal x axis and prevents relative rotation about the z axis; thus, it characterizes the prototype dynamic behavior for low amplitude response. The zero friction case, on the other hand, approximates prototype behavior for high amplitude response following failure of the short longitudinal restrainer bars but prior to yielding in the columns.

The overall bridge structure is very stiff in the infinite friction case due to arching action of the deck from abutment to abutment which carries most of the lateral loads. It is however, relatively flexible in the zero friction case since arching action cannot occur and the lateral loads must be carried in the supporting columns.

F. LINEAR AND NONLINEAR MATHEMATICAL MODELS

In summary, the mathematical model used for linear dynamic analysis consists of (1) linear elastic straight beam elements, (2) linear elastic curved (circular) beam elements, (3) linear boundary spring elements, and (4) linear expansion joint elements, while the model used for nonlinear analysis consists of (1) linear elastic straight beam elements, (2) linear elastic curved beam elements, (3) elasto-plastic flexural beam elements, (4) bilinear hysteretic boundary spring elements, and (5) nonlinear expansion joint elements. Figure 4 shows the discrete parameter model for the 5/14 South Connector Overcrossing.

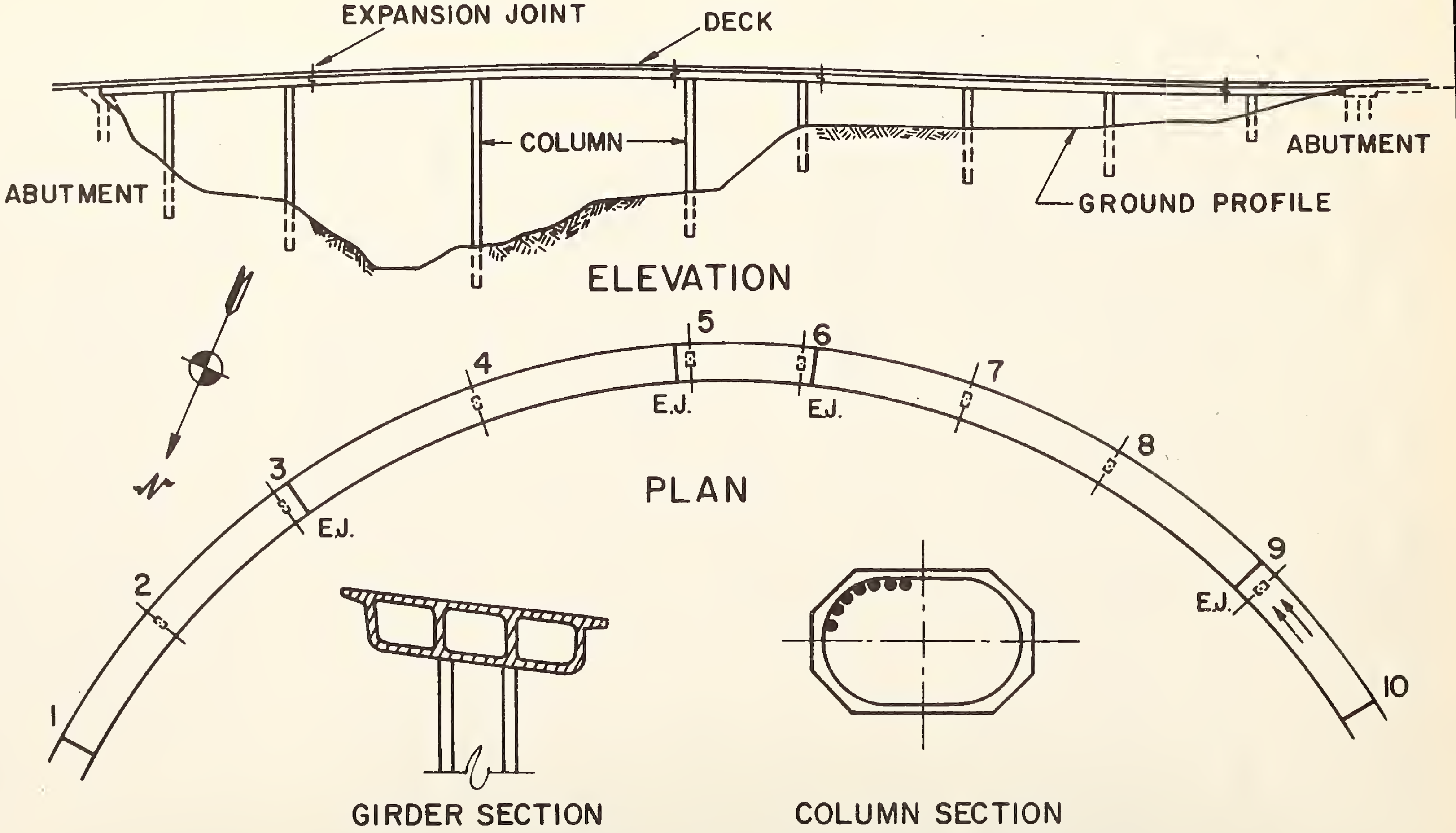
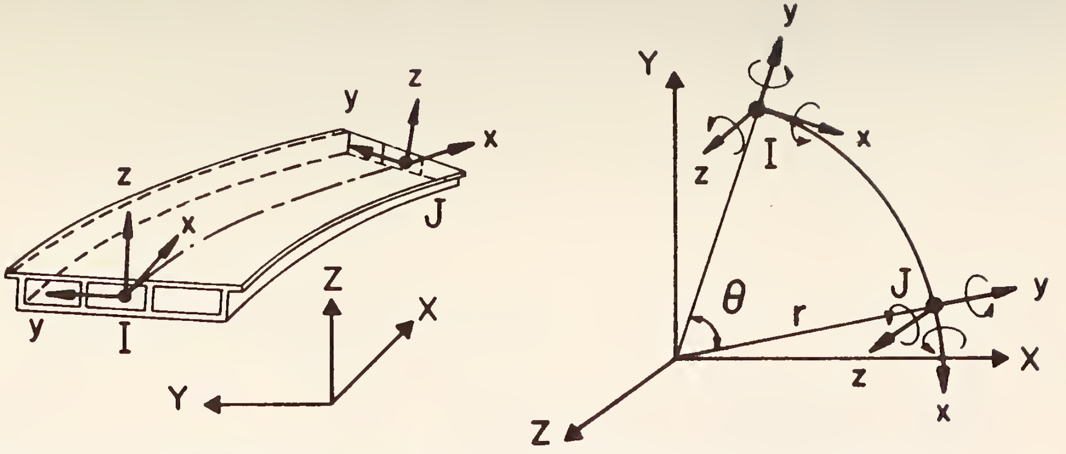
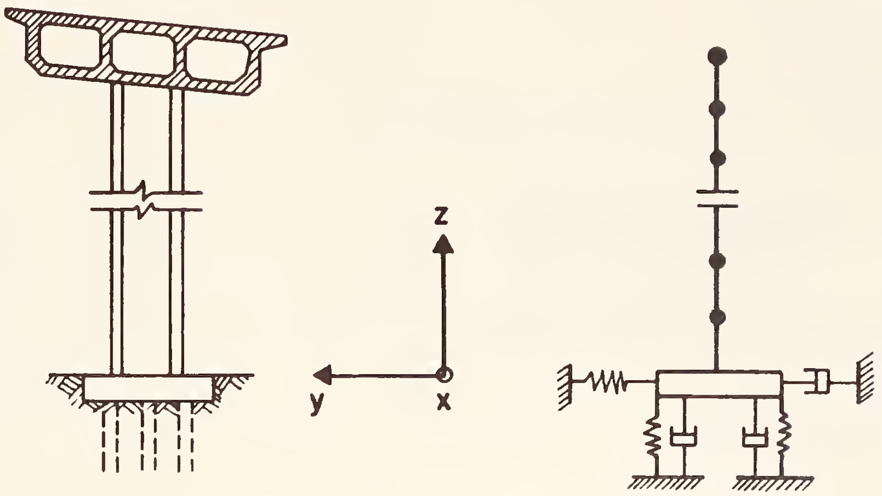


Fig. 1 5/14 South Connector Overcrossing

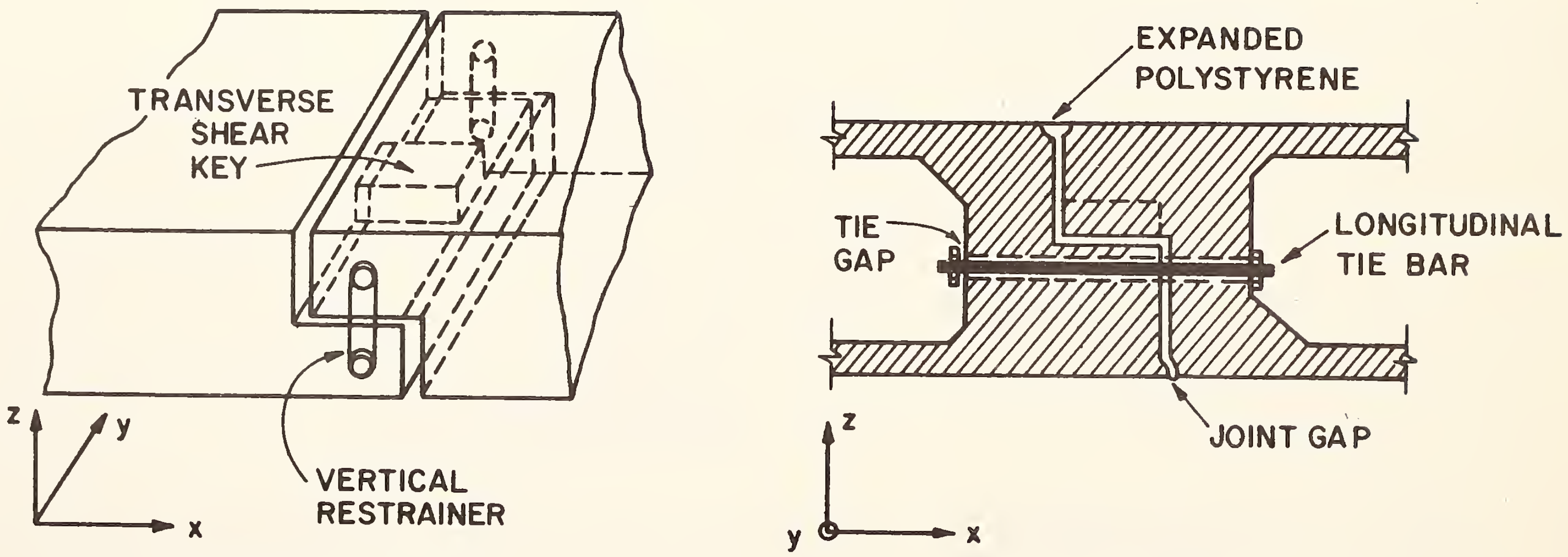


(a) TYPICAL BOX GIRDER ELEMENT (b) IDEALIZED MODEL



(c) COLUMN AND FOUNDATION IDEALIZATION

Fig. 2 Mathematical Modeling of Bridge Structural System



(a) EXPANSION JOINT

(b) SECTION THROUGH JOINT

Fig. 3 Bridge Expansion Joint

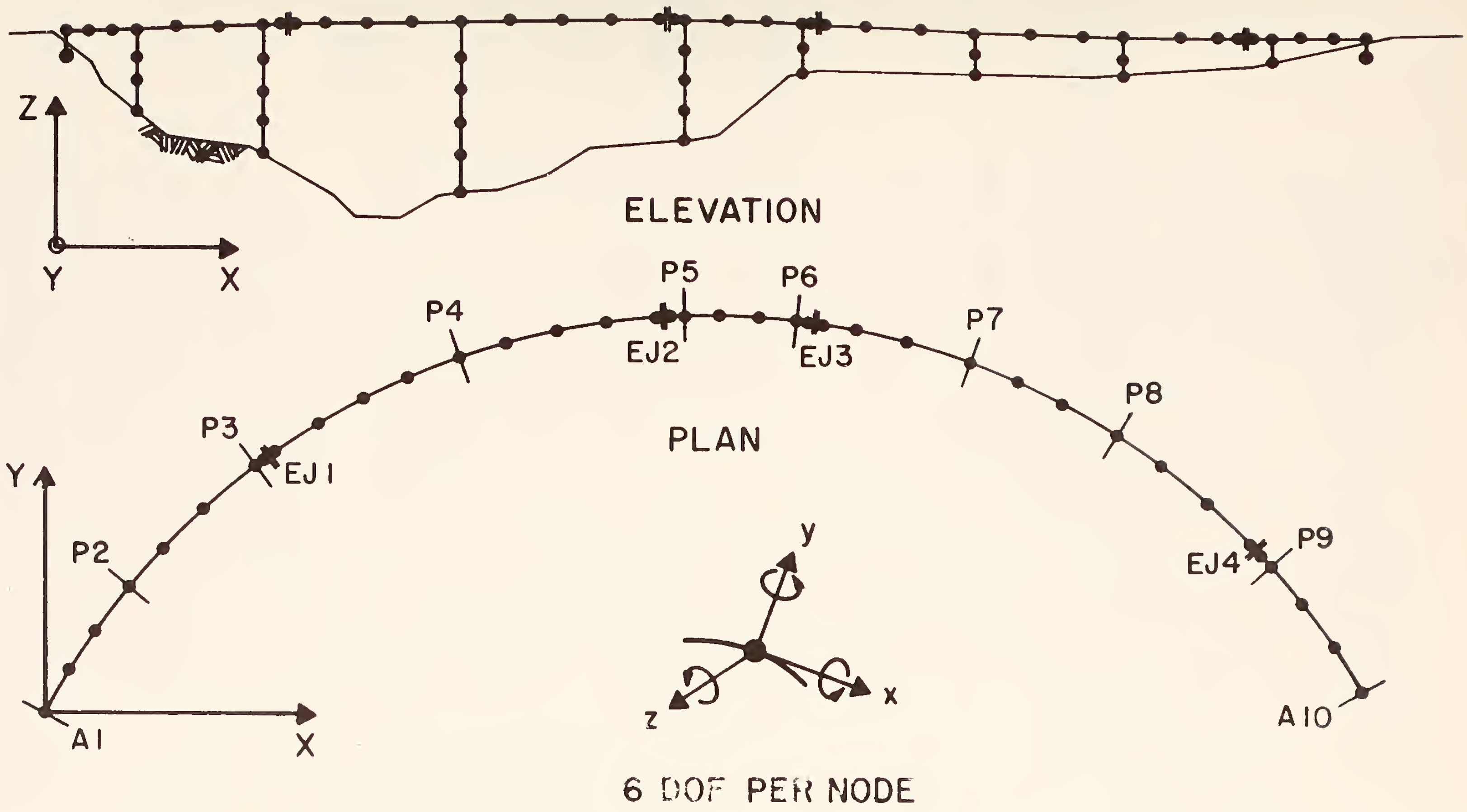


Fig. 4 Idealized Lumped Parameter System For 5/14 South Connector Overcrossing

III ELASTO-PLASTIC FLEXURAL COLUMN ELEMENT FOR REINFORCED CONCRETE COLUMNS

A. BASIC ASSUMPTIONS

As pointed out previously, limited inelastic deformations may be permitted in reinforced concrete bridge columns under severe earthquake conditions. These deformations should however be permitted only when adequate ductility and energy absorption capacities are available. Therefore, cyclic yielding should be allowed in flexure only where high ductility can be provided. Since shear inelastic deformations are usually brittle in character, they should be avoided. Thus, it is essential in providing high seismic resistance under severe overload conditions that reinforced concrete columns be sized and detailed so that yielding will be limited to flexure.

Because reinforced concrete is a highly heterogeneous composite material, the overall inelastic behavior of column elements should be defined in terms of generalized forces and their corresponding generalized deformations. Six generalized forces exist at any section of an element, namely, an axial force P, two components of transverse shear Q_y and Q_z, torsion T, and bending moments M_y and M_z, Fig. 5. Axial normal stresses are caused by generalized forces P, M_y, and M_z while transverse shear stresses are caused by Q_y, Q_z, and T. In constructing an inelastic element for reinforced concrete columns, the following basic assumptions have been made (see Fig. 5):

(1) The generalized force-deformation relations of the element follows that of an elasto-perfectly plastic (elasto-plastic) material model having yield strengths corresponding to the ultimate capacities of the member.

(2) The ultimate shear strengths, Q_{yu} and Q_{zu}, and the ultimate torsional strength, T_u, are infinitely large.

(3) The ultimate axial strength P_u and the ultimate bending strengths M_{yu} and M_{zu} are determined from the axial stress distribution present on the cross-section under ultimate conditions and are independent of the shear stresses caused by Q_y, Q_z, and T.

(4) The interaction among P_u, M_{yu}, and M_{zu} can be represented by a three-dimensional generalized yield surface in the force space spanned by P, M_y, and M_z.

(5) The element retains linear elastic behavior between nodal points; however, elasto-plastic behavior as defined in assumptions (1) through (4) above is permitted to occur over the cross-section at each end for a distance Δx approaching zero, i.e., the inelastic behavior is fully concentrated at the ends of the element.

Assumptions (1) through (4) relate to material properties within the concentrated elasto-plastic hinges at the ends of the elements. Between these hinges the material remains elastic. It should be understood that the above assumptions have been made to approximate the actual case in which inelastic deformations may occur at any section along a column. If a sufficient number of elements are used, a good approximation results. This is due to the fact that the real case is approached in the limit with increasing numbers of elements.

The inelastic column element resulting from the above assumptions may be constructed mathematically by carrying out the following three steps:

(1) Determine the ultimate strengths P_u, M_{yu}, and M_{zu} for the given section under ultimate conditions.

(2) Define the generalized yield function which controls the yielding interaction of P_u, M_{yu}, and M_{zu}.

(3) Formulate the elasto-plastic tangent stiffnesses in terms of the element force and corresponding nodal displacement increments during yielding.

The analytical details used in carrying out these steps, as well as the computational techniques used in the step-by-step dynamic analysis procedure, are presented in the subsequent sections of this chapter.

B. ULTIMATE STRENGTHS

The ultimate strengths of a reinforced concrete column subjected to combined axial load and biaxial bending can be determined using the following idealizations [20,21]:

(1) Plane sections remain plane.

(2) Concrete carries no tension.

(3) Maximum compressive strain of concrete equals 0.003.

(4) The concrete stress distribution in the compression zone can be replaced by an equivalent stress block having uniform intensity equal to 0.85 times the concrete cylinder strength f'_c and distributed over an equivalent compression zone bounded by the boundary of the cross-section and a straight line located at a distance $(1-k_1)c$ from and parallel to the neutral axis where c is the distance from the location of maximum compressive strain to the neutral axis and where

$$k_1 = \begin{cases} 0.85 & , \quad f'_c \leq 4000 \text{ psi} \\ 0.85-0.05\left(\dfrac{f'_c-4000}{1000}\right) & , \quad f'_c > 4000 \text{ psi} \end{cases} \tag{9}$$

(5) The stress-strain relation for the steel reinforcement follows an elasto-perfectly plastic law having equal yield stresses in tension and compression.

The idealized stress-strain distribution at ultimate condition is shown in Fig. 6 for a typical bridge column section. In this figure, the y axis is selected as the minor axis of the section and the z axis is selected as the major axis. Using conditions of equilibrium, the ultimate axial force P_u and the ultimate bending moments M_{yu} and M_{zu} can be expressed in the form

$$
\begin{aligned}
P_u &= -0.85\, f'_c A_c - \Sigma_c A_s f_{sc} + \Sigma_t A_s f_{st} \\
M_{yu} &= -0.85\, f'_c A_c z_c - \Sigma_c A_s f_{sc} z_{sc} + \Sigma_t A_s f_{st} z_{st} \qquad (10) \\
M_{zu} &= 0.85\, f'_c A_c y_c + \Sigma_c A_s f_{sc} y_{sc} - \Sigma_t A_s f_{st} y_{st}
\end{aligned}
$$

where the signs of P_u, M_{yu}, and M_{zu} agree with the convention shown in Fig. 6, the symbol Σ_c sums over the steel reinforcements in the compression zone, and Σ_t sums over the steel reinforcements in the tension zone.

Since the ultimate strengths P_u, M_{yu}, and M_{zu} of a given column section are nonlinear functions of the geometric parameters of the section, an iterative procedure must be used to compute their values. An initial position of the neutral axis is first assumed in this procedure; then, following an equilibrium correction using Eqs. (10), a new neutral axis position is determined. This procedure is repeated until convergence to the correct neutral axis position is sufficiently obtained.

Because modern bridge columns usually have rather complex shapes of cross-sections and the steel reinforcements are quite dense and peripherally arranged, the above described iterative procedure can be very tedious if carried out by traditional methods. For this reason, an efficient procedure using Newton's method was developed which can treat cross-sections of arbitrary shapes having arbitrary distributions of steel reinforcements. A computer program YIELD has been developed for this purpose

Having calculated the numerical values of P_u, M_{yu}, and M_{zu} for a given cross-section and distribution of steel reinforcements using Eqs. (10), they represent a point in a three-dimensional generalized force space spanned by P, M_y, and M_z. If all such possible points were computed, they would define a three-dimensional surface which controls the interactions among ultimate strengths P_u, M_{yu}, and M_{zu}. It is convenient to represent this surface by two parameters, namely, an axial force ratio p_u and an eccentricity angle θ as defined by

$$p_u = \frac{P_u}{P_o} \qquad\qquad \theta = \tan^{-1} \left(\frac{M_{yu}}{M_{zu}} \right) \tag{11}$$

where P_o is the ultimate axial compressive strength under conditions of no bending. The interaction surface can then be represented in terms of

$$\begin{aligned} P_u &= P_u\,(p_u,\theta) \quad ; \quad P_o = P_u\,(-1,0) \\ M_{yu} &= M_{yu}\,(p_u,\theta) \quad ; \quad M_{yo} = M_{yu}\,(0,0) \\ M_{zu} &= M_{zu}\,(p_u,\theta) \quad ; \quad M_{zo} = M_{zu}\,(0\,,\,\tfrac{\pi}{2}) \end{aligned} \tag{12}$$

For a fixed value of p_u, Eqs. (12) define the biaxial bending interaction curve between moments M_{yu} and M_{zu} and, for a fixed value of θ, these equations specify the interaction curve between the ultimate axial force P_u and the ultimate bending moment $M_u = (M_{yu}^{\,2} + M_{zu}^{\,2})^{\frac{1}{2}}$ about an axis located at an angle θ from the y axis. As an example, consider the typical bridge column section shown in Fig. 7. Ultimate strengths have been calculated for this section and interaction curves have been established. Figure 8 presents a single biaxial bending interaction curve which as shown is representative for widely differing values of p_u. The ultimate moments M_{yu} and M_{zu} in this case, as in others, have been normalized with respect to M_{yo} and M_{zo}, respectively. Figure 9 shows a single interaction curve for axial force P_u and the normalized M_{yu} moment which is representative for θ over the full range $0 < \theta < 90°$.

The shape of the yield surface as represented by Eq. (10) is obviously dependent upon the geometric shape of the cross-section and upon the amount and distribution of reinforcing steel. It is very tedious however to generate the complete yield surface for each type of section under consideration. Therefore, for analytical purposes, approximate yield functions have been developed which depend on fewer parameters.

C. GENERALIZED YIELD FUNCTION

Many approximate yield surfaces have been proposed for reinforced concrete sections under combined loadings [22,23]. Most forms developed however are for rectangular sections having simple arrangements of reinforcing bars. Therefore, they cannot be readily applied to the more complex cases arising in modern bridge design.

The approximate yield surface developed herein follows a form similar to that proposed by Bresler [22] for biaxial bending and uses a cubic polynomial approximation for the axial force and bending interaction curves. As proposed by Bresler, the biaxial bending interaction curves for fixed values of p_u can be approximated by the relation

$$\left| \frac{M_{yu}}{M_{yp}} \right|^a + \left| \frac{M_{zu}}{M_{zp}} \right|^b = 1 \qquad (13)$$

where a and b are constants which depend upon the cross-section geometries and where M_{yp} and M_{zp} are the ultimate bending moments about the y and z axes, respectively, for a fixed value of p_u and for M_{zu} and M_{yu} set equal to zero, respectively. Moments M_{yp} and M_{zp} can be approximated by the following cubic equations:

$$\left| \frac{M_{yp}}{M_{yo}} \right| = 1.0 + a_1 \left(\frac{P_u}{P_o} \right) + a_2 \left(\frac{P_u}{P_o} \right)^2 + a_3 \left(\frac{P_u}{P_o} \right)^3 , \quad - P_o < P_u < P_t$$

$$\left| \frac{M_{zp}}{M_{zo}} \right| = 1.0 + b_1 \left(\frac{P_u}{P_o} \right) + b_2 \left(\frac{P_u}{P_o} \right)^2 + b_3 \left(\frac{P_u}{P_o} \right)^3 , \quad - P_o < P_u < P_t \tag{14}$$

where P_t is the ultimate axial tension for the column under conditions of no bending and a_1, a_2, a_3, b_1, b_2, and b_3 are constants. Combining Eqs. (13) and (14), the approximate yield surface can be represented in the normalized form

$$\left| \frac{m_{yu}}{1 + a_1 p_u + a_2 p_u^2 + a_3 p_u^3} \right|^a + \left| \frac{m_{zu}}{1 + b_1 p_u + b_2 p_u^2 + b_3 p_u^3} \right|^b = 1 \tag{15}$$

or symbolically

$$f(p_u, m_{yu}, m_{zu}) = 1 \tag{16}$$

where

$$p_u = P_u/P_o, \quad m_{yu} = M_{yu}/M_{yo}, \quad m_{zu} = M_{zu}/M_{zo}$$

are the normalized resultant force components.

The normalized yield function, Eq. (16), contains eleven constants, namely, P_o, M_{yo}, M_{zo}, a, a_1, a_2, a_3, b, b_1, b_2, and b_3. These constants are determined by fitting smooth curves through ultimate strength values obtained by the methods previously described. For typical bridge columns which have nearly oval-shaped cross-sections and are peripherally reinforced, an ellipse can be used to approximate the biaxial bending interaction curve. In this case, a value of 2 is used for both a and b. If, on the other hand, the cross-section is a long, narrow, rectangular section, the corresponding interaction curve approaches a straight line in which case a and b both equal unity. Figure 10 shows a three-dimensional plot of the yield surface given by Eq. (15) which is smooth everywhere except at $P = P_o$ and $P = P_t$ when a and b are both equal to 2.

In accordance with the definition of a yield function, if the values of P, M_y, and M_z are such that $f(p, m_y, m_z) < 1$, where $p = P/P_o$, $m_y = M_y/M_{yo}$, and $m_z = M_z/M_{zo}$, then the cross-section is in an elastic state. If $p = P_u$, $m_y = m_{yu}$, and $m_z = m_{zu}$, the section is in a yield (or impending yield) state. When $df = o$, the values of p_u, m_{yu}, and m_{zu} can change but they must always be compatible with the yield surface. If $df < o$, the section is being unloaded. A set of values for p, m_y, and m_z which gives $f(p, m_y, m_z) > 1$ is, of course, not possible using an elasto-perfectly plastic model.

D. TANGENT STIFFNESSES FOR ELASTO-PLASTIC FLEXURAL COLUMN ELEMENTS

In this section, the elasto-plastic tangent stiffness matrix $\underline{k}_t^{EP}$ which relates the element force increments $d\underline{S}$ to the element nodal displacement increments $d\underline{r}$ for an elasto-plastic flexural column element yielding at one or both ends will be derived using the generalized yield function $f(p_u, m_{yu}, m_{zu})$ established previously and the associated flow rule for an elasto-plastic solid. The desired relation is

$$d\underline{S} = \underline{k}_t^{EP}\, d\underline{r} \tag{17}$$

where

$$d\underline{S} = \begin{Bmatrix} d\underline{S}_I \\ \\ d\underline{S}_J \end{Bmatrix} \qquad d\underline{r} = \begin{Bmatrix} d\underline{r}_I \\ \\ d\underline{r}_J \end{Bmatrix}. \tag{18}$$

where $d\underline{S}_K$, $d\underline{r}_K$, K = I, J are the vectors of element force increments and corresponding element nodal displacement increments at node K.

By virtue of assumption (5) in Section A and the assumption that the element deformation increments can be additively decomposed into an elastic component and a plastic component, the element nodal displacement increments $d\underline{r}$ can be expressed as

$$d\underline{r} = d\underline{r}^E + d\underline{r}^P \tag{19}$$

where

$$d\underline{r}^{E} = \begin{Bmatrix} d\underline{r}_I^{E} \\ \\ d\underline{r}_J^{E} \end{Bmatrix} \qquad d\underline{r}^{P} = \begin{Bmatrix} d\underline{r}_I^{P} \\ \\ d\underline{r}_J^{P} \end{Bmatrix} \tag{20}$$

and where $d\underline{r}_K^E$, K = I, J, represents the vector of elastic nodal displacement increments at node K, and $d\underline{r}_K^P$, K = I, J represents the plastic displacement increments resulting from concentrated plastic deformations at end K. The element forces are uniquely determined by the element elastic nodal displacements, i.e.,

$$\underline{S} = \underline{k}^{E} \ \underline{r}^{E} \tag{21}$$

and therefore

$$d\underline{S} = \underline{k}^{E} \ d\underline{r}^{E} \tag{22}$$

where $\underline{k}^E$ is the elastic stiffness matrix of the element.

According to the associated flow rule for an elasto-plastic solid [24], the plastic deformation increments $d\underline{r}_K^P$ at a particular section K, is governed by

$$d\underline{r}_K^{P} = \left[\frac{\partial f}{\partial \underline{S}_K}\right] d\lambda_K \tag{23}$$

where $d\lambda_K$ is a positive scalar proportionality constant which can be determined by the yield condition, i.e.,

$$f(\underline{S}_K) = 1 \quad , \quad df = \left[\frac{\partial f}{\partial \underline{S}_K}\right]^T d\underline{S}_K = 0 \tag{24}$$

where T indicates the transpose operation. Note that if the section is not yielding, Eq. (23) still applies provided $\partial f/\partial \underline{S}_K$ is set equal to $\underline{0}$. Letting K = I, J, the flow rule can be expressed as

$$d\underline{r}^{P} = \begin{Bmatrix} d\underline{r}_I^{P} \\ \\ d\underline{r}_J^{P} \end{Bmatrix} = \begin{bmatrix} \dfrac{\partial f}{\partial \underline{S}_I} & \underline{0} \\ \\ \underline{0} & \dfrac{\partial f}{\partial \underline{S}_J} \end{bmatrix} \begin{Bmatrix} d\lambda_I \\ \\ d\lambda_J \end{Bmatrix} \equiv \left[\frac{\partial f}{\partial \underline{S}}\right] d\underline{\lambda} \tag{25}$$

and the yield condition becomes

$$\underset{\sim}{f} = \begin{Bmatrix} f(\underset{\sim}{S}_I) \\ \\ f(\underset{\sim}{S}_J) \end{Bmatrix} = \begin{Bmatrix} 1 \\ \\ 1 \end{Bmatrix}, \quad d\underset{\sim}{f} = \begin{bmatrix} \frac{\partial f}{\partial \underset{\sim}{S}_I} & \underset{\sim}{o} \\ \underset{\sim}{o} & \frac{\partial f}{\partial \underset{\sim}{S}_J} \end{bmatrix}^T \begin{Bmatrix} d\underset{\sim}{S}_I \\ \\ d\underset{\sim}{S}_J \end{Bmatrix}$$

$$= \left[\frac{\partial f}{\partial \underset{\sim}{S}} \right]^T d\underset{\sim}{S} = \underset{\sim}{o} \tag{26}$$

Premultiplying both sides of the equation

$$d\underset{\sim}{S} = \underset{\sim}{k}^E \, d\underset{\sim}{r}^E = \underset{\sim}{k}^E \, (d\underset{\sim}{r} - d\underset{\sim}{r}^P) \tag{27}$$

by $[\partial f/\partial \underset{\sim}{S}]^T$, one obtains upon using Eq. (26) the relation

$$\left[\frac{\partial f}{\partial \underset{\sim}{S}} \right]^T d\underset{\sim}{S} = \left[\frac{\partial f}{\partial \underset{\sim}{S}} \right]^T \underset{\sim}{k}^E \, (d\underset{\sim}{r} - d\underset{\sim}{r}^P) = \underset{\sim}{o} \tag{28}$$

Substituting Eq. (25) into Eq. (28), one obtains

$$\left[\frac{\partial f}{\partial \underset{\sim}{S}} \right]^T \underset{\sim}{k}^E \left[\frac{\partial f}{\partial \underset{\sim}{S}} \right] d\underset{\sim}{\lambda} = \left[\frac{\partial f}{\partial \underset{\sim}{S}} \right]^T \underset{\sim}{k}^E \, d\underset{\sim}{r} \tag{29}$$

Solving for $d\underset{\sim}{\lambda}$ gives

$$d\underset{\sim}{\lambda} = \left\{ \left[\frac{\partial f}{\partial \underset{\sim}{S}} \right]^T \underset{\sim}{k}^E \left[\frac{\partial f}{\partial \underset{\sim}{S}} \right] \right\}^{-1} \left[\frac{\partial f}{\partial \underset{\sim}{S}} \right]^T \underset{\sim}{k}^E \, d\underset{\sim}{r} \tag{30}$$

which after substituting into Eq. (25) gives

$$d\underset{\sim}{r}^P = \left[\frac{\partial f}{\partial \underset{\sim}{S}} \right] \left\{ \left[\frac{\partial f}{\partial \underset{\sim}{S}} \right]^T \underset{\sim}{k}^E \left[\frac{\partial f}{\partial \underset{\sim}{S}} \right] \right\}^{-1} \left[\frac{\partial f}{\partial \underset{\sim}{S}} \right]^T \underset{\sim}{k}^E \, d\underset{\sim}{r} \tag{31}$$

Substituting Eq. (31) into Eq. (27) the required relation

$$d\underset{\sim}{S} = (\underset{\sim}{k}^E - \underset{\sim}{k}_t^Y) \, d\underset{\sim}{r} \equiv \underset{\sim}{k}_t^{EP} \, d\underset{\sim}{r} \tag{32}$$

is obtained where

$$\underline{k}_t^Y \equiv \underline{k}^E \left[\frac{\partial f}{\partial \underline{S}}\right] \left\{ \left[\frac{\partial f}{\partial \underline{S}}\right]^T \underline{k}^E \left[\frac{\partial f}{\partial \underline{S}}\right] \right\}^{-1} \left[\frac{\partial f}{\partial \underline{S}}\right]^T \underline{k}^E \tag{33}$$

Physically speaking, the matrix $\underline{k}_t^Y$ represents the losses of elastic stiffness in the element due to yielding, and the elasto-plastic tangent stiffness matrix $\underline{k}_t^{EP}$ represents the reduced elastic stiffnesses of the element. It should be mentioned that matrix $\underline{k}_t^{EP}$ may be singular depending upon the location of the point of interest on the yield surface.

E. COMPUTATIONAL TECHNIQUES

The evaluation of the elasto-plastic tangent stiffness matrix appearing in Eq. (33) is straight forward theoretically once the yield point representing the element force state on yield surface is determined. However, in practice a step-by-step analysis procedure is usually required in which case the yield point determination is only approximate. To be more specific, a point inside the yield surface can move outside the surface during a single time interval provided the element is in a yielding state.

To treat this problem correctly, it is necessary to use an iteration procedure within each time interval. This procedure however requires a tremendous amount of computational effort. Therefore, in the present analysis, an approximate procedure which avoids iteration is adopted. This procedure is similar to one used by Chi and Powell [25].

Referring to Fig. 11, assume that the column element at the beginning of a time interval Δt and at time t is in the elastic state as represented by point A inside the yield surface. Following the established procedures a vector of element nodal displacement increments $\Delta\underline{r}_t$ is determined and an apparent new element force vector $\bar{\underline{S}}_{t+\Delta t}$ is then found using the relations

$$\bar{\underline{S}}_{t+\Delta t} = \underline{S}_t + \Delta\underline{S}_t^E = \underline{S}_t + \underline{k}^E \Delta\underline{r}_t \tag{34}$$

and

$$f(\bar{\underline{S}}_{t+\Delta t}) > 1 \tag{35}$$

The method then used to determine yield point C proceeds by determining a scalar factor μ_1, $0 \leq \mu_1 \leq 1$ such that $f(\underline{S}_t + \mu_1 \Delta\underline{S}_t^E) = 1$. One can then evaluate the elasto-plastic tangent stiffness matrix for point C. A new element force vector $\bar{\bar{\underline{S}}}_{t+\Delta t}$, as represented by point D, is then computed using the relation

$$\bar{\bar{\underline{S}}}_{t+\Delta t} = \underline{S}_t + \mu_1 \, \Delta\underline{S}_t^E + (1-\mu_1) \, \underline{k}_t^{EP} \, \Delta\underline{r}_t \tag{36}$$

Usually, point D is also located outside the yield surface due to the local convex shape of the yield surface. A second scalar factor μ_2 is now introduced to bring point D to point E on the yield surface along the vector $\bar{\bar{\underline{S}}}_{t+\Delta t}$. The final element force vector is determined by

$$\underline{S}_{t+\Delta t} = \mu_2 \, \bar{\bar{\underline{S}}}_{t+\Delta t} \quad , \quad 0 \leq \mu_2 \leq 1 \tag{37}$$

where μ_2 is determined so that $f(\mu_2 \bar{\bar{\underline{S}}}_{t+\Delta t}) = 1$.

Finally, at the end of the time interval, a new elasto-plastic tangent stiffness matrix is evaluated, using the new yielding force vector $\underline{S}_{t+\Delta t}$, i.e., the vector to point E. This new tangent stiffness matrix is then used as the element stiffness matrix during the next time interval.

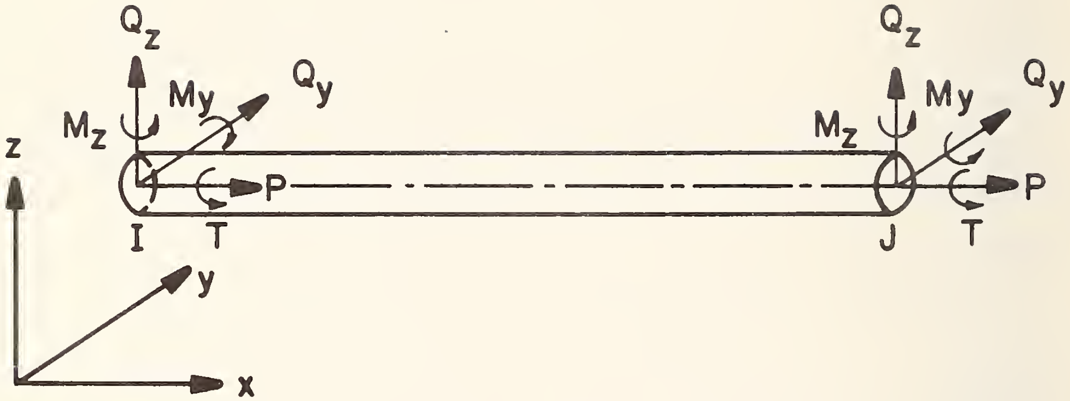


Fig. 5 Local Coordinate System For a Bridge Column

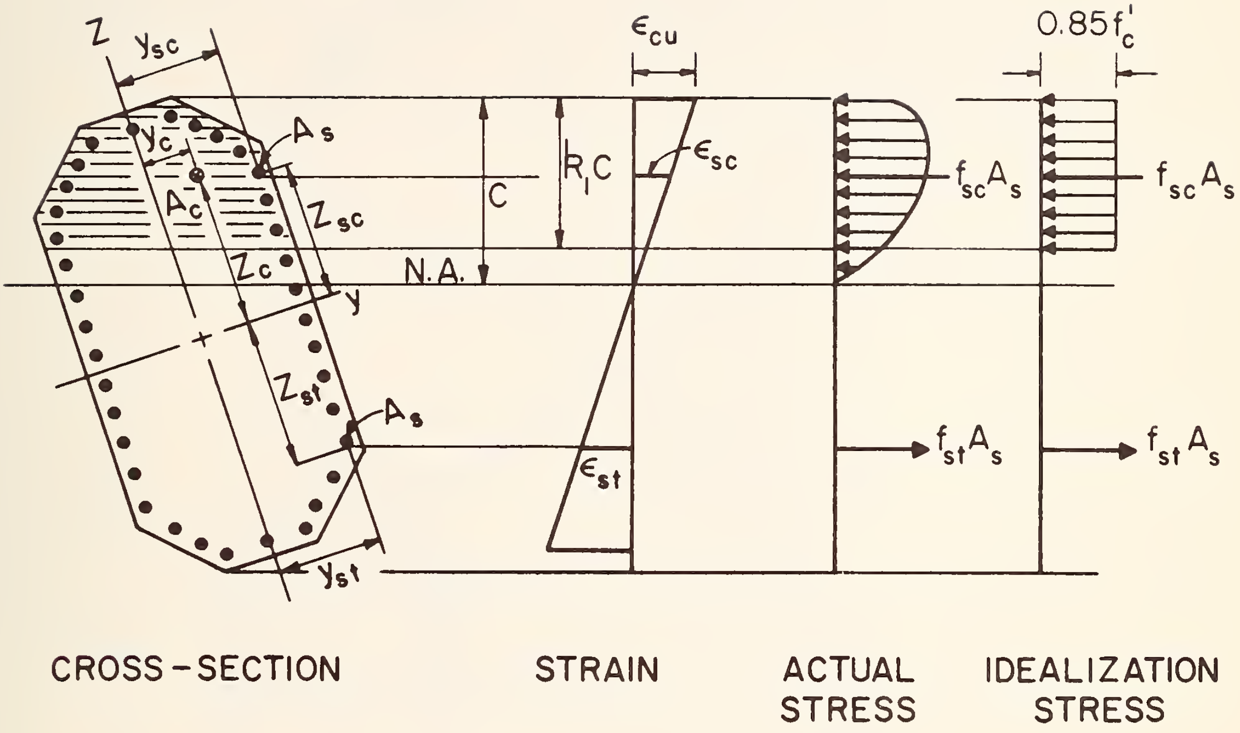
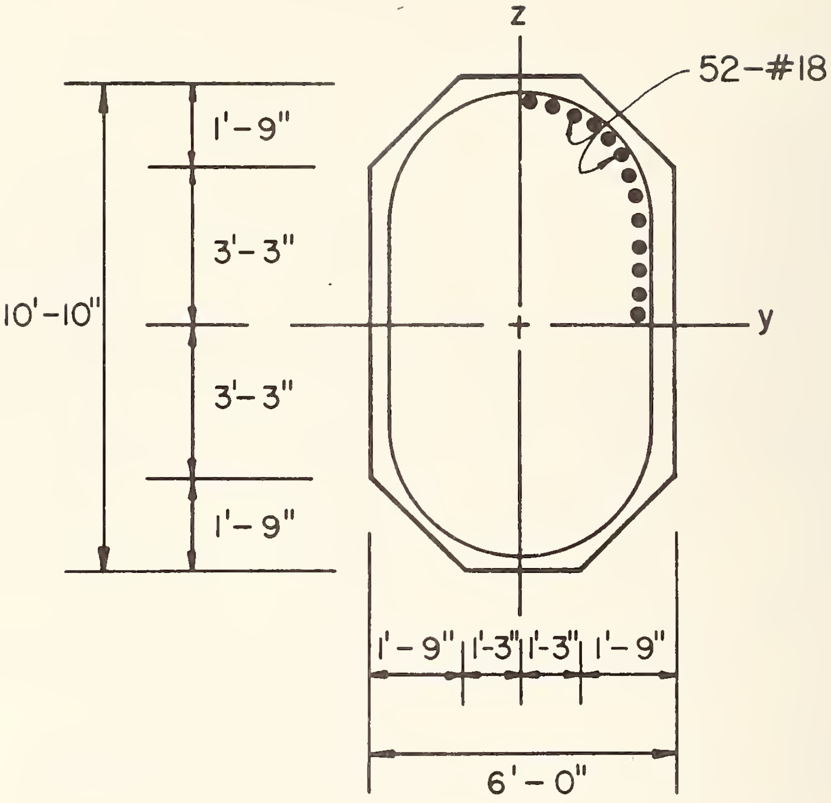


Fig. 6 Idealized Ultimate Stress-Strain Distribution For a Typical Bridge Column Section



$f'_c = 4{,}000$ PSI

$f'_{sy} = 40{,}000$ PSI

$E_s = 2.9 \times 10^7$ PSI

$P_o = 3.47 \times 10^7$ LB.

$M_{yo} = 3.99 \times 10^8$ LB.-IN

$M_{zo} = 2.58 \times 10^8$ LB.-IN

Fig. 7 Cross-Section of a Typical Bridge Column

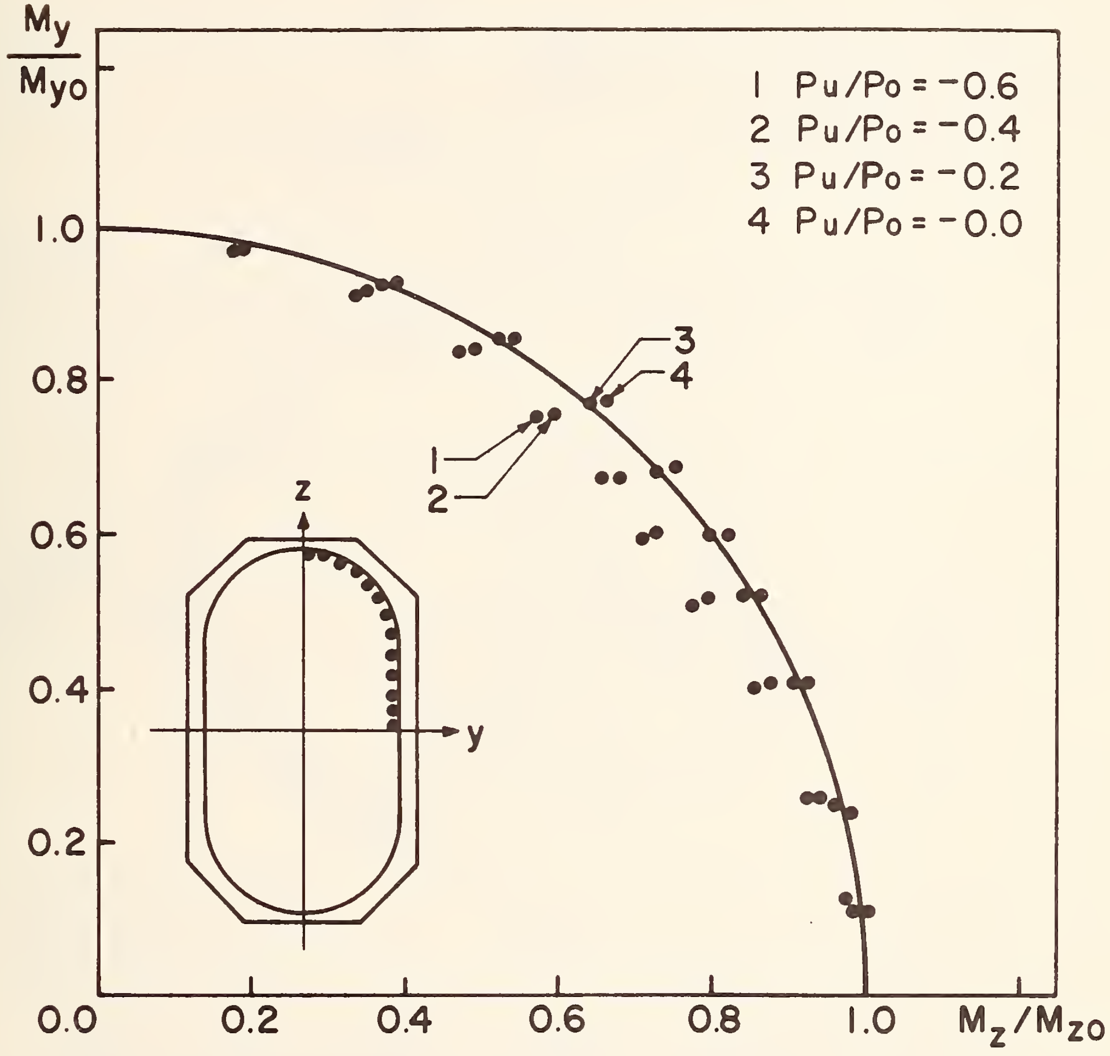


Fig. 8 Biaxial Bending Interaction Curve

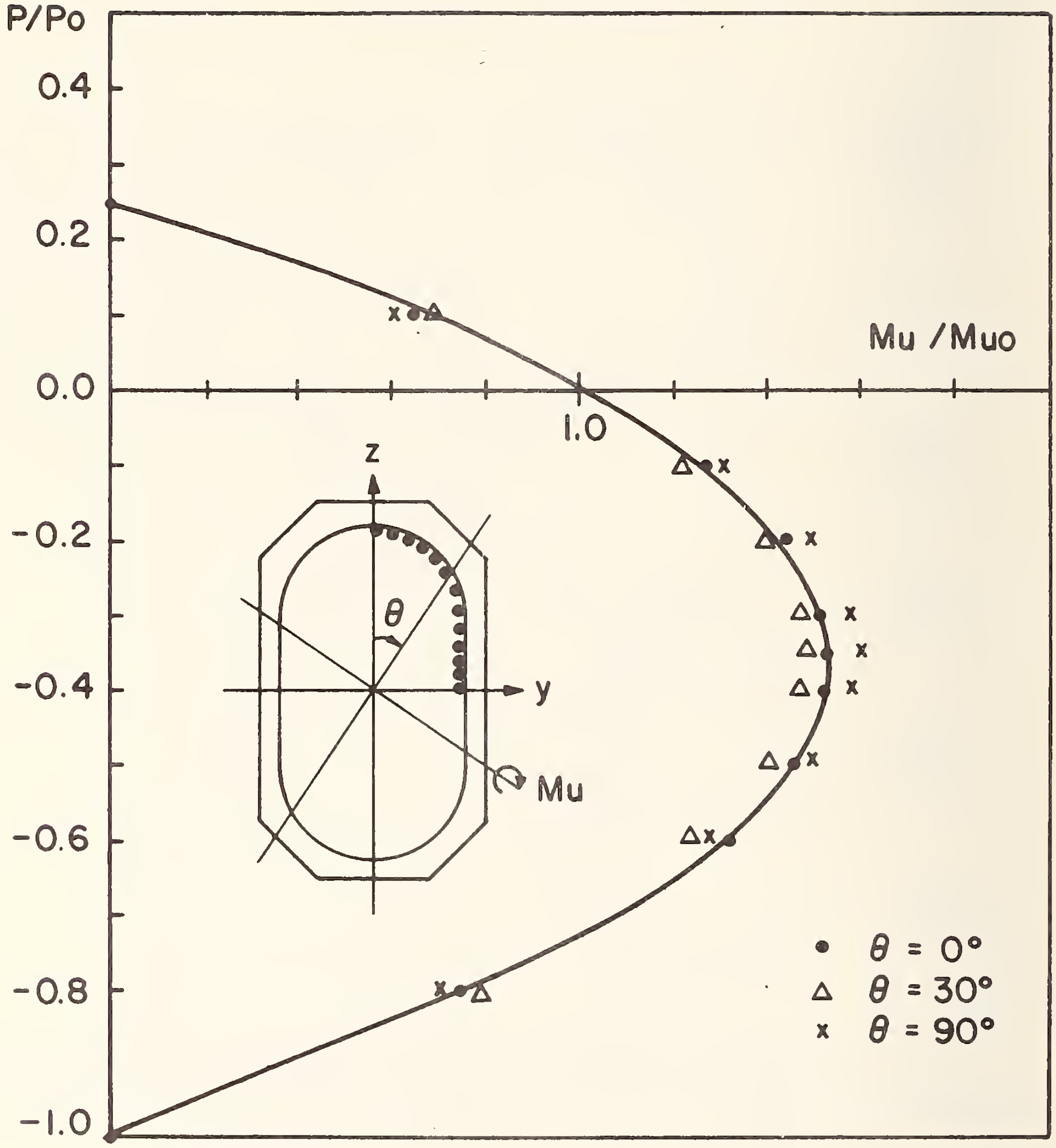


Fig. 9 Axial Force and Bending Moment Interaction Curve

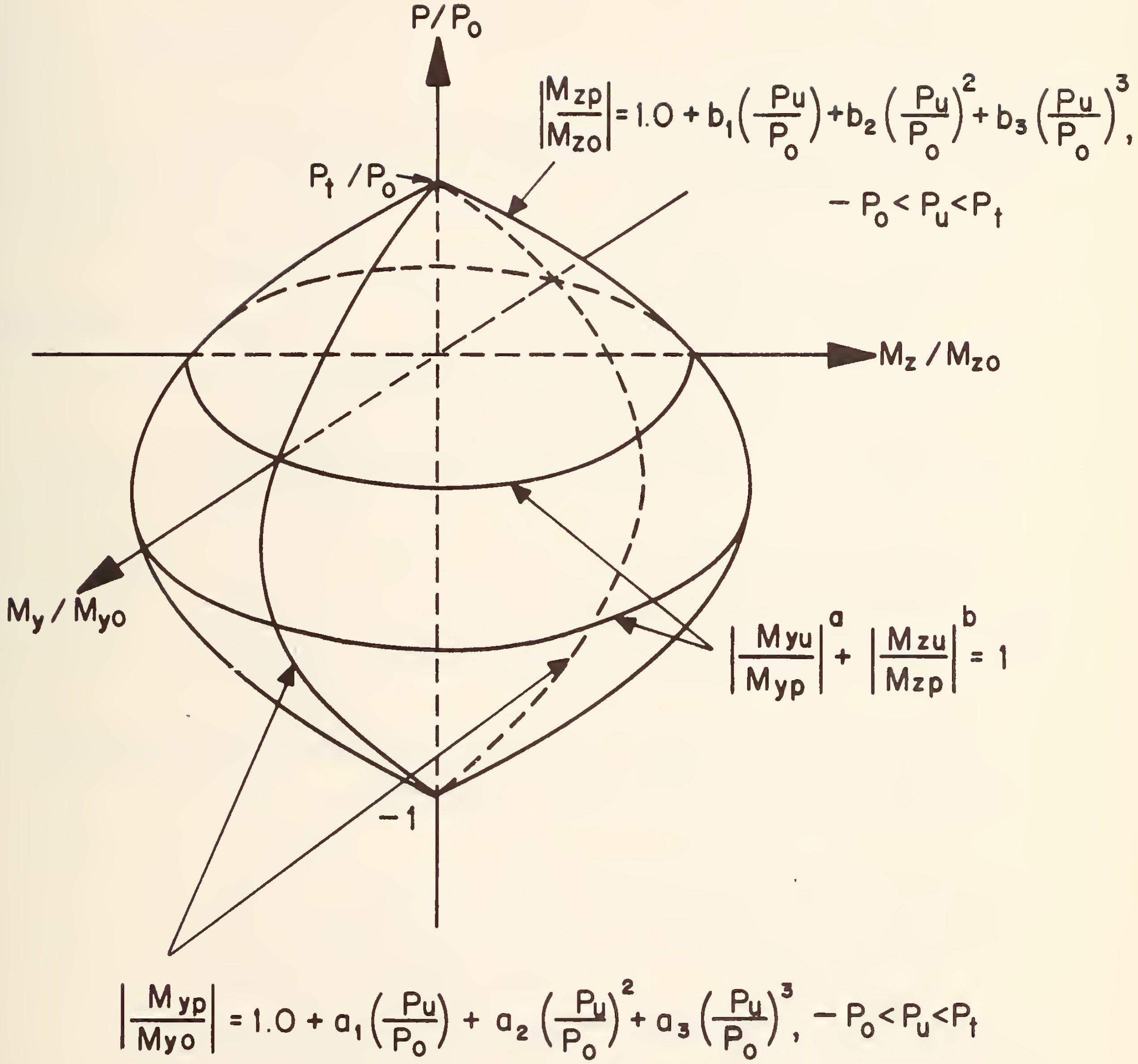


Fig. 10 Yield Surface of Typical Bridge Column

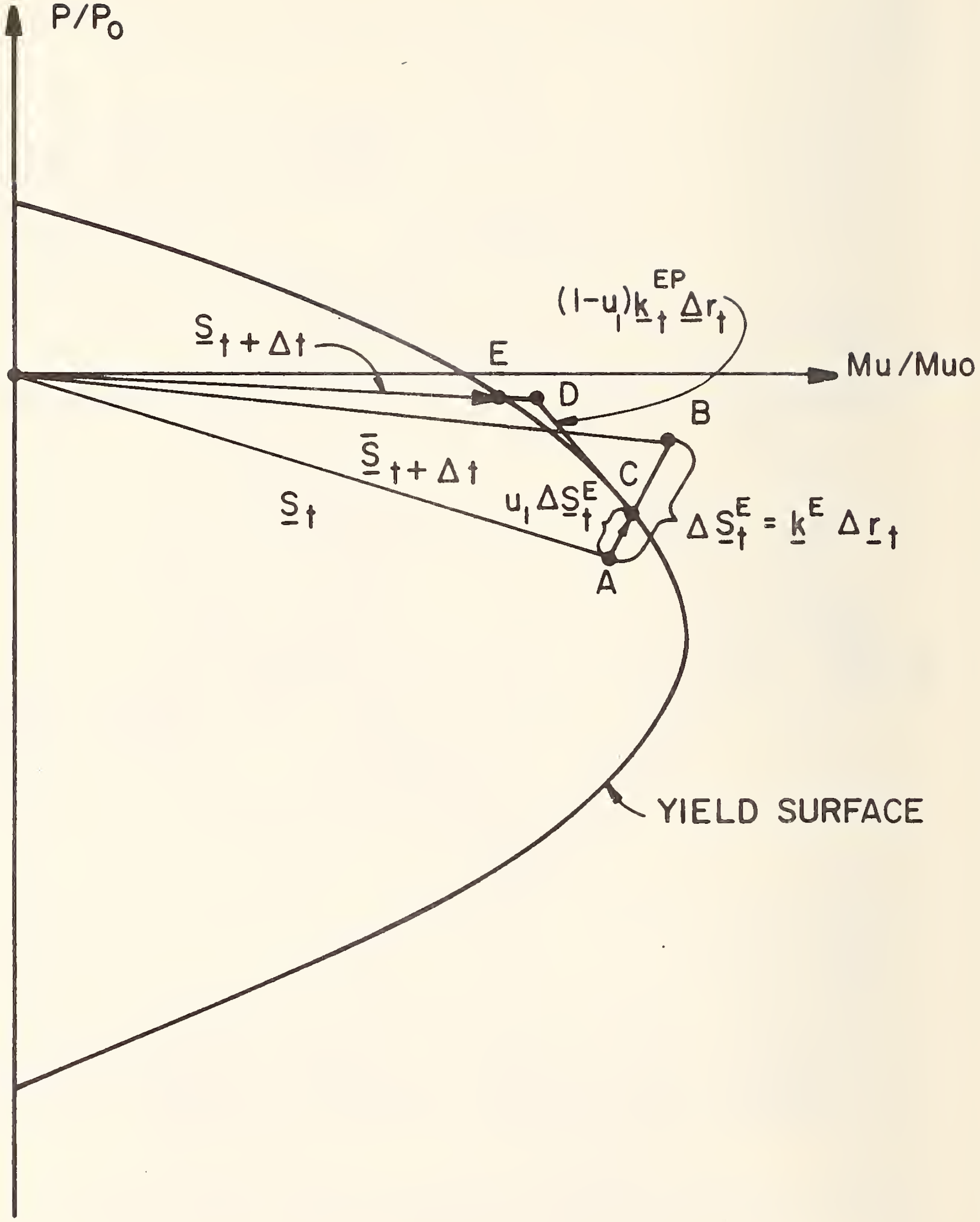


Fig. 11 Determination of Yield Points

IV NONLINEAR MODEL FOR EXPANSION JOINTS

A. IDEALIZATION

A nonlinear mathematical model for simulating the dynamic behavior of expansion joints has been defined. This model is idealized as follows (see Fig. 12):

(1) The end diaphragms of the deck at each expansion joint are rigid.

(2) Contact within the expansion joint can develop at only two points, A and B, located at the ends of the rigid diaphragms and separated by a distance d in the transverse direction (y axis).

(3) At contact points A and B, longitudinal (x axis) impact springs having a large stiffness* k_I are attached to one rigid diaphragm leaving a small gap Δ_G between the other diaphragm and the springs.

(4) Longitudinal impact starts at points A and B when the relative displacements of the two rigid diaphragms close the gap Δ_G.

(5) Relative slippage between the two diaphragms can take place in the longitudinal direction at points A and B under the action of Coulomb friction.

(6) During periods of slippage, the Coulomb friction forces at points A and B are proportional to their respective vertical contact forces and act in directions opposite to their directions of slippage.

---

* The larger the stiffness is, the closer that true impact condition is attended. The practical limit of this value depends upon the number of significant digit carried by the computer used and the time interval chosen in the analysis.

(7) A total of N_T longitudinal restrainer ties having equal elastic tensile stiffnesses k_T and zero compressive stiffnesses are placed through the two rigid diaphragms with a tie gap Δ_T at one end of each bar and at transverse distance y_i $(i=1,2,---,N_T)$ from the longitudinal center line axis of the deck.

(8) Longitudinal restrainer ties are elasto-plastic in tension and have equal yield forces, S_T.

(9) Relative transverse motions of the two rigid diaphragms are restrained by a very stiff elastic shear spring of stiffness k_S located at the center point C.

(10) At contact points A and B, the rigid diaphragms are interconnected by vertical (z axis) springs having stiffnesses equal to k_V.

(11) To provide generality in the model, the expansion joint may be skewed with respect to the deck, i.e., the transverse axis of the point (s axis; Fig. 13) may not be normal to the x axis.

The behavior of the idealized expansion joint shown in Fig. 12 can be characterized conveniently using displacement coordinates, $\bar{\underline{r}}$, as shown in Fig. 13, i.e.

$$\bar{\underline{r}} = \begin{Bmatrix} \bar{\underline{r}}_I \\ \\ \bar{\underline{r}}_J \end{Bmatrix} \quad ; \quad \bar{\underline{r}}_K = \begin{bmatrix} r_{Ax} \\ r_y \\ r_{Az} \\ r_{Bx} \\ \theta_s \\ r_{Bz} \end{bmatrix}_K \qquad (K = I,J) \tag{38}$$

where θ_s is a joint rotation about the s axis which may be skewed by an angle ψ as shown. The expansion joint coordinates $\bar{\underline{r}}$ can be related to the local nodal coordinates $\underline{r}$ as defined by

$$\underline{r} = \begin{Bmatrix} \underline{r}_I \\ \\ \underline{r}_J \end{Bmatrix} \qquad \underline{r}_K = \begin{bmatrix} r_x \\ r_y \\ r_z \\ \theta_x \\ \theta_y \\ \theta_z \end{bmatrix}_K \qquad (K = I,J) \tag{39}$$

through the following transformation:

$$\underline{\bar{r}} = \begin{Bmatrix} \underline{\bar{r}}_I \\ \\ \underline{\bar{r}}_J \end{Bmatrix} = \begin{bmatrix} \underline{a} & \underline{0} \\ \\ \underline{0} & \underline{a} \end{bmatrix} \begin{Bmatrix} \underline{r}_I \\ \\ \underline{r}_J \end{Bmatrix} \equiv \underline{A}\ \underline{r} \tag{40}$$

where transformation matrix $\underline{a}$ is given by

$$\underline{a} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & \frac{d}{2} \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -\frac{d}{2} & \frac{d}{2}\tan\psi & 0 \\ 1 & 0 & 0 & 0 & 0 & -\frac{d}{2} \\ 0 & 0 & 0 & 0 & \frac{1}{\cos\psi} & 0 \\ 0 & 0 & 1 & \frac{d}{2} & -\frac{d}{2}\tan\psi & 0 \end{bmatrix} . \tag{41}$$

The forces associated with the expansion joint coordinates are

$$\underline{\bar{S}} = \begin{Bmatrix} \underline{\bar{S}}_I \\ \\ \underline{\bar{S}}_J \end{Bmatrix} \qquad \underline{\bar{S}}_K = \begin{bmatrix} S_{Ax} \\ S_y \\ S_{Az} \\ S_{Bx} \\ M_s \\ S_{Bz} \end{bmatrix}_K \quad , \ (K = I,J) \tag{42}$$

while the forces associated with the local nodal coordinates are:

$$\underline{S} = \left\{ \begin{matrix} \underline{S}_I \\ \\ \underline{S}_J \end{matrix} \right\} \qquad \underline{S}_K = \begin{bmatrix} S_x \\ S_y \\ S_z \\ M_x \\ M_y \\ M_z \end{bmatrix}_K \qquad (K = I,J) \tag{43}$$

It is easy to show that $\bar{\underline{S}}$ are related to $\underline{S}$ by the transformation

$$\underline{S} = \left\{ \begin{matrix} \underline{S}_I \\ \\ \underline{S}_J \end{matrix} \right\} = \begin{bmatrix} \underline{a}^T & \underline{0} \\ \\ \underline{0} & \underline{a}^T \end{bmatrix} \left\{ \begin{matrix} \bar{\underline{S}}_I \\ \\ \bar{\underline{S}}_J \end{matrix} \right\} \equiv \underline{A}^T \bar{\underline{S}} \tag{44}$$

B. PIECEWISE LINEAR EXPANSION JOINT STIFFNESSES

The idealized expansion joint can be characterized by a stiffness matrix which is piecewise linear with a number of discontinuities depending upon the relative displacements of the joint. For a step-by-step dynamic analysis, if it is assumed that these finite number of discontinuities occur at the end of each time step, then the vector of expansion joint force increments $\Delta\bar{\underline{S}}$ can be related to the vector of the expansion joint displacement increments $\Delta\bar{\underline{r}}$ by a stiffness matrix $\bar{\underline{k}}_t^{EJ}$ which is constant within each time step, i.e.,

$$\Delta\bar{\underline{S}} = \bar{\underline{k}}_t^{EJ} \, \Delta\bar{\underline{r}} \tag{45}$$

where $\bar{\underline{k}}_t^{EJ}$ is a function of $\bar{\underline{r}}$ at time t.

It is convenient to define the stiffness coefficients of $\bar{\underline{k}}_t^{EJ}$ in terms of the relative expansion joint displacements $\bar{\underline{u}}$ defined by

$$\bar{\underline{u}} = \begin{bmatrix} u_{Ax} \\ u_y \\ u_{Az} \\ u_{Bx} \\ u_S \\ u_{Bz} \end{bmatrix} \equiv \begin{bmatrix} r_{Ax} \\ r_y \\ r_{Az} \\ r_{Bx} \\ \theta_S \\ r_{Bz} \end{bmatrix}_J - \begin{bmatrix} r_{Ax} \\ r_y \\ r_{Az} \\ r_{Bx} \\ \theta_S \\ r_{Bz} \end{bmatrix}_I = \bar{\underline{r}}_J - \bar{\underline{r}}_I \tag{46}$$

and the corresponding expansion joint forces $\bar{\underline{F}}$ defined by

$$\bar{\underline{F}} = \begin{bmatrix} F_{Ax} \\ F_y \\ F_{Az} \\ F_{Bx} \\ F_S \\ F_{Bz} \end{bmatrix} = \begin{bmatrix} S_{Ax} \\ S_y \\ S_{Az} \\ S_{Bx} \\ M_S \\ S_{Bz} \end{bmatrix}_J = \bar{\underline{S}}_J = -\bar{\underline{S}}_I \tag{47}$$

Thus, the idealized expansion joint can be characterized by a stiffness matrix $\bar{\underline{k}}$ relating $\Delta\bar{\underline{F}}$ to $\Delta\bar{\underline{u}}$, i.e.,

$$\Delta\bar{\underline{F}} = \bar{\underline{k}}\ \Delta\bar{\underline{u}} \tag{48}$$

Stiffness matrix $\bar{\underline{k}}$ represents contributions from two sources. One contribution is the stiffness matrix $\bar{\underline{k}}_1$ of the idealization expansion joint without longitudinal restrainer ties and the other contribution is the matrix $\bar{\underline{k}}_2$ due to the presence of longitudinal ties, i.e.,

$$\bar{\underline{k}} = \bar{\underline{k}}_1 + \bar{\underline{k}}_2 \tag{49}$$

The stiffness coefficients of $\bar{\underline{k}}_1$ can be written in the form

$$\bar{\underset{-}{k}}_1 = \begin{bmatrix} k_A & 0 & 0 & 0 & 0 & 0 \\ 0 & k_S & 0 & 0 & 0 & 0 \\ 0 & 0 & k_V & 0 & 0 & 0 \\ 0 & 0 & 0 & k_B & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & k_V \end{bmatrix} \tag{50}$$

where

$$\begin{aligned} k_A &= k_I \quad \text{for} \quad u_{Ax} < -\Delta_G \text{ and } \dot{u}_{Ax} \le 0, \\ &= 0 \quad \text{for} \quad u_{Ax} < -\Delta_G \text{ and } \dot{u}_{Ax} > 0, \\ &\qquad \text{or} \quad u_{Ax} \ge -\Delta_G \\ k_B &= k_I \quad \text{for} \quad u_{Bx} < -\Delta_G \text{ and } \dot{u}_{Bx} \le 0, \\ &= 0 \quad \text{for} \quad u_{Bx} < -\Delta_G \text{ and } \dot{u}_{Bx} > 0, \\ &\qquad \text{or} \quad u_{Bx} \ge -\Delta_G \end{aligned} \tag{51}$$

and where k_S, k_V are those quantities defined previously.

The stiffness coefficient of each longitudinal restrainer tie within a time step can be expressed as

$$\Delta F_{Ti} = k_{Ti} \Delta u_{Ti} \quad , \quad (i=1,---,N_T) \tag{52}$$

where ΔF_{Ti} and Δu_{Ti} are, respectively, the increment of tie force F_{Ti} and the increment of tie deformation u_{Ti} in the ith tie during each time interval. The stiffness k_{Ti} is the instantaneous stiffness of the ith tie bar at time t, which can be expressed as

$$k_{Ti} \begin{cases} = 0 & \text{for} \quad u_{Ti} \le (\Delta_T + u^P_{Ti}) \,, \\ = k_T & \text{for} \quad (\Delta_T + u^P_{Ti}) < u_{Ti} \le (\Delta_T + u^P_{Ti} + u^E_T), \\ = 0 & \text{for} \quad u_{Ti} > (\Delta_T + u^P_{Ti} + u^E_T). \end{cases} \tag{53}$$

where $u^P_{Ti} \ge 0$ is the current plastic elongation of the ith tie, and u^E_T is the elastic elongation of the ties, i.e., $u^E_T = S_T/k_T$.

Since the end diaphragms are rigid, the tie deformation u_{Ti} can be determined from the longitudinal expansion joint deformations u_{Ax} and u_{Bx} by the relation

$$u_{Ti} = \left\langle \frac{1}{2} + \frac{2y_i}{d} \quad , \quad \frac{1}{2} - \frac{2y_i}{d} \right\rangle \begin{Bmatrix} u_{Ax} \\ u_{Bx} \end{Bmatrix} \, , \quad (i=1,2,---,N_T) \tag{54}$$

From conditions of equilibrium, the longitudinal expansion joint forces F_{Ax} and F_{Bx} can be expressed as the sum of all tie bar forces F_{Ti}, i.e.,

$$\begin{Bmatrix} F_{Ax} \\ F_{Bx} \end{Bmatrix} = \sum_{i=1}^{N_T} \begin{Bmatrix} \frac{1}{2} + \frac{2y_i}{d} \\ \frac{1}{2} - \frac{2y_i}{d} \end{Bmatrix} F_{Ti} \tag{55}$$

Combining Eq. (52), (53), and (54), the following relation is obtained

$$\begin{Bmatrix} F_{Ax} \\ F_{Bx} \end{Bmatrix} = \begin{bmatrix} k_{AA} & k_{AB} \\ k_{AB} & k_{BB} \end{bmatrix} \begin{Bmatrix} u_{Ax} \\ u_{Bx} \end{Bmatrix} \tag{56}$$

where

$$k_{AA} = \sum_{i=1}^{N_T} k_{Ti} \left(\frac{1}{2} + \frac{2y_i}{d} \right)^2$$

$$k_{AB} = \sum_{i=1}^{N_T} k_{Ti} \left(\frac{1}{2} + \frac{2y_i}{d} \right)\left(\frac{1}{2} - \frac{2y_i}{d} \right) \tag{57}$$

$$k_{BB} = \sum_{i=1}^{N_T} k_{Ti} \left(\frac{1}{2} + \frac{2y_i}{d} \right)^2$$

Thus, the stiffness matrix $\bar{\underline{k}}_2$ in Eq. (44) can be expressed as

$$\bar{\underline{k}}_2 = \begin{bmatrix} k_{AA} & 0 & 0 & k_{AB} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ k_{AB} & 0 & 0 & k_{BB} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{58}$$

Using Eqs. (46) and (47), the stiffness matrix $\bar{\underline{k}}_t^{EJ}$ can now be expressed as

$$\bar{\underline{k}}_t^{EJ} = \begin{bmatrix} \bar{\underline{k}} & -\bar{\underline{k}} \\ -\bar{\underline{k}} & \bar{\underline{k}} \end{bmatrix} \tag{59}$$

Finally, the stiffness matrix $\bar{\underline{k}}_t^{EJ}$ must be transformed to the local nodal coordinate system using Eqs. (40) and (44). This transformation results in the relation

$$\Delta \underline{S} = \underline{k}_t^{EJ} \; \Delta \underline{r} \quad , \quad \underline{k}_t^{EJ} = \underline{A}^T \; \bar{\underline{k}}_t^{EJ} \; \underline{A} \tag{60}$$

Matrix $\underset{\sim}{k}_t^{EJ}$ is the stiffness matrix which relates the vector of local nodal force increments $\Delta\underset{\sim}{S}$ to the vector of local nodal displacement increments $\Delta\underset{\sim}{r}$ at time t, and which, after local to global coordinate transformation, can be used to assemble the total stiffness $\underset{\sim}{K}_t$ for the complete structural system.

At the end of each time interval during the analysis process, joint conditions as expressed by Eq. (51) and the condition of each longitudinal restrainer tie are checked and expansion joint stiffnesses are formed accordingly. The quantities, u_{Ax}, u_{Bx}, and u_{Ti}^P, $(i=1,2,---,N_T)$ are carefully monitored since the maximum value of u_{Ax} and u_{Bx} represent the maximum expansion joint separations at points A and B and the maximum values of u_{Ti}^P, $(i=1,2,---,N_T)$ indicate the maximum ductilities that the longitudinal restrainer ties should provide.

C. COULOMB FRICTION FORCES

Coulomb friction forces C_{Ax} and C_{Bx}, which develop at contact points A and B when the expansion joint undergoes longitudinal relative displacements and when the vertical contact forces F_{Az} and F_{Bz} are compressive, can be considered as pairs of self-equilibrating forces acting on the expansion joint along the longitudinal x axis at points A and B. According to the assumptions stated previously, these forces can be expressed as

$$\begin{aligned} C_{Ax} &= \nu \langle F_{Az} \rangle \left| F_{Az} \right| \text{sign } (\dot{u}_{Ax}) \\ C_{Bx} &= \nu \langle F_{Bz} \rangle \left| F_{Bz} \right| \text{sign } (\dot{u}_{Bx}) \end{aligned} \qquad (61)$$

where ν is a constant coefficient of friction and where

$$\langle F_{Az} \rangle \begin{array}{ll} = 1 & \quad F_{Az} < 0 \\ = 0 & \quad F_{Az} \geq 0 \end{array} \qquad (62)$$

$$\langle F_{Bz} \rangle \begin{array}{ll} = 1 & \quad F_{Bz} < 0 \\ = 0 & \quad F_{Bz} \geq 0 \end{array} \qquad (63)$$

If it is assumed that the relative velocities $\dot{u}_{Ax}$ and $\dot{u}_{Bx}$, and the forces F_{Az} and F_{Bz} do not change signs during a time interval Δt, then the changes of Coulomb friction forces during the interval can be expressed by

$$\begin{aligned} \Delta C_{Ax} &= \nu \langle F_{Az} \rangle \,|\, \Delta F_{Az} \,|\, \text{sign } (\dot{u}_{Ax}) \\ \Delta C_{Bx} &= \nu \langle F_{Bz} \rangle \,|\, \Delta F_{Bz} \,|\, \text{sign } (\dot{u}_{Bx}) \end{aligned} \tag{64}$$

Let $\Delta \underline{\bar{P}}^C$ represent the incremental Coulomb friction force vector in the expansion joint coordinate system during time interval Δt. Then $\Delta \underline{\bar{P}}^C$ can be expressed as

$$\Delta \underline{\bar{P}}^C = \left\{ \begin{matrix} \Delta \bar{P}^C_{-I} \\ \\ \Delta \bar{P}^C_{-J} \end{matrix} \right\} , \quad \Delta \bar{P}^C_{-I} = \begin{bmatrix} \Delta C_{Ax} \\ 0 \\ 0 \\ \Delta C_{Bx} \\ 0 \\ 0 \end{bmatrix} , \quad \Delta \bar{P}^C_{-J} = \begin{bmatrix} -\Delta C_{Ax} \\ 0 \\ 0 \\ -\Delta C_{Bx} \\ 0 \\ 0 \end{bmatrix} \tag{65}$$

Thus, the incremental Coulomb friction force vector $\Delta \underline{P}^C$ in the local nodal coordinate system can be obtained using Eq. (44); i.e.

$$\Delta \underline{P}^C = \underline{A}^T \Delta \underline{\bar{P}}^C \tag{66}$$

When $\Delta \underline{P}^C$ for all expansion joints in the bridge system are transformed to the global coordinate system and are assembled, an incremented Coulomb friction force vector $\Delta \underline{R}^C$ in the global coordinate system is obtained. The vectors $\Delta \underline{R}^C$ are then added to the incremental external force vectors during each time interval.

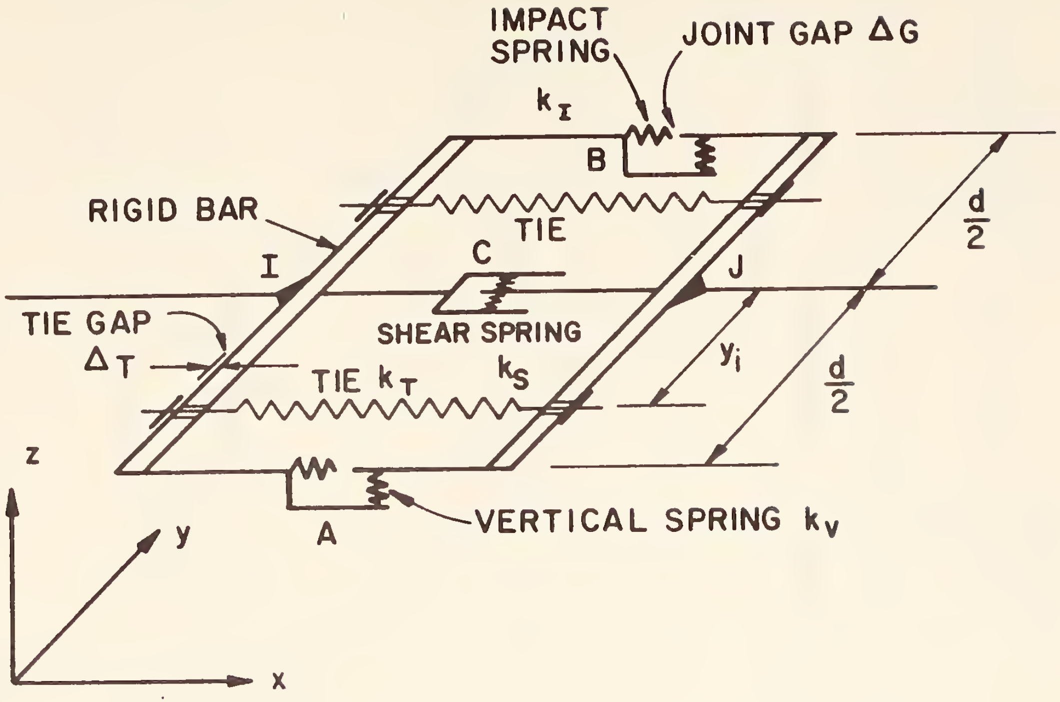
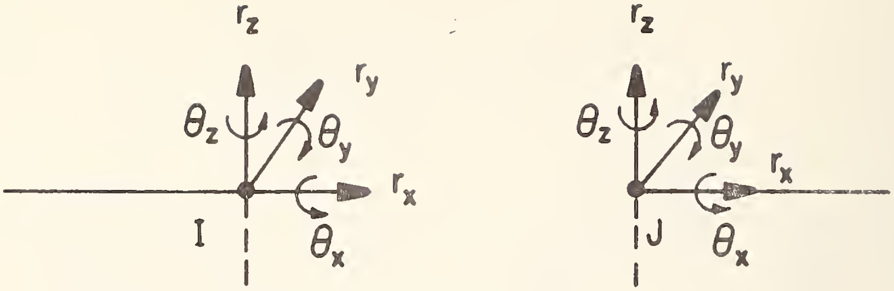
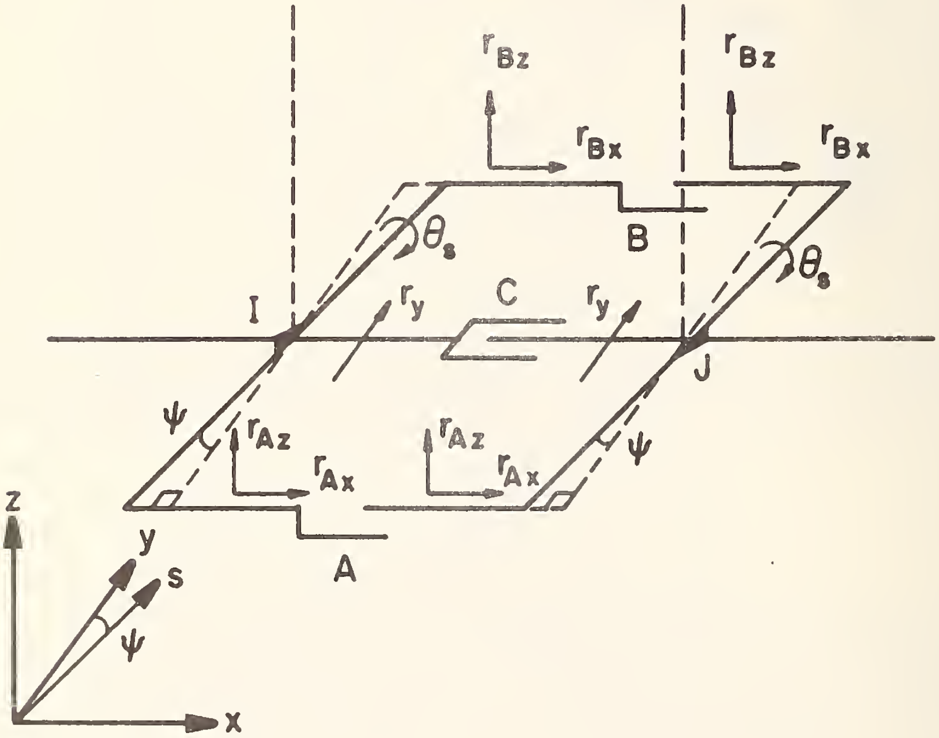


Fig. 12 Idealized Bridge Expansion Joint



LOCAL NODAL COORDINATE SYSTEM $\underline{r}$



EXPANSION JOINT COORDINATE SYSTEM $\underline{\bar{r}}$

Fig. 13 Coordinate Systems For Expansion Joint

A. EQUATIONS OF MOTION

The equations of motion for an n degree of freedom system expressing dynamic equilibrium at time t can be expressed in the standard matrix form

$$\underline{M}\,\ddot{\underline{u}}_t + \underline{C}_t\dot{\underline{u}}_t + \underline{K}_t\underline{u}_t = \underline{R}(t) \tag{67}$$

where $\underline{M}$, $\underline{C}_t$, and $\underline{K}_t$ are the mass, damping, and stiffness matrices, respectively, and where $\underline{R}(t)$ is the applied dynamic load vectors. Vectors $\ddot{\underline{u}}_t$, $\dot{\underline{u}}_t$, and $\underline{u}_t$ are the absolute acceleration, velocity, and displacement vectors, respectively, as measured with respect to a fixed set of global coordinate axes.

If the system is subjected to prescribed support excitations, a complete set of nodal displacements $\underline{u}_t^c$ should be considered which include, in addition to the n free nodal displacements, the n^b prescribed non-zero support displacements. Thus, the complete nodal displacement vector can be expressed in the partitioned form

$$\underline{u}_t^c = \begin{Bmatrix} \underline{u}_t \\ \underline{u}_t^b \end{Bmatrix} \tag{68}$$

where $\underline{u}_t^b$ is a vector containing the n^b non-zero support displacements (translations and/or rotations). Vector $\underline{u}_t^c$ can be conveniently decomposed into a quasi-static displacement vector $\underline{u}_s^c$ and a dynamic displacement vector $\underline{u}^c$, i.e.

$$\underline{u}_t^c = \begin{Bmatrix} \underline{u}_t \\ \\ \underline{u}_t^b \end{Bmatrix} = \begin{Bmatrix} \underline{u}_s \\ \\ \underline{u}_s^b \end{Bmatrix} + \begin{Bmatrix} \underline{u} \\ \\ \underline{u}^b \end{Bmatrix} \equiv \underline{u}_s^c + \underline{u}^c \tag{69}$$

where by definition $\underline{u}^b = \underline{0}$. Enlarging the mass, damping, and stiffness matrices as well as the dynamic load vector in Eq. (67) to account for the n^b support displacements, the equations of motion for the complete system can be expressed in the partitioned matrix form

$$\left[\begin{array}{c|c} \underline{M} & \underline{M}^{b} \\ \hline (\underline{M}^{b})^{T} & \underline{M}^{bb} \end{array}\right] \left\{\begin{array}{c} \ddot{\underline{u}}_{t} \\ \hline \ddot{\underline{u}}_{t}^{b} \end{array}\right\} + \left[\begin{array}{c|c} \underline{C}_{t} & \underline{C}_{t}^{b} \\ \hline (\underline{C}_{t}^{G})^{T} & \underline{C}_{t}^{bb} \end{array}\right] \left\{\begin{array}{c} \dot{\underline{u}}_{t} \\ \hline \dot{\underline{u}}_{t}^{b} \end{array}\right\} + \left[\begin{array}{c|c} \underline{K}_{t} & \underline{K}_{t}^{b} \\ \hline (\underline{K}_{t}^{b})^{T} & \underline{K}_{t}^{bb} \end{array}\right] \left\{\begin{array}{c} \underline{u}_{t} \\ \hline \underline{u}_{t}^{b} \end{array}\right\}$$

$$= \left\{\begin{array}{c} \underline{R}(t) \\ \hline \underline{R}^{b}(t) \end{array}\right\} \tag{70}$$

The equations of motion associated with the n free nodal displacements now become

$$\left[\underline{M}\ \underline{M}^{b}\right] \left\{\begin{array}{c} \ddot{\underline{u}}_{t} \\ \hline \ddot{\underline{u}}_{t}^{b} \end{array}\right\} + \left[\underline{C}_{t}\ \underline{C}_{t}^{b}\right] \left\{\begin{array}{c} \dot{\underline{u}}_{t} \\ \hline \dot{\underline{u}}_{t}^{b} \end{array}\right\} + \left[\underline{K}_{t}\ \underline{K}_{t}^{b}\right] \left\{\begin{array}{c} \underline{u}_{t} \\ \hline \underline{u}_{t}^{b} \end{array}\right\} = \underline{R}(t) \tag{71}$$

Substituting Eq. (69) into Eq. (71), one obtains

$$\underline{M}\,\ddot{\underline{u}} + \underline{C}_{t}\,\dot{\underline{u}} + \underline{K}_{t}\,\underline{u} = \underline{R}(t) - \left[\underline{M}\ \underline{M}^{b}\right] \left\{\begin{array}{c} \ddot{\underline{u}}_{s} \\ \ddot{\underline{u}}_{s}^{b} \end{array}\right\} - \left[\underline{C}_{t}\ \underline{C}_{t}^{b}\right] \left\{\begin{array}{c} \dot{\underline{u}}_{s} \\ \dot{\underline{u}}_{s}^{b} \end{array}\right\} \tag{72}$$

after making use of the relation

$$\underline{K}_{t}\,\underline{u}_{s} + \underline{K}_{t}^{b}\,\underline{u}_{s}^{b} = \underline{0}\ , \tag{73}$$

which is satisfied by definition of the quasi-static vector $\underline{u}_{s}$. This vector can thus be obtained directly from Eq. (73), i.e.,

$$\underline{u}_{s} = -\underline{K}_{t}^{-1}\,\underline{K}_{t}^{b}\,\underline{u}_{s}^{b} \equiv -\underline{B}_{t}\,\underline{u}_{s}^{b} \tag{74}$$

where $\underline{B}_{t} \equiv \underline{K}_{t}^{-1}\,\underline{K}_{t}^{b}$ is a matrix of quasi-static influence coefficients resulting from the n^{b} non-zero support displacements. If the system is linear, all coefficients in matrix $\underline{B}_t$ are invariant with time.

Usually the damping terms on the right hand side of Eq. (72) are small compared to the inertia terms and therefore may be dropped from the equation without introducing significant errors. Also, the coefficients in matrix $\underline{M}^b$ can be set equal to zero since mass coupling vanishes for a lumped mass model. Thus, after substituting Eq. (74) into Eq. (72), the equations of motion reduce to the form

$$\underline{M}\;\ddot{\underline{u}} + \underline{C}_t\;\dot{\underline{u}} + \underline{K}_t\;\underline{u} = \underline{R}(t) + \underline{M}\;\underline{B}_t\;\ddot{\underline{u}}_s^b \tag{75}$$

where $\ddot{\underline{u}}_s^b$ is a vector containing the prescribed support excitations.

1. Multiple Support Excitations - When ground excitations corresponding to each of the n^b support displacements are prescribed by a vector $\ddot{\underline{u}}_g^m(t)$, the vector $\ddot{\underline{u}}_s^b$ in Eq. (75) can be expressed as

$$\ddot{\underline{u}}_s^b = \ddot{\underline{u}}_g^m(t) \tag{76}$$

and the equations of motion become

$$\underline{M}\;\ddot{\underline{u}} + \underline{C}_t\;\dot{\underline{u}} + \underline{K}_t\;\underline{u} = \underline{R}(t) + \underline{M}\;\underline{B}_t\;\ddot{\underline{u}}_g^m(t) \tag{77}$$

2. Rigid Support Excitations - When ground excitations at all support points along the base of the structure are identical and are prescribed by a rigid ground acceleration vector $\ddot{\underline{u}}_g^r$ consisting of three translational components $\ddot{u}_{gX}$, $\ddot{u}_{gY}$, and $\ddot{u}_{gZ}$, measured along their corresponding global axes X, Y, and Z, i.e.,

$$\ddot{\underline{u}}_g^r = \begin{Bmatrix} \ddot{u}_{gX} \\ \ddot{u}_{gY} \\ \ddot{u}_{gZ} \end{Bmatrix} \tag{78}$$

the equations of motion become

$$\underline{M}\;\ddot{\underline{u}} + \underline{C}_t\;\dot{\underline{u}} + \underline{K}_t\;\underline{u} = \underline{R}(t) + \underline{M}\;\underline{B}^r\;\ddot{\underline{u}}_g^r\,(t) \tag{79}$$

reliable phase difference predictions can be made. Therefore, ground motions in the form of rigid base motions are used in analyses.

C. LINEAR ANALYSIS PROCEDURES - MODE SUPERPOSITION METHOD

When the structural system is linear, the stiffness and damping matrices are invariant with time, i.e., $\underline{K}_t = \underline{K}$ and $\underline{C}_t = \underline{C}$. Thus Eq. (81) takes on the linear form

$$\underline{M}\,\ddot{\underline{u}} + \underline{C}\,\dot{\underline{u}} + \underline{K}\,\underline{u} = \underline{R}(t) + \underline{M}\,\underline{B}\,\ddot{\underline{u}}_g(t) \tag{82}$$

A standard procedure can be followed in obtaining dynamic response, namely, solving the generalized eigenvalue problem for mode shapes and frequencies, solving a decoupled set of normal equations of motion, and using mode superposition to obtain time histories of response [18].

1. Mode Shapes and Frequencies - The desired undamped free vibration mode shapes and corresponding frequencies can be obtained by solving the equation

$$\underline{K}\,\underline{\phi}_i = \omega_i^2\,\underline{M}\,\underline{\phi}_i \qquad (i=1,2,---,q);\ q \le n \tag{83}$$

where ω_i and $\underline{\phi}_i$ are the frequency and shape vector, respectively, for the ith mode and where q is the number of lowest modes required for the accuracy of solution desired. Usually, q is much less than n and can be efficiently determined using a determinant search technique combined with inverse iteration or a subspace iteration technique as recently developed by Bathe [26].

The modal matrix $\underline{\Phi} \equiv [\underline{\phi}_1\ \underline{\phi}_2 --- \underline{\phi}_q]$ must satisfy the orthogonality condition

$$\underline{\Phi}^T\,\underline{K}\,\underline{\Phi} = \underline{\Omega}^2 \tag{84}$$

where $\underline{\Omega}^2$ is a diagonal matrix containing the squared frequencies $\omega_1^2, \omega_2^2, ---, \omega_q^2$. It is convenient to normalize the modal matrix so that it satisfies the condition

$$\underline{\Phi}^T \underline{M}\, \underline{\Phi} = \underline{I} \tag{85}$$

where $\underline{I}$ is the unit (or identity) matrix.

Vectors $\underline{u}(t)$, $\underline{\dot{u}}(t)$, and $\underline{\ddot{u}}(t)$ can be expressed in terms of the modal matrix and the normal coordinate (mode amplitude) vector $\underline{\upsilon}(t)$ as follows:

$$\underline{u} = \underline{\Phi}\, \underline{\upsilon} \; ; \; \underline{\dot{u}} = \underline{\Phi}\, \underline{\dot{\upsilon}} \; ; \; \underline{\ddot{u}} = \underline{\Phi}\, \underline{\ddot{\upsilon}} \tag{86}$$

Substituting Eq. (86) into Eq. (82), premultiplying the resulting equation by $\underline{\Phi}^T$, and making use of Eq.(84) and (85), one obtains

$$\underline{\ddot{\upsilon}} + \underline{\Lambda}\, \underline{\upsilon} + \underline{\Omega}^2\, \underline{\upsilon} = \underline{R}^*(t) \tag{87}$$

where $\underline{\Lambda}$ is a diagonal matrix containing the terms $2\xi_i\omega_i$, for $i=1,2,---,q$ and where $\underline{R}^*(t)$ is a normal load vector defined by

$$\underline{R}^*(t) = \underline{\Phi}^T \left[\underline{R}(t) + \underline{M}\, \underline{B}\, \underline{\ddot{u}}_g(t) \right] \tag{88}$$

Using Eq. (6), matrix $\underline{\Lambda}$ can be expressed in the form

$$\underline{\Lambda} = \underline{\Phi}^T \underline{C}\, \underline{\Phi} = \underline{\Phi}^T [\, \alpha\underline{M} + \beta\underline{K}\,] \underline{\Phi} = \alpha\underline{I} + \beta\underline{\Omega}^2 \tag{89}$$

2. <u>Response Time Histories</u> - The solution of Eq. (87) can be carried out in the time domain using the convolution integral [18]

$$\upsilon_i(t) = \frac{1}{\omega_{Di}} \int_0^t R_i^*(\tau)\, e^{-\xi_i\omega_i(t-\tau)} \sin \omega_{Di}(t-\tau)\, d\tau \tag{90}$$

for $i=1,2,---,q$;

where ω_{Di} is the damped frequency

$$\omega_{Di} = \omega_i (1-\xi_i^2)^{\frac{1}{2}} \tag{91}$$

Thus, the response time histories $\underline{u}(t)$, $\dot{\underline{u}}(t)$, and $\ddot{\underline{u}}(t)$ can now be obtained using Eq. (86).

3. Spectral Responses - For a given single component of earthquake ground acceleration $\ddot{u}_g(t)$, the displacement response spectra $S_d(\omega,\xi)$ define the maximum (or extreme) values of displacement governed by the equation [27]

$$\ddot{\upsilon} + 2\xi\omega\,\dot{\upsilon} + \omega^2\,\upsilon = \ddot{u}_g(t) \tag{92}$$

These spectra are usually obtained directly from the pseudo-velocity response spectra as defined by

$$S_d(\omega,\xi) \equiv \frac{S_v(\omega,\xi)}{\omega} \tag{93}$$

Suppose these response spectra are known for each ground acceleration input in the vector $\ddot{\underline{u}}_g(t)$ appearing in Eq. (88) and suppose the $\underline{R}(t) = \underline{0}$, then the modal maximum (or spectral) response can be obtained from the relation

$$|\,\upsilon_i(t)\,|_{max.} = \underline{\phi}_i^T\,\underline{M}\,\underline{B}\,\underline{S}_d(\omega_i,\xi_i) \qquad (i=1,2,---,q) \tag{94}$$

The maximum displacement response $|\underline{u}_i(t)|_{max.}$ can then be determined using

$$|\,\underline{u}_i(t)\,|_{max.} = \underline{\phi}_i\,|\upsilon_i(t)|_{max.} \qquad (i=1,2,---,q) \tag{95}$$

Thus, the root-mean-square spectral displacement response can be expressed as

$$\left| \underline{u}(t) \right|_{R.M.S.} \equiv \left\{ \sum_{i=1}^{q} \left| \underline{u}_i(t) \right|^2_{max.} \right\}^{\frac{1}{2}} \tag{96}$$

D. NONLINEAR ANALYSIS PROCEDURE - STEP-BY-STEP INTEGRATION

When the structural system is nonlinear, the coupled equations of motion, Eq. (81), must be solved by step-by-step integration methods since mode superposition no longer applies. To carry out this integration, the equations of motion have been formulated on an incremental basis. Considering a time interval Δt starting at time t and assuming the stiffness and damping matrices at time t, i.e., $\underline{K}_t$ and $\underline{C}_t$, can be applied over the full time interval, one obtains the equations of motion in the incremental form

$$\underline{M}\,\Delta\ddot{\underline{u}} + \underline{C}_t\,\Delta\dot{\underline{u}} + \underline{K}_t\,\Delta\underline{u} = \Delta\,\underline{R}(t) + \underline{M}\,\underline{B}\,\Delta\ddot{\underline{u}}_g(t) \tag{97}$$

where

$$\begin{aligned} \Delta\ddot{\underline{u}} &= \Delta\ddot{\underline{u}}(t) = \ddot{\underline{u}}(t+\Delta t) - \ddot{\underline{u}}(t) \\ \Delta\dot{\underline{u}} &= \Delta\dot{\underline{u}}(t) = \dot{\underline{u}}(t+\Delta t) - \dot{\underline{u}}(t) \\ \Delta\underline{u} &= \Delta\underline{u}(t) = \underline{u}(t+\Delta t) - \underline{u}(t) \end{aligned} \tag{98}$$

and where

$$\begin{aligned} \Delta\underline{R}(t) &= \underline{R}(t+\Delta t) - \underline{R}(t) \\ \Delta\ddot{\underline{u}}_g(t) &= \ddot{\underline{u}}_g(t+\Delta t) - \ddot{\underline{u}}_g(t) \end{aligned} \tag{99}$$

To carry out the step-by-step numerical integration of Eq. (97), it is necessary to introduce an approximate operator in the time domain to replace the differential operator. In the present analysis, the Newmark generalized acceleration method [28] is adopted which assumes the following approximations for nodal velocities and displacements:

$$
\begin{aligned}
\underline{\dot{u}}(t+\Delta t) &= \underline{\dot{u}}(t) + [(1-\delta)\,\underline{\ddot{u}}(t) + \delta\,\underline{\ddot{u}}(t+\Delta t)] \\
\underline{u}(t+\Delta t) &= \underline{u}(t) + \underline{\dot{u}}(t)\,\Delta t + [(\tfrac{1}{2}-\sigma)\underline{\ddot{u}}(t) + \sigma\,\underline{\ddot{u}}(t+\Delta t)]
\end{aligned}
\qquad (100)
$$

where parameters δ and σ can be chosen to give the required integration stability and accuracy. When $\delta = \frac{1}{2}$ and $\sigma = \frac{1}{6}$, the approximations correspond to the linear acceleration method, and when $\delta = \frac{1}{2}$ and $\sigma = \frac{1}{4}$ they correspond to the constant acceleration method. While the linear acceleration method is conditionally stable depending upon the magnitude of Δt, the constant acceleration method is unconditionally stable for any magnitude of Δt [29,30].

Using Eq. (97), the approximations given by Eq. (100) can be expressed in the incremental form

$$
\begin{aligned}
\Delta\underline{\ddot{u}}(t) &= C_1\,\Delta\underline{u}(t) - C_3\,\underline{\dot{u}}(t) - C_4\,\underline{\ddot{u}}(t) \\
\Delta\underline{\dot{u}}(t) &= C_2\,\Delta\underline{u}(t) - C_4\,\underline{\dot{u}}(t) - C_5\,\underline{\ddot{u}}(t)
\end{aligned}
\qquad (101)
$$

where

$$
\begin{aligned}
&C_1 = \frac{4}{\Delta t^2}\ , \qquad C_2 = \frac{2}{\Delta t}\ , \qquad C_3 = \frac{4}{\Delta t} \\
&C_4 = 2\ , \qquad C_5 = 0\ ;
\end{aligned}
\qquad (102)
$$

for the constant acceleration method and where

$$
\begin{aligned}
&C_1 = \frac{6}{\Delta t^2}\ , \qquad C_2 = \frac{3}{\Delta t}\ , \qquad C_3 = \frac{6}{\Delta t} \\
&C_4 = 3\ , \qquad C_5 = \frac{\Delta t}{2}\ ,
\end{aligned}
\qquad (103)
$$

for the linear acceleration method.

Substituting Eq. (101) into Eq. (97), one obtains after some manipulation

$$\left[C_1\,\underline{M} + C_2\,\underline{C}_t + \underline{K}_t\right]\,\Delta\underline{u} = \Delta\underline{\bar{\bar{R}}}(t) \tag{104}$$

where

$$\begin{aligned}\Delta\underline{\bar{\bar{R}}}(t) = \Delta\underline{R}(t) + \underline{M}\,\underline{B}\,\Delta\ddot{\underline{u}}_g(t) &+ \left\{C_3\underline{M} + C_4\underline{C}_t\right\}\dot{\underline{u}}(t) \\ &+ \left\{C_4\underline{M} + C_5\underline{C}_t\right\}\ddot{\underline{u}}(t)\end{aligned} \tag{105}$$

Using Eq. (8), i.e., $\underline{C}_t = \alpha\underline{M} + \beta\underline{K}_t$, and introducing the following normalized constants

$$\begin{aligned} C_7 &= 1/(1+\beta\,C_2) \quad , & C_8 &= C_7\,(C_1 + \alpha C_2) \\ C_9 &= C_7\,(C_3 + \alpha C_4) \quad , & C_{10} &= \beta C_4 C_7 \\ C_{11} &= C_7\,(C_4 + \alpha C_5) \quad , & C_{12} &= \beta C_5 C_7 \end{aligned} \tag{106}$$

Eq. (104) can be put into the form

$$\begin{aligned}\left\{C_8\underline{M} + \underline{K}_t\right\}\Delta\underline{u} = C_7 &\left\{\Delta\underline{R}(t) + \underline{M}\,\underline{B}\,\Delta\ddot{\underline{u}}_g(t)\right\} \\ &+ \left\{C_9\,\underline{M} + C_{10}\underline{K}_t\right\}\dot{\underline{u}}(t) + \left\{C_{11}\underline{M} + \underline{C}_{12}\underline{K}_t\right\}\ddot{\underline{u}}(t)\end{aligned} \tag{107}$$

Defining an effective displacement vector $\Delta\bar{\underline{u}}(t)$ by the relation

$$\Delta\bar{\underline{u}}(t) = \Delta\underline{u}(t) - C_{10}\,\dot{\underline{u}}(t) - C_{12}\,\ddot{\underline{u}}(t) \tag{108}$$

and defining two new constants

$$\begin{aligned} C_{13} &\equiv C_9 - C_8\,C_{10} \\ C_{14} &\equiv C_{11} - C_8\,C_{12} \end{aligned} \tag{109}$$

Eq.(107) can be written in the form

$$\bar{\underline{K}}_t\,\Delta\bar{\underline{u}} = \Delta\bar{\underline{R}}(t) \tag{110}$$

where

$$\bar{\underline{K}}_t = C_8 \underline{M} + \underline{K}_t \tag{111}$$

and

$$\Delta\bar{\underline{R}}(t) = C_7 \left\{ \Delta\underline{R}(t) + \underline{M}\,\underline{B}\,\ddot{\underline{u}}_g(t) \right\} + C_{13}\,\underline{M}\,\dot{\underline{u}}(t) + C_{14}\,\underline{M}\,\ddot{\underline{u}}(t) \tag{112}$$

Matrix $\bar{\underline{K}}_t$ is the effective dynamic stiffness matrix and $\Delta\bar{\underline{R}}(t)$ is the effective load vector.

Equation (110) can be solved for $\Delta\bar{\underline{u}}(t)$ at each time instant t and Eqs. (108) and (101) can be used to obtain $\Delta\underline{u}(t)$, $\Delta\dot{\underline{u}}(t)$, and $\Delta\ddot{\underline{u}}(t)$. The displacements, velocities, and accelerations at time t + Δt can then be determined from Eq. (98).

The displacement solutions $\underline{u}(t + \Delta t)$ can be used to calculate the internal force vector $\underline{S}(t + \Delta t)$ and the new tangent stiffness matrix $\underline{k}_{t + \Delta t}$ for each nonlinear element in the system. The new total tangent stiffness matrix $\underline{K}_{t + \Delta t}$ is then obtained by the standard assemblage procedure. It is important to note that when determining the element internal forces, the static forces existing in the element must always be included since the calculation of element tangent stiffnesses depend upon the magnitudes of the element forces.

The step-by-step integration algorithm for the nonlinear system presented above may be summarized as follows:

(1) Initial calculations
   (a) Form the initial stiffness matrix $\underline{K}$ and the mass matrix $\underline{M}$ for the system.
   (b) Solve for the initial displacements and the element forces due to static loads.
   (c) Compute the initial quasi-static influence matrix $\underline{B}$, if required.

(d) Set up the dynamic load and ground excitation time histories $\underline{R}(t)$ and $\ddot{\underline{u}}_g(t)$.

(e) Calculate the step-by-step integration constants, C_i, $i=1,2,---,14$.

(2) For each time increment Δt at time t

(a) Form the tangent stiffness matrix $\underline{K}_t$.

(b) Compute the current quasi-static influence matrix $\underline{B}_t$, if required.

(c) Form the effective dynamic stiffness matrix $\bar{\underline{K}}_t$.

(d) Triangularize $\bar{\underline{K}}_t$.

(e) Form the effective load vector $\Delta\bar{\underline{R}}(t)$.

(f) Solve for the effective displacement increments $\Delta\bar{\underline{u}}(t)$

(g) Compute current accelerations, velocities, and displacements $\ddot{\underline{u}}(t)$, $\dot{\underline{u}}(t)$, $\underline{u}(t)$.

(h) Calculate current element forces, check nonlinearity conditions, and compute new element tangent stiffness matrices, inelastic deformation vectors, and unbalanced force vectors, if necessary.

(i) Return to step (a) or (e), as necessary, for next time increment.

VI LINEAR AND NONLINEAR SEISMIC ANALYSIS COMPUTER PROGRAMS

A. LINEAR ANALYSIS COMPUTER PROGRAM - BSAP

A computer program BSAP has been developed for dynamic response analysis of linear bridge structural systems. This program is a modified version of the SAP program originally developed by Wilson and co-workers for static and dynamic analyses of linear structural systems [31]. Although the original version of SAP is very efficient and can be used for analyzing a wide variety of structures, some modifications are necessary to make it suitable for the analysis of bridge structural systems. While most of BSAP follows the organization of SAP [31], additional development and modifications have been made on the SAP program as follows:

(1) New eigenvalue solution routines using a determinant search technique combined with inverse iteration or subspace iteration, both developed by Bathe [26], have been incorporated into the program for the determination of frequencies and mode shapes.

(2) In the dynamic response time history analysis option of the program, modifications have been made to permit multiple support excitations of the structural system. One may wish to specify this type of input for some bridge structures which are extremely long.

(3) A subroutine which optimizes the block storage of the stiffness and mass matrices of the system on the basis of available core storage and analysis type has been incorporated into the program.

(4) A linear elastic circularly curved beam element has been developed and added to the element library for modelling curved decks.

(5) A three-dimensional boundary spring element has been added to the element library for modelling foundation stiffnesses.

(6) A linear expansion joint element has been developed and added to the element library for modelling joint behavior.

Thus, the present version of BSAP can be used to analyze linear bridge structural systems modelled by any combination of the following element types: (1) three-dimensional truss element, (2) three-dimensional straight beam element, (3) three-dimensional curved beam element, (4) three-dimensional boundary spring element, and (5) linear expansion joint element. It can evaluate (1) response to static loadings, (2) frequencies and mode shapes, (3) time history responses to apply dynamic loadings and/or prescribed ground excitations (rigid or multiple), and (4) modal spectral responses and the root-mean-square values using prescribed earthquake spectra.

The organization of BSAP follows the linear analysis procedures established previously which can be described by the following operational sequence [31]:

(1) Nodal coordinates and the degrees of freedom of the system are generated from nodal input data. The nodal points may be specified as fully or partially constrained in any of its displacement coordinates. Nodes at multiple support input points are identified separately.

(2) Element data are input; then, matrices of stiffness, mass, and local-to-global coordinate transformation are generated and placed in low-speed storage units, e.g., disc files.

(3) Total stiffness and mass matrices for the complete system are assembled in one or more core blocks according to element connection arrays. Load vectors are formed in

the case of static analysis and the support coupling stiffness matrix $\underline{K}^b$ is assembled in the case of time history response analysis when considering multiple support excitations.

(4) Equilibrium equations are solved for static response or for quasi-static influence matrix $\underline{B}$ when prescribing multiple support excitations.

(5) For dynamic analysis, a specified number of lowest frequencies and mode shapes are determined.

(6) For time history response analysis, applied dynamic load and/or ground excitation time functions are input and digitized at equal time intervals. The response time histories are calculated using mode superposition procedure as described previously.

(7) For spectral response analysis, earthquake response spectra are input and modal spectral responses, as well as the root-mean-square responses, are determined based on prescribed modal damping ratios and the computed vibration frequencies.

Computer program BSAP is coded in FORTRAN IV. All storages are allocated in terms of variable dimensions at the time of execution, and an out-core block-by-block solution technique is adopted. Therefore, the program is capable of analyzing large bridge systems. It has been used to analyze several bridge structures using CDC 6600 and 7600 computers.

B. NONLINEAR ANALYSIS COMPUTER PROGRAM - NEABS

A new computer program NEABS has been developed for earthquake analysis of nonlinear bridge structural systems of the type considered herein. This computer program uses the step-by-step integration procedures established in the previous chapter. Either the linear acceleration or constant acceleration method can be chosen for integration. The excitations can result from applied dynamic loads and/or from support motions (rigid or multiple). The program has an element library consisting of the five linear elements used in BSAP and nonlinear elements as follows: (1) three-dimensional elasto-plastic flexural column (straight) element for reinforced concrete members, (2) three-dimensional bilinear boundary spring element, and (3) nonlinear expansion joint element. In the program, a nonlinear element indicator has been used to allow a system modelled with both linear and nonlinear elements.

The organization of computer program NEABS can be described by the operational sequence:

(1) Static linear equilibrium equations are first generated and assembled in the same manner as in the linear analysis program BSAP. They are then solved for the static responses to be used as initial conditions in the subsequent nonlinear earthquake response analysis. In this same process, the initial quasi-static influence matrix is also determined, if needed.

(2) Dynamic loads and/or ground excitation time functions are input and are digitized for equal time intervals.

(3) Checks are then made on the available core storage to determine whether an in-core or an out-core block-by-block solution process is appropriate.

(4) For an in-core solution, an in-core step-by-step solution package is used to determine the response time histories of the system based on the procedures described previously.

(5) If an out-core block-by-block solution is necessary, an out-core step-by-step solution package can be used to evaluate the response time histories.

Computer program NEABS is written in FORTRAN IV and is coded in variable-dimension form. The program has been used to determine the nonlinear earthquake responses of a number of bridge structures using the CDC 6600 and 7600 computers.

Because of the extensive computational effort involved in a nonlinear earthquake response analysis, the out-core step-by-step solution package requires many slow-speed storage operations. Therefore, this procedure can be very costly and insufficient. For the bridge structures analyzed herein, the out-core solution has not been required.

VII NUMERICAL RESULTS

The previously described mathematical modelling and analysis procedures have been employed to determine the response of over 10 bridge structural systems using computer programs BSAP and NEABS. These systems are basically patterned after 3 prototype bridge structures but with variations in certain parameters allowed. The 3 prototype structures are the 5/14 South Connector Overcrossing of the Golden State freeway and Antelope Valley freeway interchange, the curved Figueroa Street Undercrossing Connector in the Los Angeles area, and a straight version of the Figueroa Street Undercrossing Connector. The structural data and numerical results obtained for each bridge system are presented subsequently along with summaries of extreme-values of response.

A. 5/14 SOUTH CONNECTOR OVERCROSSING

1. The Structural System - The structural system of this bridge consists of a curved, continuous, reinforced concrete box girder deck of 9 spans (Nos. 1-9 from left to right in Fig. 14) supported on 8 single, central, reinforced concrete columns (Nos. 2-9 in Fig. 14), and 2 reinforced concrete diaphragm abutments (Nos. 1 and 10 in Fig. 14). The deck has a radius of curvature of 667 ft. measured to its center line. Four expansion joints (1-4 from left to right in Fig. 14) divide the deck into 5 continuous segments. The structural layout of this bridge system is shown in Fig. 14 and the structural properties of the bridge components are given in Table 1. During the San Fernando earthquake of February 9, 1971, this bridge suffered severe damage. Central spans Nos. 3 and 4 collapsed along with column No. 4.

For purpose of analysis, this bridge has been idealized by a discrete-parameter system consisting of 71 nodal points of which the 10 located at the bases of the 8 columns and the 2 abutments are fixed to the ground; thus, a total of 366 (61x6) degrees of freedom are present in the system (Fig. 15). For linear analysis, a system of 28 three-dimensional linear straight beam elements, 38 three-dimensional linear curved beam elements, and 4 linear expansion joint elements have been used to model the structure as shown in Fig. 16.

For nonlinear analysis, linear elastic straight beam elements Nos. 3-28 as shown in Fig. 16 are replaced by 26 corresponding three-dimensional elasto-plastic flexural column elements and the 4 linear expansion joint elements are replaced by nonlinear elements.

The ultimate column strengths as computed by program YIELD are given in Table 2. The corresponding generalized yield function constants are shown in Table 3.

For nonlinear analysis, three different systems of longitudinal restrainers have been considered for the expansion joints (1) a strong system consisting of 3 short tie bars each having a $2\frac{1}{4}$ in. diameter, a 4ft. length, and a yield strength of 480 kips, (2) a weak system consisting of 3 short tie bars each having a $1\frac{1}{2}$ in. diameter, a 4ft. length, and a yield strength of 70.8 kips, and (3) a system consisting of no ties. The longitudinal tie bars are located across each expansion joint at the centers of the 3 box girder cells which are 9ft. apart. Prior to the San Fernando earthquake, longitudinal restrainer system No. 2 was generally used. However following this earthquake, restrainer system No. 1 has been proposed [32].

In all nonlinear cases studied, the vertical restrainers of the expansion joints were assumed to have infinite yield strengths, the Coulomb coefficients of friction were set at 0.4, the expansion joint gaps Δ_G were taken equal to 1 in., and the tie gaps were assumed equal to zero.

Foundation flexibilities are approximated in all cases by assuming the columns and abutments to be fully fixed at a depth 10ft. below the ground profile.

2. <u>Mode Shapes and Frequencies</u> - Frequencies and corresponding mode shapes were determined for two structural cases using program

BSAP. In the first case, zero friction was assumed for the expansion joints and, in the second case, infinite friction was assumed. The computed frequencies and corresponding periods for the lowest 10 modes of vibration are shown in Table 4 for the zero friction case and in Table 5 for the infinite friction case. The shapes of the 4 lowest modes are plotted in Figs. 17 and 18 for the zero and infinite friction cases, respectively.

3. Spectral Responses - Spectral responses of the linear bridge system subjected to the N-S El Centro 1940 earthquake motions in its transverse direction, i.e., the global Y direction in Fig. 15 were determined for both the zero and infinite friction cases using program BSAP. A damping ratio of 0.02 was used for all modes in these analyses.

Spectral responses for maximum bending moments about the local z axes in columns Nos. 4, 5, and 7 are shown for the lowest 10 modes in tables 6 and 7 for the zero and infinite friction cases, respectively. Root-mean-square values for all 10 modes are also shown in these tables.

4. Nonlinear Earthquake Responses - Nonlinear earthquake response analyses using computer program NEABS were carried out for five structural cases. In all cases, the ground excitation inputs were derived from an artificial earthquake record (OVH accelerogram) generated by Chopra [33] as shown in Fig. 19. This accelerogram was generated to simulate the ground motions produced by the San Fernando earthquake at the site of the Olive View Hospital located about 6 miles southwest of the epicenter. It has a peak acceleration of 0.5g, a uniform phase of high intensity shaking for 8 seconds, and a spectrum intensity of 4.01 ft. which is approximately 1.5 times greater than the N-S El Centro 1940 earthquake.

For all five nonlinear cases studied, rigid ground excitations were assumed and the intensity levels, i.e., peak accelerations, were adjusted to those values shown in Table 8. The structural parameter varied in these five cases was a type of longitudinal restrainer system.

The numerical results obtained for these five cases are based on damping constants $\alpha = 0.0419$ and $\beta = 0.0079$ which correspond to damping ratios of 0.02 in the first two transverse modes of vibration for the zero friction linear case as shown in Fig. 17. In all cases, a time interval of 0.02 seconds and the constant acceleration method have been used in step-by-step integration. Representative time histories of response for these five cases are presented as follows:

Case 1 - The horizontal accelerations (absolute) and the displacements in the global X and Y directions at the top of columns Nos. 3, 4, and 5 are shown in Figs. 20, 21, and 22, respectively. The vertical accelerations and displacements at the center of spans Nos. 3 and 4 are given in Fig. 23. The generalized forces at the center of span No. 3 are shown in Fig. 24. The bending moments and corresponding plastic bending rotations at the bases of columns Nos. 3, 4, 5, and 6 are presented in Figs. 25, 26, 27, and 28, respectively. Longitudinal joint separations at expansion joints Nos. 1 and 2 are shown in Fig. 29. The forces and corresponding plastic elongations for the 3 longitudinal tie bars at expansion joints Nos. 1 and 2 are given in Figs. 30 and 31, respectively. The forces in the vertical restrainers of expansion joints Nos. 1 and 2 are presented in Fig. 32.

Case 2 - The horizontal accelerations and displacements at the top of columns Nos. 3, 4, and 5 are shown in Figs. 33, 34, and 35, respectively. The vertical accelerations and displacements at the centers of spans Nos. 3 and 4 are given in Fig. 36. The generalized forces at the center of span No. 4 are shown in Fig. 37. The bending moments and corresponding plastic bending rotations at the bases of columns Nos. 3, 4, 5, and 6 are presented in Figs. 38, 39, 40, and 41, respectively. The longitudinal joint separations at expansion joints Nos. 1 and 2 are shown in Fig. 42. The longitudinal tie bar forces and corresponding plastic elongations at expansion joints Nos. 1

and 2 are shown in Figs. 43 and 44, respectively. Vertical restrainer forces at expansion joints Nos. 1 and 2 are given in Fig. 45.

Case 3 - This particular bridge system experienced excessively large amplitudes of response. For example, note the large horizontal displacements which occurred at the top of column No. 4 as shown in Fig. 46 and the large joint separations which developed at expansion joint No. 2 as presented in Fig. 47. Due to the fact that very large geometry changes resulted in this case, the numerical results obtained are not valid. They do however strongly indicate that some portions of the overall structure would collapse.

Case 4 - The horizontal accelerations and displacements at the top of column No. 4 are shown in Fig. 48. The vertical accelerations and displacements at the top of this same column and at the center of span No. 4 are presented in Fig. 49. The generalized forces in the box girder at the center of span No. 4 are given in Fig. 50. The bending moments and corresponding plastic bending rotations at the bases of columns No. 3, 4, and 5 are shown in Figs. 51, 52, and 53, respectively. The longitudinal joint separations at expansion joint No. 2 are presented in Fig. 54. The longitudinal tie bar forces and corresponding plastic elongations at expansion joint No. 2 are given in Fig. 55.

Case 5 - The horizontal accelerations and displacements at the top of column No. 4 are given in Fig. 56 while the vertical accelerations and displacements at the top of this same column and at the center of span No. 4 are presented in Fig. 57. The generalized forces in the box girder at the center of span No. 4 are shown in Fig. 58. The bending moments and corresponding plastic bending rotations at the bases of columns Nos. 3, 4, and 5 are presented in Figs. 59, 60, and 61, respectively. The longitudinal joint separations at expansion joint No. 2 as shown in Fig. 62 and the longitudinal tie bar forces and corresponding plastic elongations are shown in Fig. 63.

B. CURVED FIGUEROA STREET UNDERCROSSING CONNECTOR

1. The Structural System - The structural system of this bridge consists of a curved, continuous, reinforced concrete box girder deck of 6 spans (Nos. 1-6 from left to right in Fig. 64) supported on 5 reinforced concrete columns (Nos. 2-6 in Fig. 64) and 2 reinforced concrete diaphragm abutments (Nos. 1 and 7 in Fig. 64). The deck has a radius of curvature of 886 ft. measured to its center line. Only one expansion joint is present (No. 1). All 5 columns have a length of 40ft. and are flared at their upper ends as shown in Fig. 65. These columns are rigidly connected at their bases to massive reinforced concrete footings (25' x 25' x 6') which are in turn supported on 36 piles each. Therefore, the foundation stiffnesses are sufficiently high so that the columns can be assumed as rigidly fixed at their bases. The box girders are rigidly connected to the diaphragm abutments which are relatively short (6 ft.). These abutments are supported on spread footings which provide high shear resistances but low bending resistances. The structural properties of this bridge system are summarized in Table 9.

The discrete parameter system used to model this bridge is shown in Fig. 66. It contains 62 nodal points of which 5 are located at the fixed bases of columns and 2 are located at partially fixed bases which provide full constraint in the three translational directions and in the rotational direction about a longitudinal axis parallel to the deck; however, no significant degree of fixity is provided in the other two rotational directions. Considering these support constraints, the overall system has 274 degrees of freedom.

For linear analysis, a finite element system consisting of 27 three-dimensional straight beam elements, 23 three-dimensional curved beam elements, and 1 linear expansion joint element as shown in Fig. 67 was used. Each column was modelled by 5 straight beam elements of which 3 were used to model the upper flared section and 2 were used to model the prismatic portion. For nonlinear analysis, the 2 straight beam elements of the prismatic section were replaced by 2 corresponding elasto-plastic flexural column elements. The ultimate strengths of these elements as calculated by YIELD are shown in Table 10. The corresponding generalized yield function constants are given in Table 11.

The expansion joint of this bridge uses a longitudinal restrainer system consisting of 6 cable units, placed across the joint at distances of ± 8.92, ± 5.85, ± 2.92 ft. from the center line of the deck. Each cable unit consists of 5 - $1\frac{3}{4}$ in. diameter cables, 127.5 ft. long, having yield strengths of 154.6 kips each. The expansion joint is constructed with an initial joint gap Δ_G = 6 in. but with all tie gaps Δ_T equal to zero. The vertical restrainers at the expansion joint are assumed to have infinite yield strengths. The Coulomb friction coefficient is again assumed to be 0.4. The expansion joint width d is taken as 27ft.

The longitudinal restrainer system used in this bridge is a new version recently developed by the California State Division of Highways [32].

2. Mode Shapes and Frequencies - The mode shapes and frequencies of this bridge system have been determined only for the case of zero friction in the expansion joint. The computed frequencies and corresponding periods for the lowest 10 modes are listed in Table 12. The shapes of the lowest 4 modes are plotted in Fig. 68.

3. Nonlinear Earthquake Responses - The nonlinear response of this bridge system has been determined using NEABS for three different cases. In all three cases, rigid ground excitations corresponding to the OVH accelerogram were used as shown in Fig. 19. The structural parameter varied in these studies were the yield strengths of the columns. Both elastic and elasto-plastic flexural columns were considered. Table 13 describes the column types used in each case and also shows the intensities of excitations which were prescribed.

Numerical results have been obtained for all three cases using damping constants $\alpha = 0.09$ and $\beta = 0.0012$ which correspond to a damping ratio of 0.02 for the lowest 2 modes of vibration for the zero friction case. In all 3 cases, a time interval of 0.02 seconds has been used in step-by-step integration. Selected time history responses are presented as follows:

> Case 1 - The horizontal accelerations and displacements at the top of column No. 3 and 4 are presented in Figs. 69 and 70, respectively. The vertical accelerations and displacements at the top of column No. 3 and at the center of span No. 3 are shown in Fig. 71. The generalized forces in the box girder at the center of span No. 3 are given in Fig. 72 while the elastic bending moments at the bases of columns Nos. 3 and 4 are presented in Fig. 73. The longitudinal joint separations are shown in Fig. 74 and the longitudinal cable forces at the joint are given in Fig. 75. These cable forces remain elastic during the entire response history.
>
> Case 2 - The horizontal accelerations and displacements at the top of columns Nos. 3 and 4 are shown in Figs. 76 and 77, respectively. The vertical accelerations and displacements at the top of column No. 3 and at the center of span No. 3 are given in Fig. 78. The generalized forces in the box girder at the center of span No. 3 are presented in Fig. 79 while the bending moments and corresponding plastic rotations at the bases of columns Nos. 3 and 4 are shown in Figs. 80 and 81, respectively. The longitudinal joint separations are given in Fig. 82 and the longitudinal cable forces are presented in Fig. 83. These cable forces also remain elastic during the entire response history.
>
> Case 3 - The horizontal accelerations and displacements at the top of column No. 4 are shown in Fig. 84 while the vertical accelerations and displacements at the top of column No. 3 and at the center of span No. 3 are given in Fig. 85. The generalized forces in the box girder at the center of span No. 3 are presented in Fig. 86 while the bending moments and correspond-

ing plastic rotations at the bases of columns Nos. 3 and 4 are shown in Figs. 87 and 88, respectively. The longitudinal joint separations are presented in Fig. 89 and the longitudinal cable forces are given in Fig. 90. It is significant to note that the cables remained elastic throughout the time history of response.

C. STRAIGHT FIGUEROA STREET UNDERCROSSING CONNECTOR

1. The Structural System - The structural system of this bridge is identical to that of the curved Figueroa Street Undercrossing Connector described previously except that the curvature of the deck has been removed. The general layout of the bridge system is shown in Fig. 91 and the discrete parameter system selected for analysis is given in Fig. 92. This discrete parameter system has the same nodal point arrangement as that selected for the curved bridge model. For linear analysis, the structure has been modelled by 50 three-dimensional straight beam elements and 1 linear expansion joint as shown in Fig. 93. For nonlinear analysis, 2 three-dimensional elasto-plastic flexural elements have been used to model the lower 22ft. prismatic section of each column. The ultimate strengths and corresponding yield function constants for these columns are listed in Tables 10 and 11.

2. Mode Shapes and Frequencies - Mode shapes and frequencies of this bridge system have been determined for the zero friction case only. Frequencies and corresponding periods for the lowest 10 modes are listed in Table 14 while the corresponding shapes of the lowest 4 modes are shown in Fig. 94.

3. Nonlinear Earthquake Responses - Nonlinear response analyses for this system have been carried out using NEABS for only one case. The rigid ground excitations used corresponded with the OVH accelerograms. The intensities used were 0.5g in the transverse direction (global Y direction) and 0.3g in the vertical direction (global Z

direction).

The nonlinear analysis was carried out using damping constants $\alpha = 0.09$ and $\beta = 0.0013$ which correspond to a damping ratio of 0.02 for the lowest 2 modes of vibration in the zero friction case. A time interval of 0.02 seconds has also been used. The horizontal accelerations and displacements at the top of columns Nos. 3 and 4 are shown in Figs. 95 and 96, respectively, while the vertical accelerations and displacements at the top of column No. 3 and at the center of span No. 3 are given in Fig. 97. The generalized forces in the box girder at the center of span No. 3 are presented in Fig. 98 and the bending moments and corresponding plastic rotations at the base of columns Nos. 3 and 4 are shown in Figs. 99 and 100, respectively. The longitudinal joint separations are given in Fig. 101 and the longitudinal cable forces are shown in Fig. 102. It should be noted that the cable forces remained elastic throughout the time history of response.

D. SUMMARY OF EXTREME-VALUES OF NONLINEAR RESPONSES

The extreme-values of the responses previously presented for the 5 cases of the 5/14 South Connector Overcrossing, the 3 cases of the curved Figueroa Street Undercrossing Connector, and the single case of the straight Figueroa Street Undercrossing Connector are summarized as follows:

1. Maximum Accelerations and Displacements - The maximum horizontal accelerations (absolute) and displacements at the top of columns No. 2-7 are summarized in Tables 15 and 16, respectively, for the 5/14 South Connector Overcrossing. The maximum values shown are the vectorial sum of the horizontal components measured along the global X and Y axes. Likewise, the maximum horizontal accelerations and displacements at the top of each column for the 3 cases of the curved and the single case of the straight Figueroa Street Undercrossing Connector are shown in Tables 17 and 18, respectively.

The maximum vertical accelerations and displacements at the center of spans are shown in Tables 19 and 20, respectively, for the 5/14 South Connector Overcrossing and are shown in Tables 21 and 22, respectively, for the curved and straight Figueroa Street Undercrossing Connector.

2. Locations of Maximum Flexural Yielding in Columns - The locations of maximum flexural yielding in the columns of the 5/14 South Connector Overcrossing are shown in Fig. 103 for all 5 cases investigated. Similar locations for Cases 2 and 3 of curved Figueroa Street Undercrossing Connector and the single case of the straight Figueroa Street Undercrossing Connector are shown in Fig. 104. It should be noted that the hinge locations in these figures represent locations of maximum flexural yielding occurring during the first 10 seconds of response.

3. Maximum Local Bending Ductility Factors at the Bases of Columns - The maximum local bending ductility factor μ at a yield hinge of a column is defined as the ratio of the maximum flexural rotation to the yield rotation θ^Y, i.e.

$$\mu = \frac{\theta^P + \theta^Y}{\theta^Y} \tag{113}$$

where θ^P is the plastic component of flexural rotation. For a column of finite dimension, yielding will not occur in concentrated hinges; rather, it will occur over finite lengths h which are approximately equal to the transverse dimensions of the column cross-section in the directions normal to the axes of rotation [20]. Therefore, the flexural yield rotation θ^Y can be estimated by the relation

$$\theta^Y = \frac{M_u}{EI} h \tag{114}$$

where M_u is the ultimate bending moment which in the present analysis is assumed equal to the yield bending moment.

The flexural yield rotations θ_y^Y and θ_z^Y about the local y and z axes of the columns of the 5/14 South Connector Overcrossing have been computed using Eq. (114) and the results are listed in Table 23. Similar yield rotations for the columns of the Figueroa Street Undercrossing Connector are shown in Table 24.

The maximum local bending ductility factors at the bases of the columns of the 5/14 South Connector Overcrossing are computed using Eq. (113) for Cases 1, 2, 4, and 5. The results are shown in Table 25. Similar ductility factors for Cases 2 and 3 of the curved Figueroa Street Undercrossing Connector and the single case of the straight Figueroa Street Undercrossing Connector are given in Table 26. The values presented in Tables 25 and 26 are absolute maxima computed for the bending rotations about the local y and z axes.

4. <u>Maximum Longitudinal Joint Separations and Longitudinal Restrainer Tie Ductility Factors</u> - The maximum longitudinal expansion joint separations at the center lines of the bridge decks are shown in Table 27 for expansion joints Nos. 1-4 in Cases 1, 2, 4, and 5 of the 5/14 South Connector Overcrossing. Similar joint separations are shown in Table 28 for the single expansion joint of the Figueroa Street Undercrossing Connector (3 cases for the curved undercrossing and 1 case for the straight crossing).

The longitudinal restrainer tie ductility factor μ_T is defined as the ratio of the maximum tie elongation to its yield elongation, i.e.

$$\mu_T = \frac{u_T^E + u_T^P}{u_T^E} \tag{115}$$

where u_T^P and u_T^E are defined by Eq. (53) of Chapter IV. Factor μ_T is computed in each case for all longitudinal restrainer ties at the expansion joints. The maximum values of μ_T for each expansion joint are listed in Table 27 for 4 cases of the 5/14 South Connector Overcrossing and in Table 28 for 3 cases of the curved Figueroa Street Undercrossing Connector and for the single case of the straight

Figueroa Street Undercrossing Connector.

Table 1 Structural Properties of 5/14 South Connector Overcrossing

Structural Component	Member Dim. (ftXft)	Young's Modulus (k / ft^2)	Poisson Ratio	Unit Weight (lb/ft^3)	A (ft^2)	I_x (ft^4)	I_y (ft^4)	I_z (ft^4)
Column # 2	10 X 5	490,000	0.20	145.0	45.7	215.0	86.1	327.6
Column # 3.5	10 X 6	490,000	0.20	145.0	53.9	324.0	143.2	379.6
Column # 4	10 X 6	524,000	0.20	145.0	53.9	324.0	143.2	379.6
Column # 6	10 X 4	524,000	0.20	145.0	37.3	126.0	49.6	275.3
Column # 7.8	10 X 4	557,000	0.20	145.0	37.3	126.0	49.6	275.3
Column # 9	10 X 4	490,000	0.20	145.0	37.3	126.0	49.6	275.3
Abutment # 1	31 X 2.5	490,000	0.20	145.0	77.5	161.5	40.4	6200.0
Abutment # 10	34 X 2.5	490,000	0.20	145.0	85.0	177.2	44.3	8190.0
Girder @ Span # 1, 2, 5, 9	34 X 7	490,000	0.20	145.0	56.3	1107.6	479.3	5443.2
Girder @ Span #3, 4, 6, 7,8	34 X 7	524,000	0.20	145.0	60.9	1081.1	488.6	5722.0

Table 2 Ultimate Strength of the Columns of 5/14 South Connector Overcrossing

Column No.	f_c' (psi)	Reinforcement #18 bars	P_o 10^4 (k)	M_{yo} 10^4 (k-ft)	M_{zo} 10^4 (k-ft)
# 2	3,500	40	2.579	1.386	2.537
# 3, 5	3,500	40	2.948	1.671	2.598
# 4	4,000	52	3.470	2.156	3.331
# 6	4,000	40	2.465	1.102	2.517
# 7, 8	4,500	48	2.821	1.320	2.997
# 9	3,500	48	2.365	1.303	2.900

Table 3 Generalized Yield Function Constants of the Columns of 5/14 South Connector Overcrossing

Column No.	a	a_1	a_2	a_3	b	b_1	b_2	b_3
# 2	2.0	-3.10	-3.83	0.273	2.0	-2.97	-4.21	-0.244
# 3	2.0	-3.63	-4.53	0.102	2.0	-3.55	-4.80	-0.245
# 4	2.0	-3.20	-4.09	0.101	2.0	-3.12	-4.34	-0.225
# 6	2.0	-2.98	-3.44	0.546	2.0	-2.78	-4.00	-0.219
# 7, 8	2.0	-2.81	-3.32	0.489	2.0	-2.61	-3.85	-0.238
# 9	2.0	-2.22	-2.79	0.422	2.0	-2.01	-3.24	-0.232

Table 4 Frequencies and Periods of 5/14 South Connector Overcrossing - Zero Friction

Mode No.	Frequency (rad / sec)	Period (sec)
1	1.226	5.124
2	2.243	2.801
3	3.121	2.013
4	5.549	1.132
5	7.929	0.924
6	9.031	0.696
7	9.123	0.689
8	9.507	0.661
9	12.241	0.513
10	12.902	0.487

Table 5 Frequencies and Period of 5/14 South Connector Overcrossing - Infinite Friction

Mode No.	Frequency (rad / sec)	Period (sec)
1	4.500	1.391
2	6.209	1.012
3	9.109	0.690
4	9.585	0.655
5	11.301	0.556
6	13.487	0.466
7	14.262	0.441
8	14.668	0.428
9	15.123	0.415
10	16.013	0.392

Table 6 Spectral Responses of 5/14 South Connector Overcrossing to the N-S El Centro 1940 Earthquake —Zero Friction Case

Mode No.	Maximum Bending Moments in Columns * About the Local z Axis (k-ft)		
	Column # 4	Column # 5	Column # 7
1	22.5	102.7	1.5
2	29,311.0	31,334.0	2,115.4
3	2.1	398.7	30.4
4	1,635.1	7,828.6	2,455.0
5	639.7	2,359.4	17,458.0
6	902.3	5,519.3	12,080.0
7	40.0	18.5	16.9
8	747.9	762.2	82.9
9	437.8	697.4	1,664.9
10	1,841.6	791.3	94.2
R.M.S.	29,448.0	32,879.0	21,540.0

* Maximum moments shown all occur at the base of each column

Table 7 Spectral Responses of 5/14 South Connector Overcrossing to the N-S El Centro 1940 Earthquake — Infinite Friction Case

Mode No.	Maximum Bending Moments in Columns ** about the Local z Axis (k-ft)		
	Column # 4	Column # 5	Column # 7
1	554.2	480.7 *	884.6
2	11,573.0	10,141.0	1,928.9 *
3	41.8 *	446.9 *	237.5
4	2,136.2	4,358.1 *	12,570.0
5	5.2	5.8	4.5
6	23.6	95.5	20.9 *
7	10.0	1.4	164.9 *
8	237.1	12.8	745.4
9	100.5	79.8 *	293.3
10	61.8	38.9	222.5
R.M.S.	11,785.0	10,574.0	12,684.0

* Maximum moment occurs at the top of the column.

** All valves shown are maxima at the base of the columns, except those marked by *.

Table 8 Parameter Studies of 5/14 South Connector Overcrossing

Parameter Case No.	Ground Excitation (OVH Ground Acceleration)	Longitudinal Restrainers at Expansion Joints
Case 1	Severe Shaking 0.5g Transverse(Y) 0.3g Vertical (Z)	3 Strong Short Ties *
Case 2	Severe Shaking 0.5g Transverse(Y) 0.3g Vertical (Z)	3 Weak Short Ties **
Case 3	Severe Shaking 0.5g Transverse(Y) 0.3g Vertical (Z)	No Ties
Case 4	Moderate Shaking 0.3g Transverse(Y) 0.18g Vertical (Z)	3 Weak Short Ties
Case 5	Severe Shaking 0.5g Transverse(Y)	3 Strong Short Ties

* Each strong short tie is $2\frac{1}{4}$" diameter steel bar at 4ft. long, S_y=480 kips.
** Each weak short tie is $1\frac{1}{2}$" diameter steel bar at 4ft. long, S_y=70.8 kips.

Table 9 Structural Properties of Figueroa Street Undercrossing Connector

Structural Component	Member Dim. (ft×ft)	Young's Modulus (k/ft^2)	Poisson's Ration	Unit Weight (lb/ft^3)	A (ft^2)	I_x (ft^4)	I_y (ft^4)	I_z (ft^4)
Column #2 *	20.0×7.5 7.5×7.5	432,000	0.18	145.0	114.2 45.6	1,728.0 295.0	758.6 163.5	1,436.0 163.5
Column #3,4*	20.0×7.5 7.5×7.5	432,000	0.18	145.0	114.2 45.6	1,728.0 295.0	758.6 163.5	1,436.0 163.5
Column #5 *	20.0×7.5 7.5×7.5	432,000	0.18	145.0	114.2 45.6	1,728.0 295.0	758.6 163.5	1,436.0 163.5
Column #6 *	20.0×7.5 7.5×7.5	432,000	0.18	145.0	114.2 45.6	1,728.0 295.0	758.6 163.5	1,436.0 163.5
Girder @ All Spans	40.5×7.5	432,000	0.18	145.0	81.2	1,340.0	587.4	9,606.0

* Also see Fig. 65.

Table 10 Ultimate Strength of the Columns of Figueroa Street Undercrossing Connector

Column No.	f_c' (lb/in^2)	Long. Bars #11	P_o 10^4 (k)	M_{yo} 10^4 (k-ft)	M_{zo} 10^4 (k-ft)
# 2	3,250	96	2.698	2.456	2.456
# 3,4	3,250	87	2.614	2.253	2.253
# 5	3,250	75	2.502	1.978	1.978
# 6	3,250	100	3.189	3.555	3.555

Table 11 Yield Function Constants of the Columns of Figueroa Street Undercrossing Connector

Column No.	a	a_1	a_2	a_3	b	b_1	b_2	b_3
# 2	2.0	-1.86	-3.29	-0.440	2.0	-1.86	-3.29	-0.440
# 3,4	2.0	-1.82	-2.47	0.352	2.0	-1.82	-2.47	0.352
# 5	2.0	-2.44	-3.89	-0.455	2.0	-2.44	-3.89	-0.455
# 6	2.0	-1.16	-2.47	-0.317	2.0	-1.16	-2.47	-0.317

Table 12 Frequencies and Periods of Figueroa Street Undercrossing Connector — Zero Friction at Expansion Joint

Mode No.	Frequency (rad/sec)	Period (sec)
1	7.072	0.889
2	9.208	0.682
3	10.075	0.624
4	11.866	0.530
5	13.924	0.451
6	15.382	0.409
7	20.080	0.313
8	22.986	0.273
9	24.171	0.260
10	26.921	0.233

Table 13 Parameter Studies of Figueroa Street Undercrossing Connector

Parameters / Case No.	Ground Excitations (OVH Ground Acceleration)		Column Type (Elastic vs. Elasto-Plastic)
Case 1	Severe Shaking	0.5g Transverse 0.3g Vertical	Elastic
Case 2	Severe Shaking	0.5g Transverse 0.3g Vertical	Elasto-Plastic
Case 3	Moderate Shaking	0.3g Transverse 0.18g Vertical	Elasto-Plastic
Straight	Severe Shaking	0.5g Transverse 0.3g Vertical	Elasto-Plastic

Table 14 Frequencies and Periods of Straight Figueroa Street Undercrossing Connector - Zero Friction at Expansion Joint

Mode No.	Frequency (rad/sec)	Period (sec)
1	7.213	0.871
2	9.315	0.675
3	10.098	0.622
4	13.921	0.451
5	13.965	0.450
6	15.367	0.409
7	20.315	0.309
8	23.390	0.269
9	26.481	0.237
10	28.081	0.224

Table 15 Maximum Horizontal Accelerations (Absolute) at the Top of Columns of 5/14 South Connector Overcrossing

Column / Case No.	#2 (g)	#3 (g)	#4 (g)	#5 (g)	#6 (g)	#7 (g)
Case 1	0.93	0.86	0.62	0.87	0.83	0.64
Case 2	0.94	1.96	1.01	0.99	1.81	0.85
Case 4	0.44	0.64	0.36	0.88	0.53	0.40
Case 5	1.01	0.83	0.70	0.85	0.72	0.51

Table 16 Maximum Horizontal Displacements at the Top of Columns of 5/14 South Connector Overcrossing

Col. / Case No.	#2 (ft)	#3 (ft)	#4 (ft)	#5 (ft)	#6 (ft)	#7 (ft)
Case 1	1.21	2.50	2.57	1.96	1.91	1.03
Case 2	1.45	3.32	3.70	3.07	1.30	0.90
Case 4	0.69	1.41	1.69	1.47	0.68	0.48
Case 5	0.78	1.69	1.94	1.80	1.28	0.74

Table 17 Maximum Horizontal Accelerations (Absolute) at the Top of Columns of Figueroa Street Undercrossing Connector

Column / Case No.	#2 (g)	#3 (g)	#4 (g)	#5 (g)	#6 (g)
Case 1	1.01	0.94	1.38	1.06	0.68
Case 2	0.75	0.74	0.85	0.76	0.58
Case 3	0.49	0.51	0.55	0.39	0.44
Straight	0.86	0.84	0.91	0.61	0.72

Table 18 Maximum Horizontal Displacements at the Top of Columns of Figueroa Street Undercrossing Connector

Column / Case No.	#2 (ft)	#3 (ft)	#4 (ft)	#5 (ft)	#6 (ft)
Case 1	0.17	0.32	0.68	0.61	0.31
Case 2	0.13	0.23	0.56	0.40	0.23
Case 3	0.11	0.15	0.33	0.26	0.15
Straight	0.18	0.34	0.91	0.68	0.28

Table 19 Maximum Vertical Accelerations (Absolute) at the Center of Spans of 5/14 South Connector Overcrossing

Span / Case No.	#2 (g)	#3 (g)	#4 (g)	#6 (g)	#7 (g)	#8 (g)
Case 1	0.71	0.59	0.62	0.57	0.73	0.53
Case 2	0.71	0.62	0.59	0.54	0.71	0.44
Case 4	0.41	0.38	0.29	0.30	0.44	0.28
Case 5	0.21	0.15	0.16	0.25	0.16	0.12

Table 20 Maximum Vertical Displacements at the Center of Spans of 5/14 South Connector Overcrossing

Span / Case No.	#2 (ft)	#3 (ft)	#4 (ft)	#6 (ft)	#7 (ft)	#8 (ft)
Case 1	0.091	0.106	0.138	0.047	0.051	0.084
Case 2	0.071	0.118	0.135	0.096	0.073	0.085
Case 4	0.055	0.076	0.085	0.047	0.034	0.059
Case 5	0.055	0.047	0.054	0.042	0.022	0.029

Table 21 Maximum Vertical Accelerations (Absolute) at the Center of Spans of Figueroa Street Undercrossing Connector

Span / Case No.	#2 (g)	#3 (g)	#4 (g)	#5 (g)	#6 (g)
Case 1	0.90	0.77	1.60	1.70	0.95
Case 2	1.00	0.58	1.65	1.65	0.86
Case 3	0.55	0.39	0.97	0.99	0.56
Straight	1.12	0.92	1.30	2.64	0.99

Table 22 Maximum Vertical Displacement at the Center of Spans of Figueroa Street Undercrossing Connector

Span / Case No.	#2 (ft)	#3 (ft)	#4 (ft)	#5 (ft)	#6 (ft)
Case 1	0.046	0.129	0.117	0.262	0.253
Case 2	0.051	0.123	0.161	0.254	0.238
Case 3	0.026	0.077	0.093	0.147	0.152
Straight	0.054	0.155	0.153	0.375	0.241

Table 23 Flexural Yield Rotations of Columns of 5/14 South Connector Overcrossing

Column No.	E (k/ft^2)	I_y (ft^4)	M_{yo} 10^4(k-ft)	h_y (ft)	Θ_y^Y 10^{-3}(rad)	I_z (ft^4)	M_{zo} 10^4(k-ft)	h_z (ft)	Θ_z^Y 10^{-3}(rad)
# 2	490,000	86.0	1386	5.0	1.65	327.6	2537	10.0	1.59
# 3,5	490,000	143.2	1671	6.0	1.43	379.6	2598	10.0	1.40
# 4	524,000	143.2	2156	6.0	1.48	379.6	3331	10.0	1.69
# 6	524,000	49.6	1102	4.0	1.60	275.3	2517	10.0	1.75
# 7,8	557,000	49.6	1320	4.0	1.69	275.3	2997	10.0	1.96

Table 24 Flexural Yield Rotations at the Base of Columns of Figueroa Street Undercrossing Connector

Column No.	E (k/ft^2)	I_y (ft^4)	M_{yo} 10^4(k-ft)	h_y (ft)	Θ_y^Y 10^{-3}(rad)	I_z (ft^4)	M_{zo} 10^4(k-ft)	h_z (ft)	Θ_z^Y 10^{-3}(rad)
# 2	432,000	163.7	2470	7.5	2.17	163.7	2470	7.5	2.17
# 3,4	432,000	163.7	2250	7.5	1.99	163.7	2250	7.5	1.99
# 5	432,000	163.7	1980	7.5	1.75	163.7	1980	7.5	1.75
# 6	432,000	163.7	3560	7.5	3.24	163.7	3560	7.5	3.24

Table 25 Maximum Local Bending Ductility Factors at the Base of Columns of 5/14 South Connector Overcrossing

Base of / Case No.	Column # 2	Column # 3	Column # 4	Column # 5	Column # 6	Column # 7
Case 1	1.3	2.7	1.3	3.2	4.6	3.6
Case 2	4.8	4.0	3.0	7.4	12.2	2.2
Case 4	1.5	1.2	1.0	3.3	2.5	1.8
Case 5	1.6	1.8	1.1	3.3	7.8	2.3

Table 26 Maximum Local Bending Ductility Factors at the Base of Columns of Figueroa Street Undercrossing Connector

Base of / Case No.	Column # 2	Column # 3	Column # 4	Column # 5	Column # 6
Case 1	-*	-	-	-	-
Case 2	1.3	4.0	5.4	4.7	-
Case 3	-	2.4	4.5	3.1	-
Straight	2.3	4.2	22.2	18.0	1.2

* No yielding occurs.

Table 27 Maximum Joint Separations and Longitudinal Restrainer Tie Ductility Factors at the Expansion Joints of 5/14 South Connector Overcrossing

Case No.	Max. Separation (ft)				Tie Ductility Factor			
	EJ #1	EJ #2	EJ #3	EJ #4	EJ #1	EJ #2	EJ #3	EJ #4
Case 1	1.21	0.92	0.40	0.21	78.8	58.4	26.2	13.4
Case 2	1.40	1.63	0.37	0.77	278.0	339.0	94.5	145.1
Case 4	0.79	0.48	0.07	0.34	154.0	101.0	28.3	63.6
Case 5]	0.94	0.63	0.40	0.22	60.4	41.2	25.4	14.4

Table 28 Maximum Joint Separations and Longitudinal Restrainer Tie Ductility Factors at the Expansion Joint of Figueroa Street Undercrossing Connector

Case No.	Joint Separation (ft)	Tie Ductility Factor
Case 1	0.305	-*
Case 2	0.237	-
Case 3	0.133	-
Straight	0.001	-

* No yielding occurs.

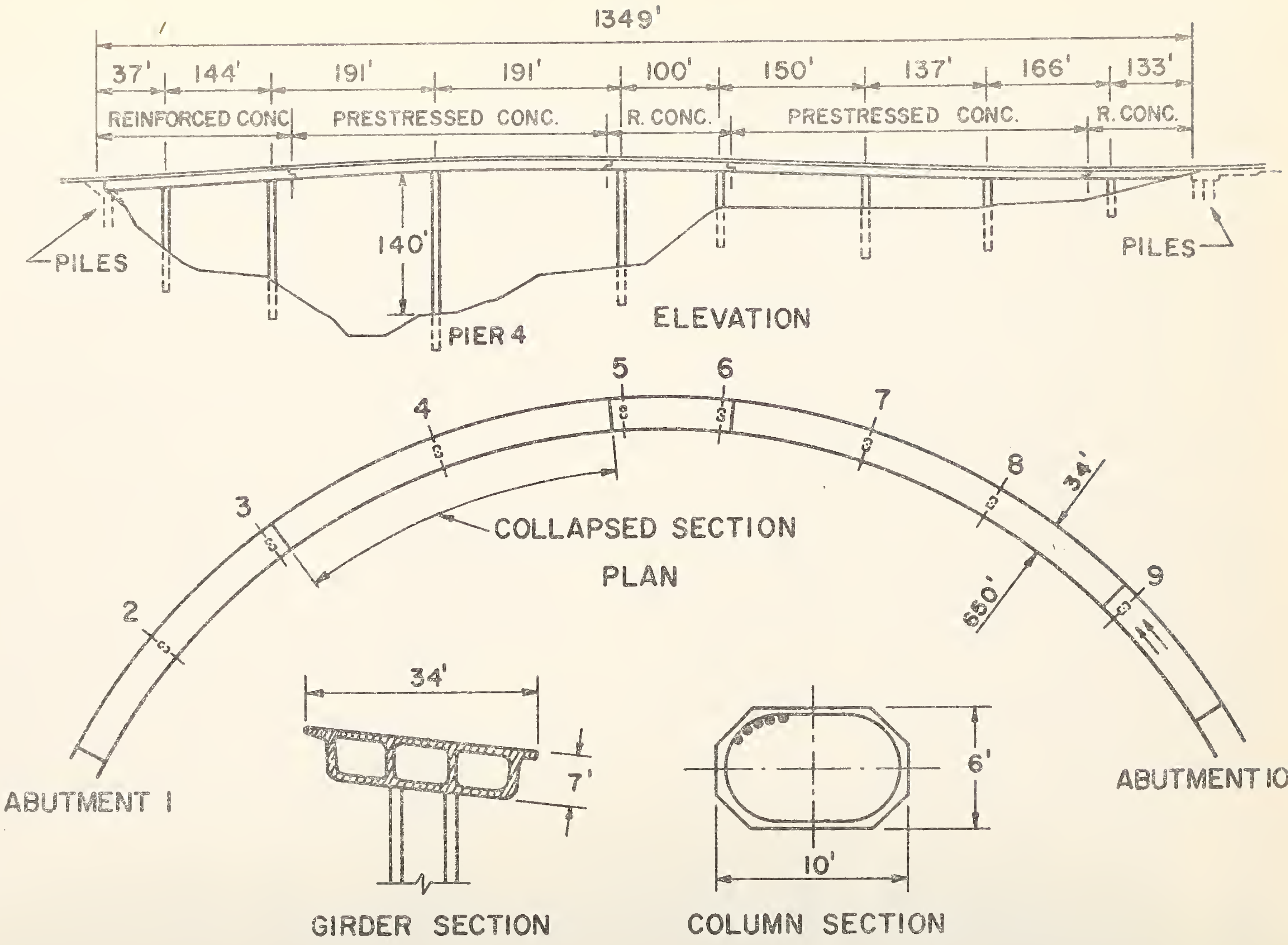


Fig. 14 The Structural System of 5/14 South Connector Overcrossing

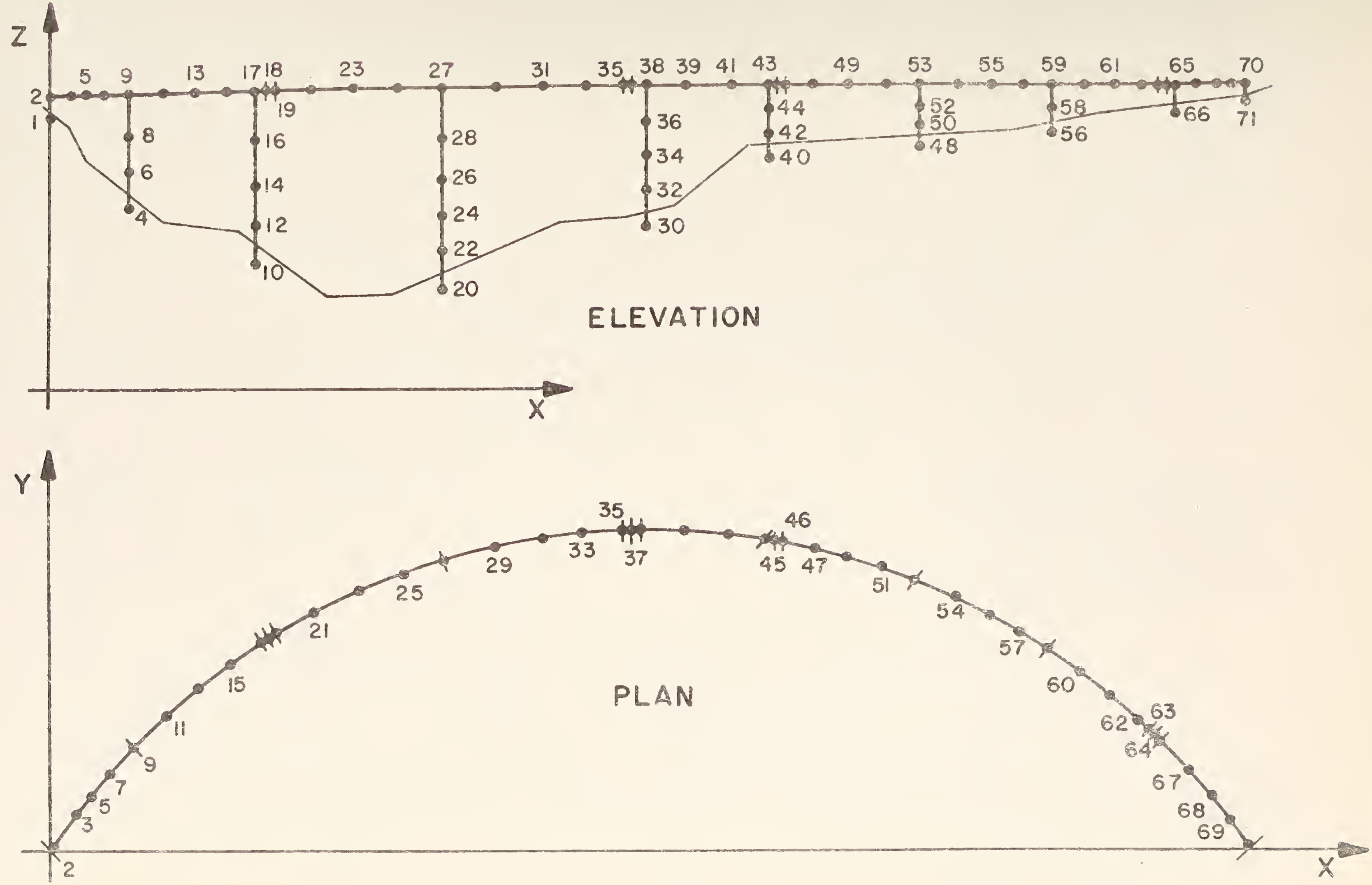


Fig. 15 Lumped Parameter System For 5/14 South Connector Overcrossing

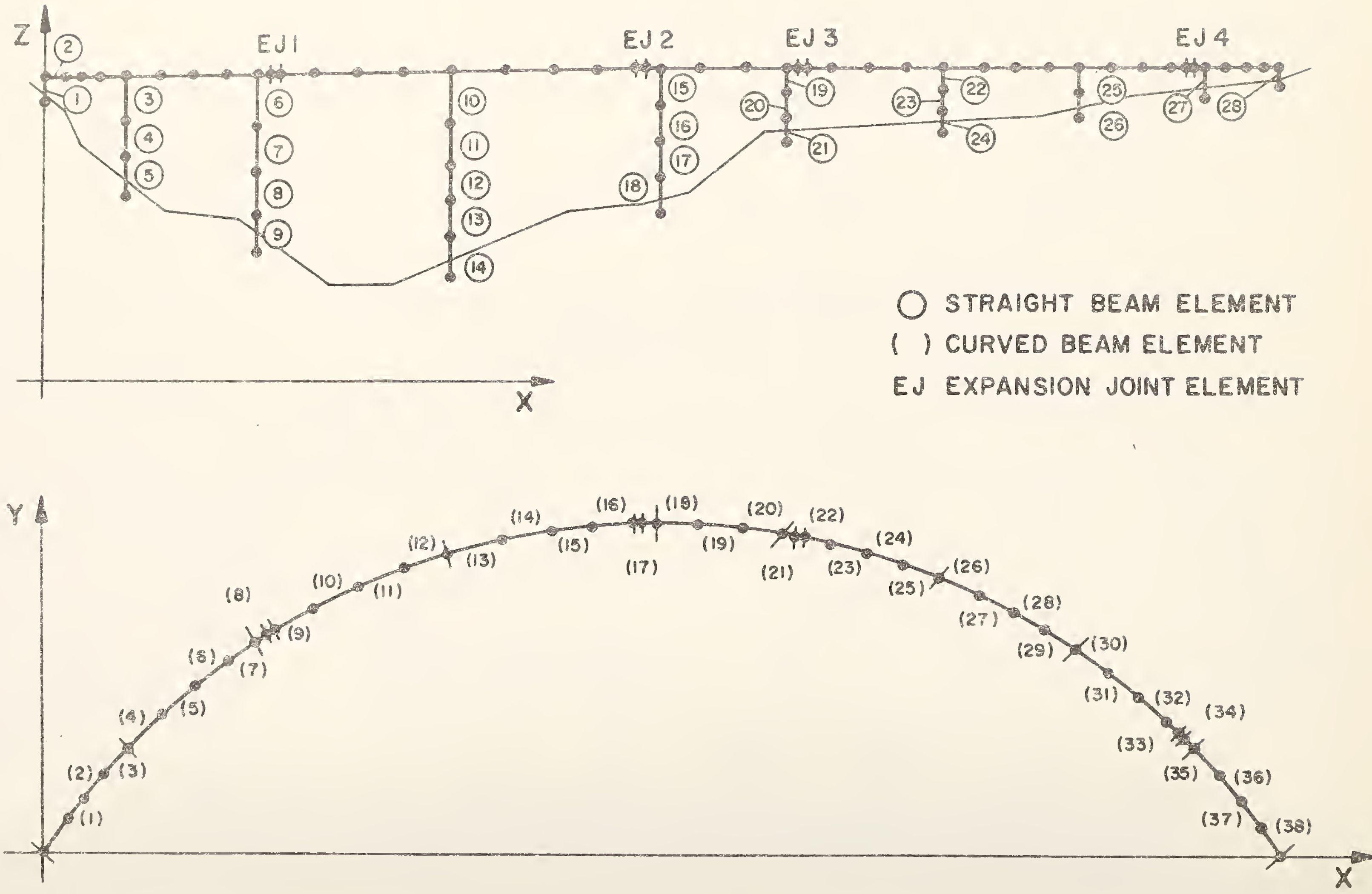


Fig. 16 Finite Element Model of 5/14 South Connector Overcrossing

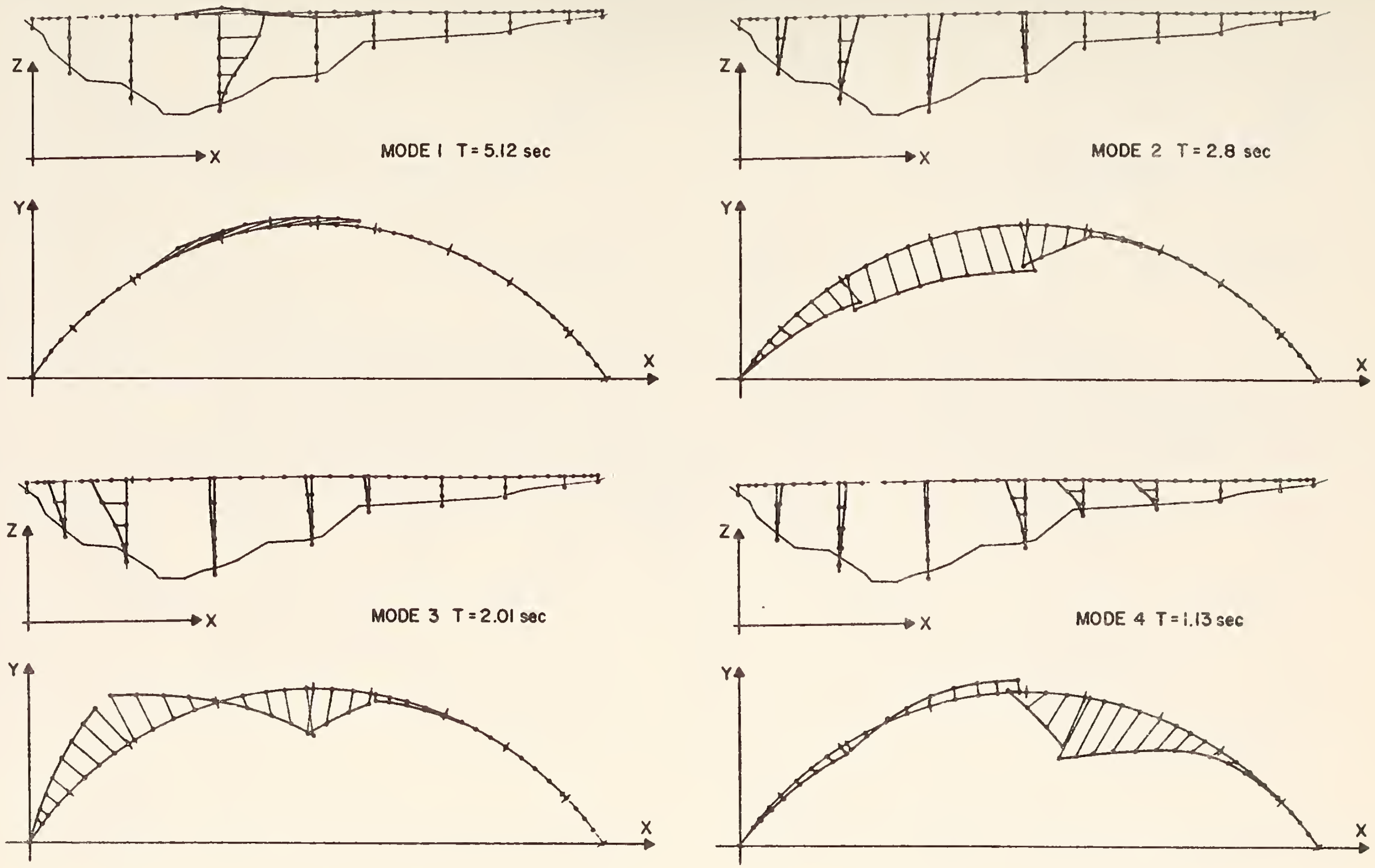


Fig. 17 Mode Shapes of 5/14 South Connector Overcrossing: Zero Friction at Expansion Joints

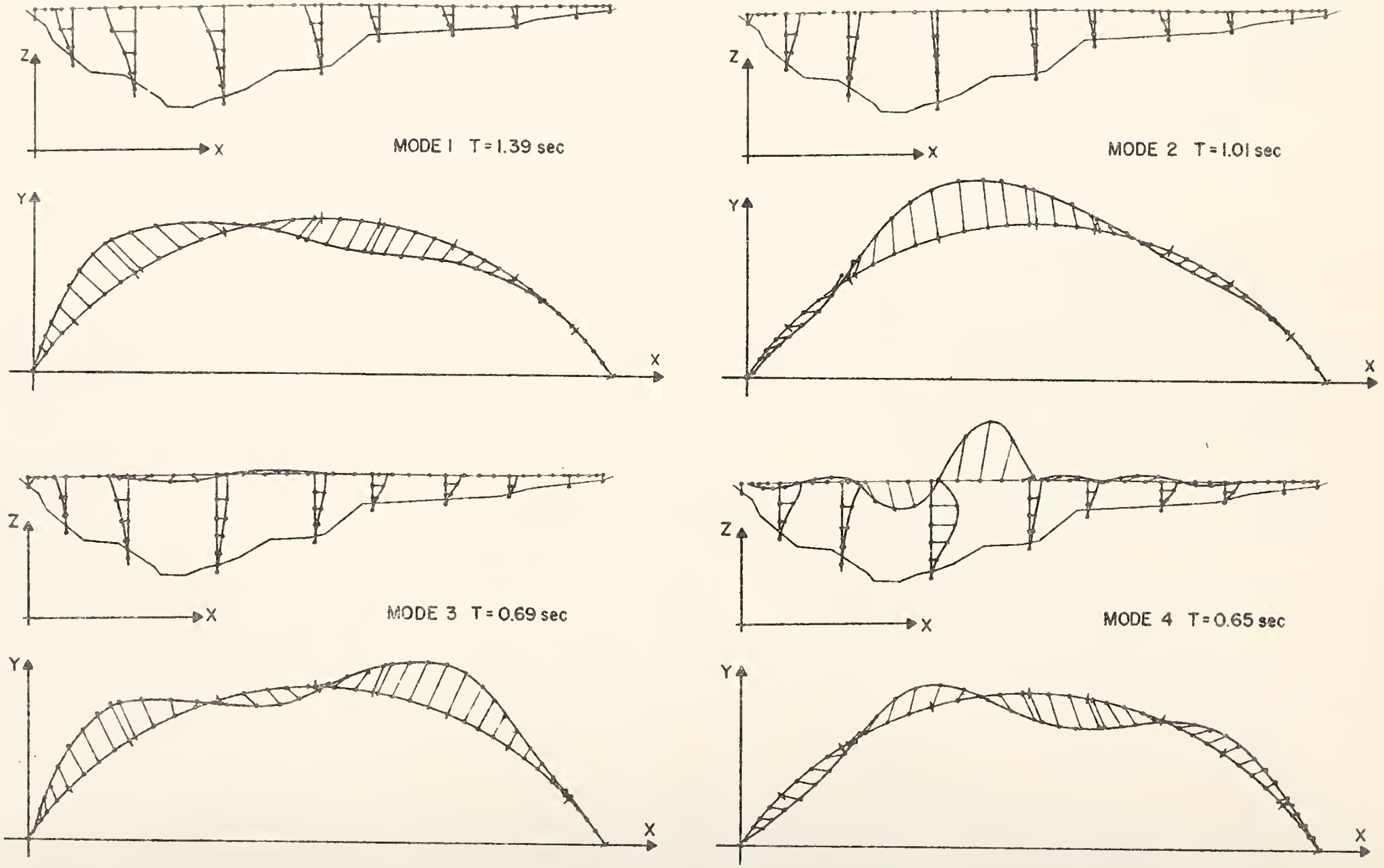


Fig. 18 Mode Shapes of 5/14 South Connector Overcrossing: Infinite Friction at Expansion Joints

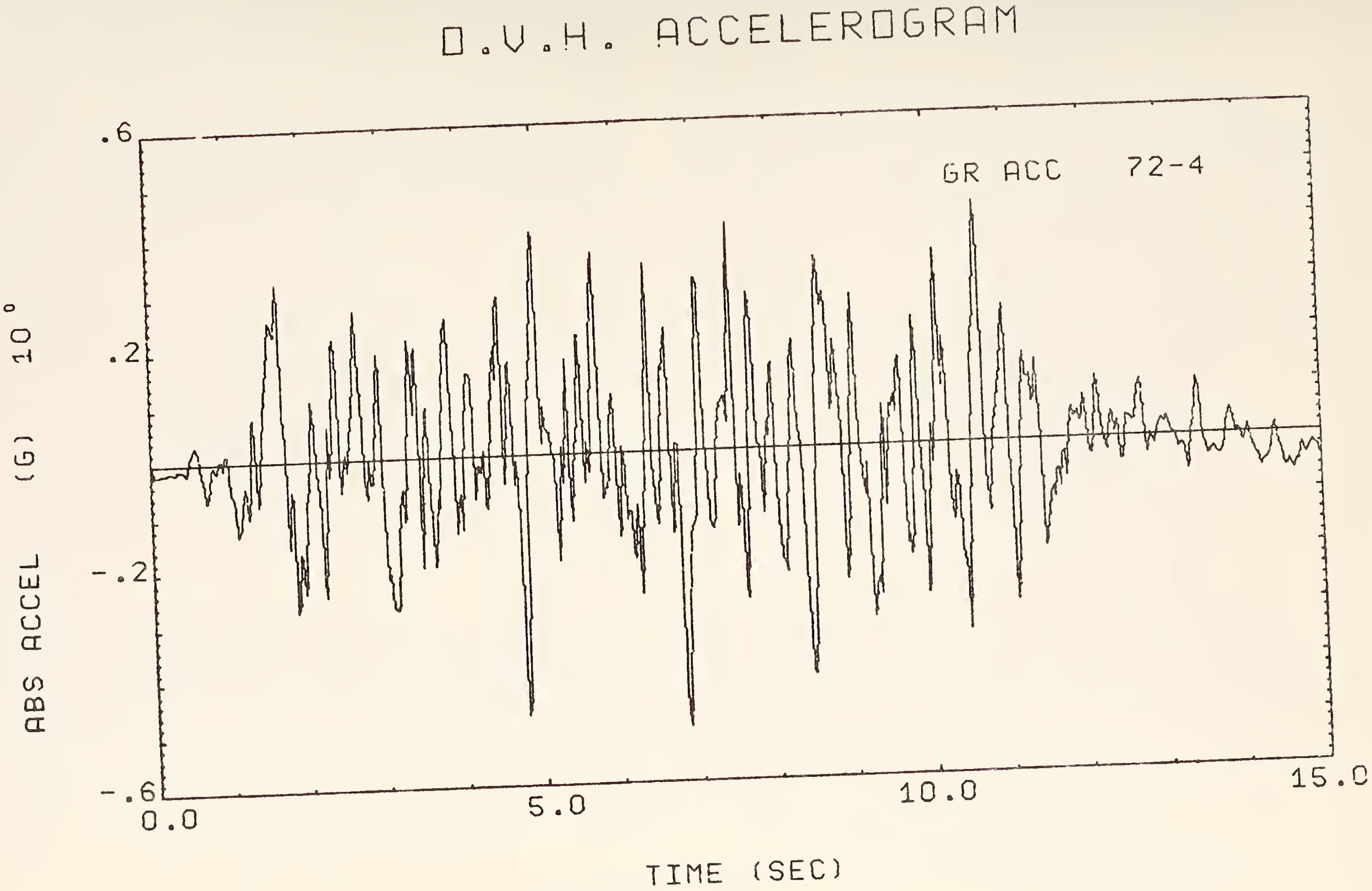
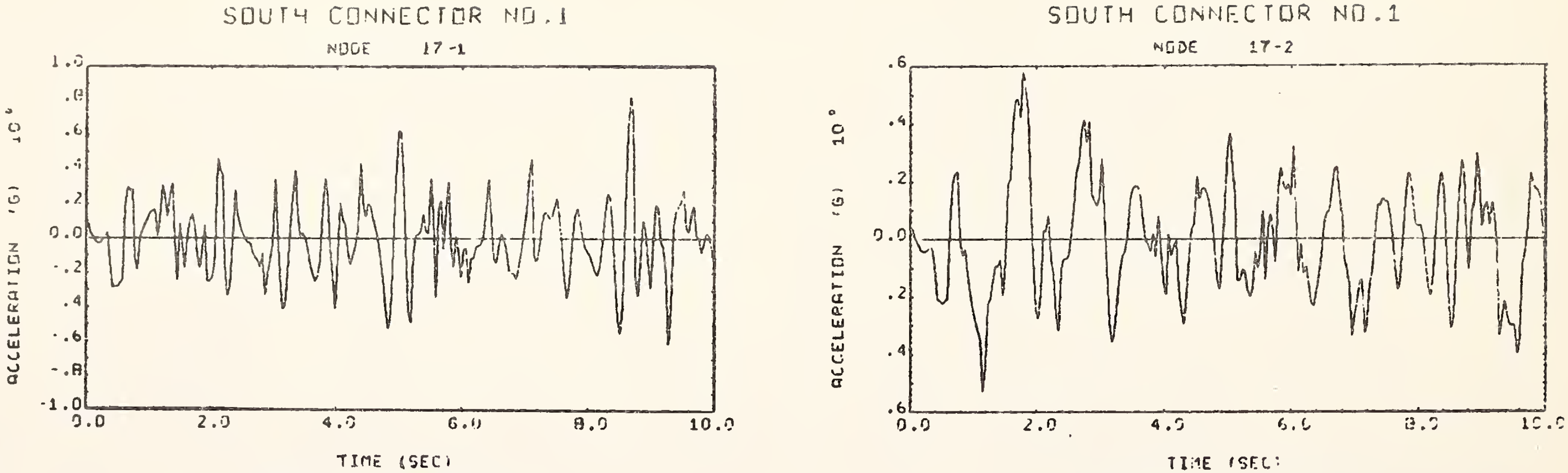


Fig. 19 Simulated Ground Acceleration Record of the San Fernando Earthquake at the Olive View Hospital Site



Global X - Component

Global Y - Component

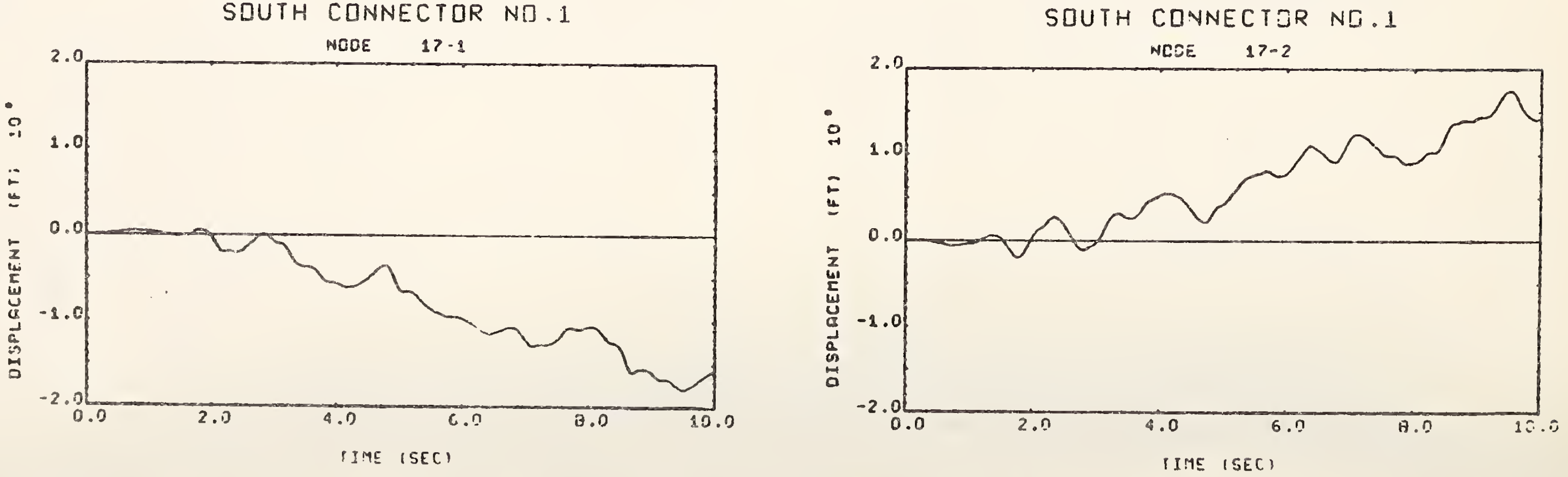
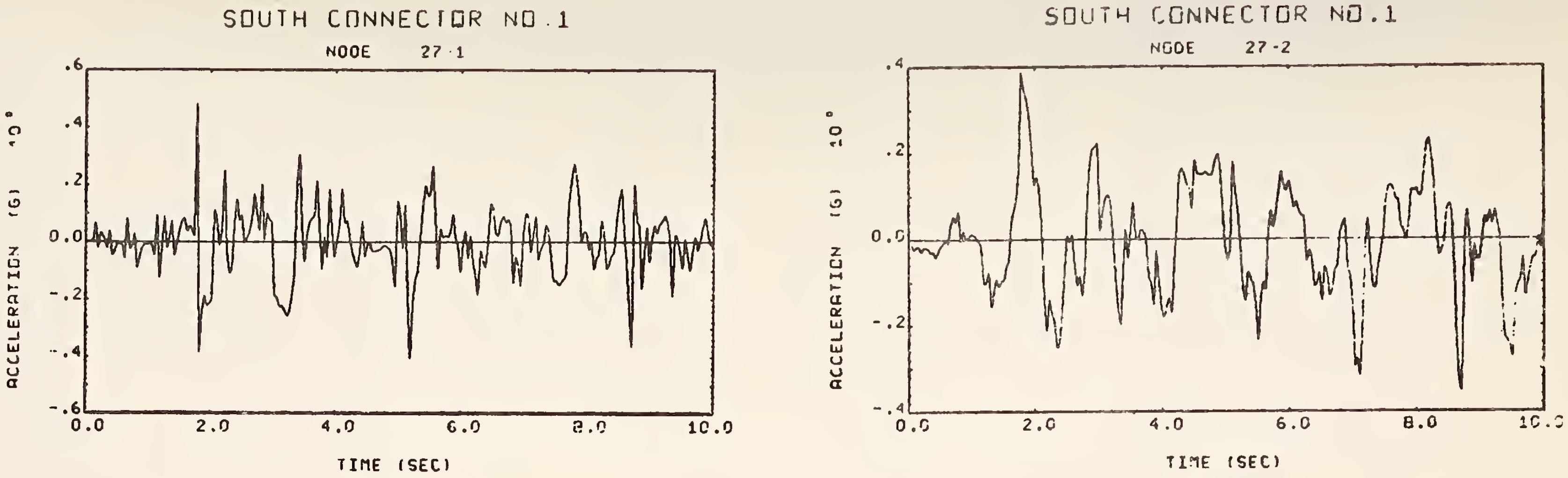


Fig. 20 Horizontal Acceleration and Displacements at the Top of Column # 3



Global X - Component

Global Y - Component

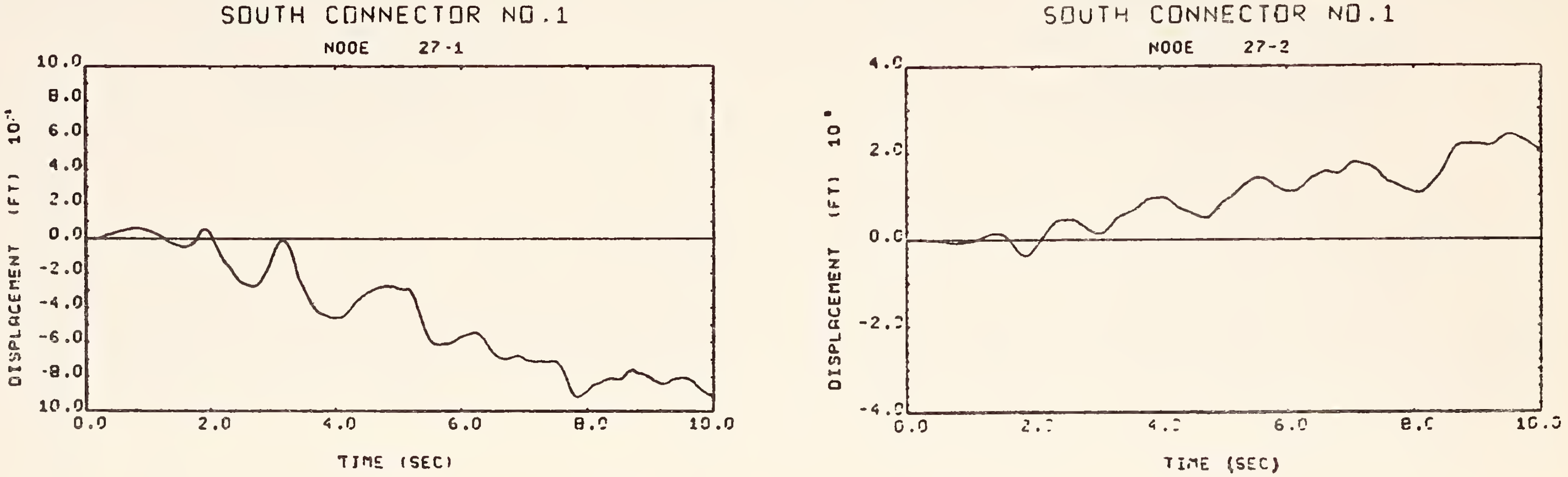
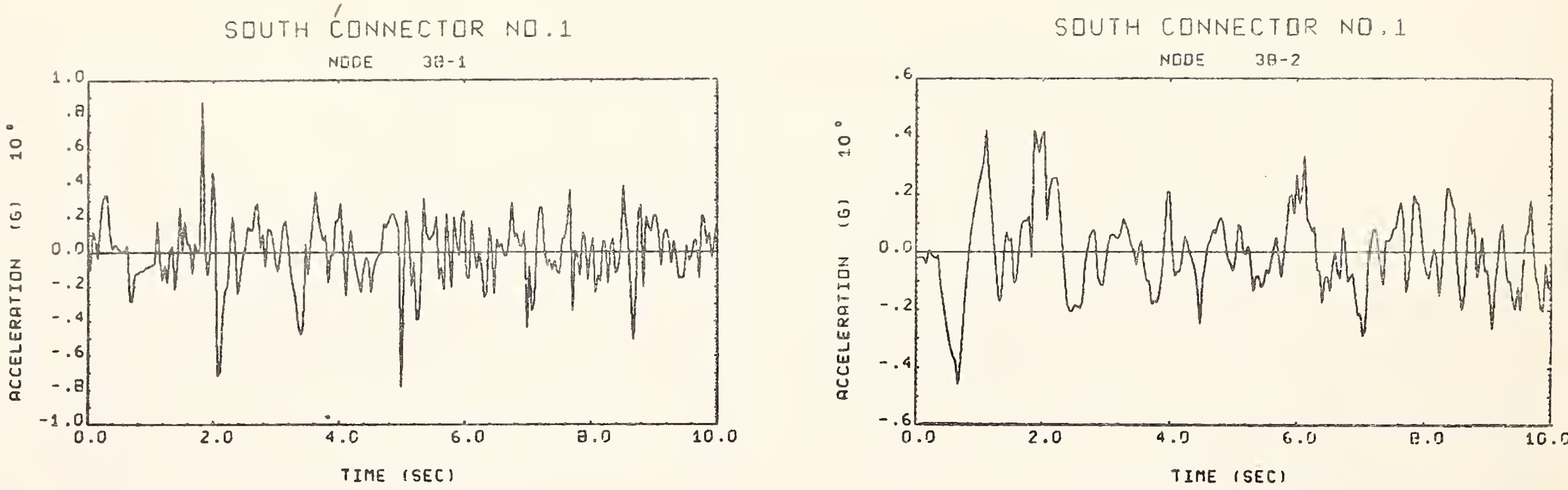


Fig. 21 Horizontal Acceleration and Displacements at the Top of Column # 4



Global X - Component

Global Y - Component

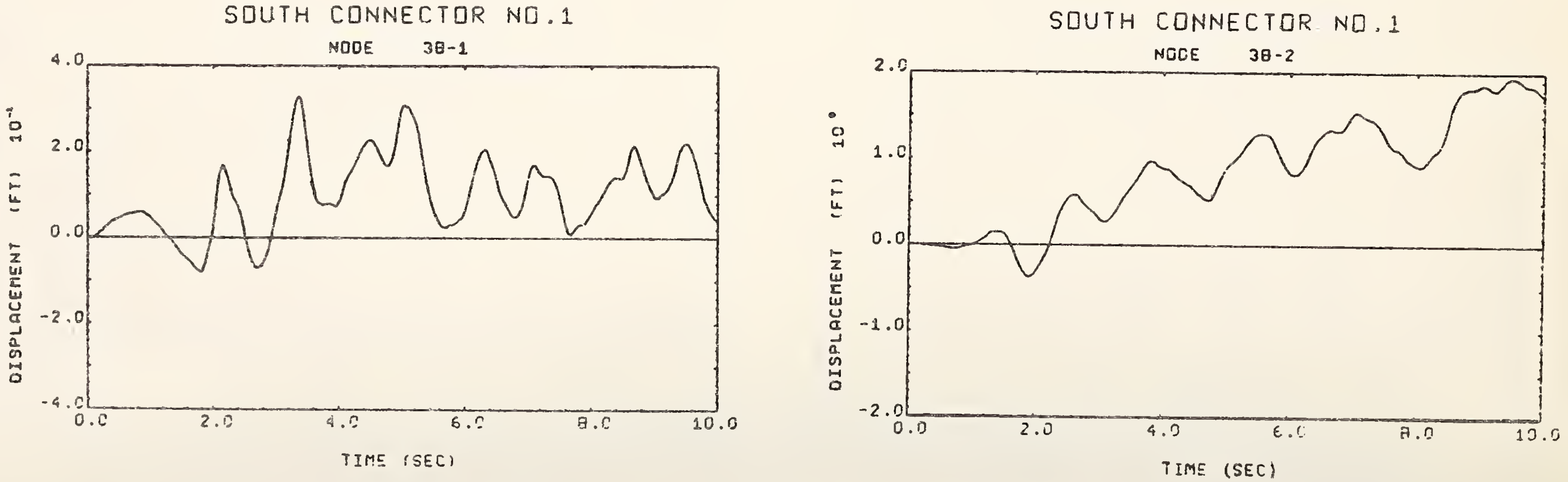
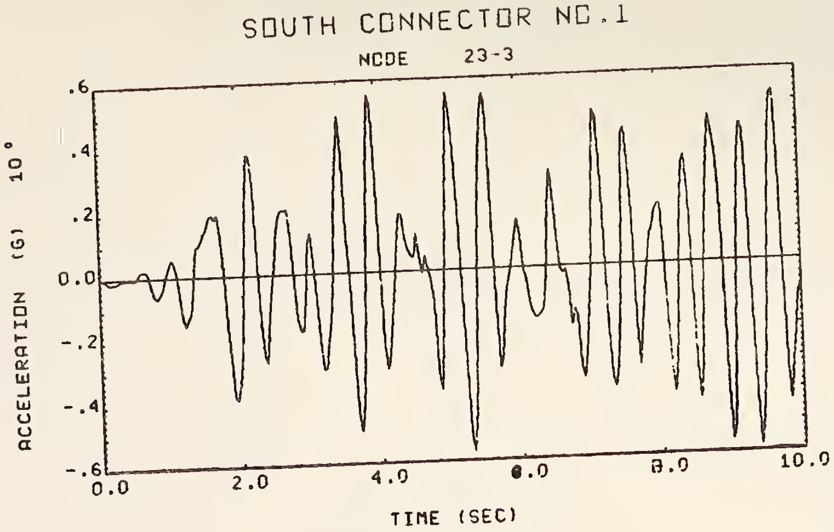
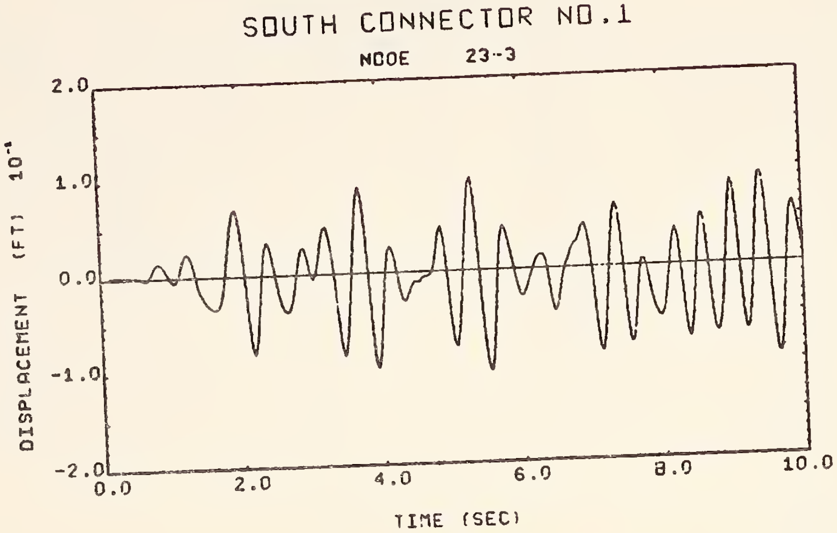
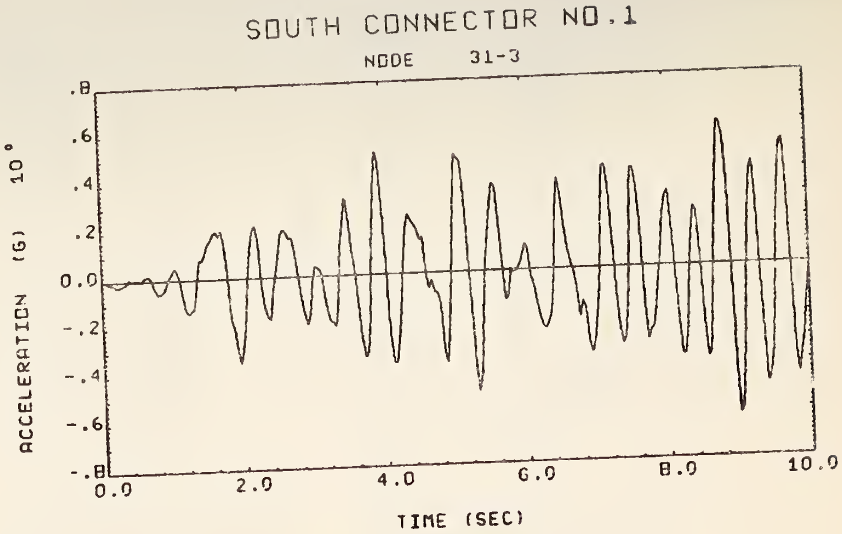


Fig. 22 Horizontal Accelerations and Displacements at the Top of Column # 5



Center of Span # 3





Center of Span # 4

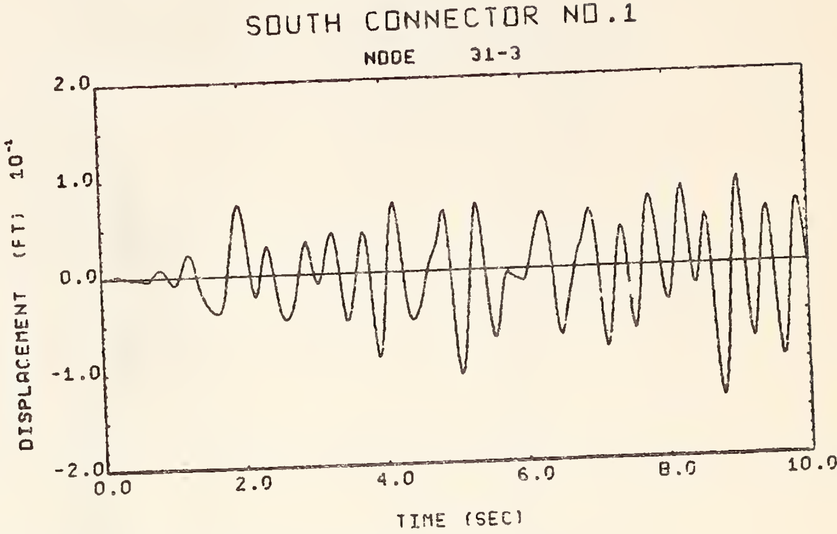
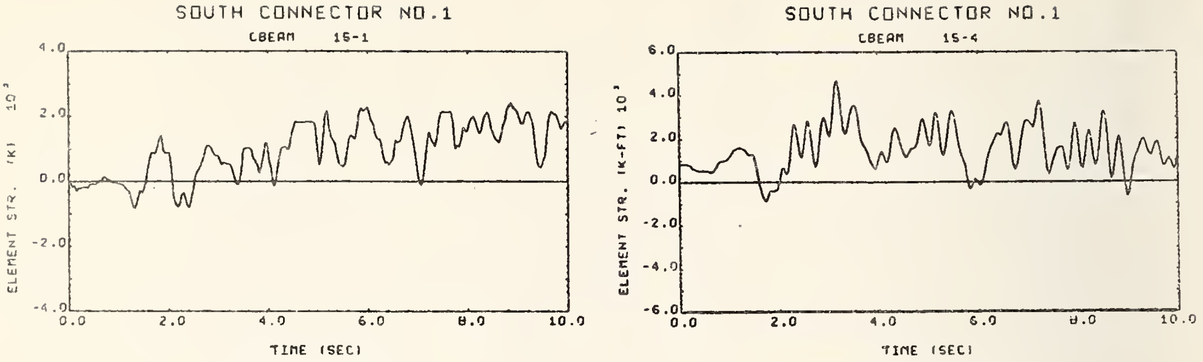
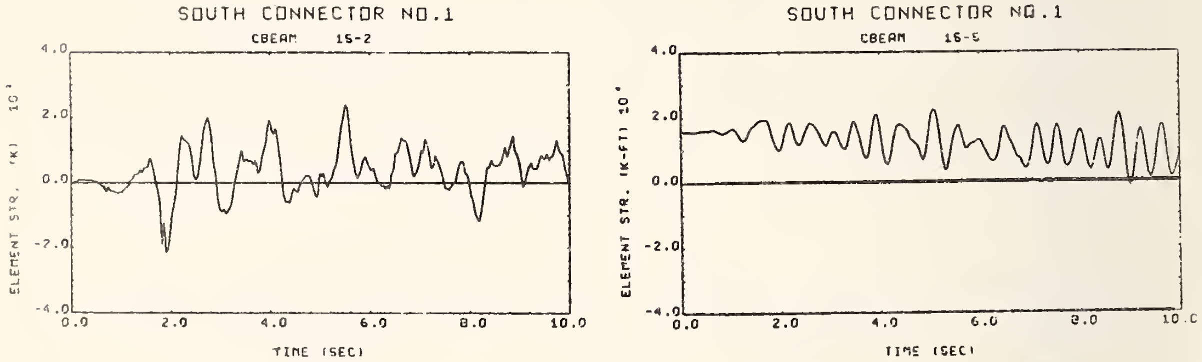


Fig. 23 Vertical Acceleration and Displacements at the Center of Spans # 3 and # 4



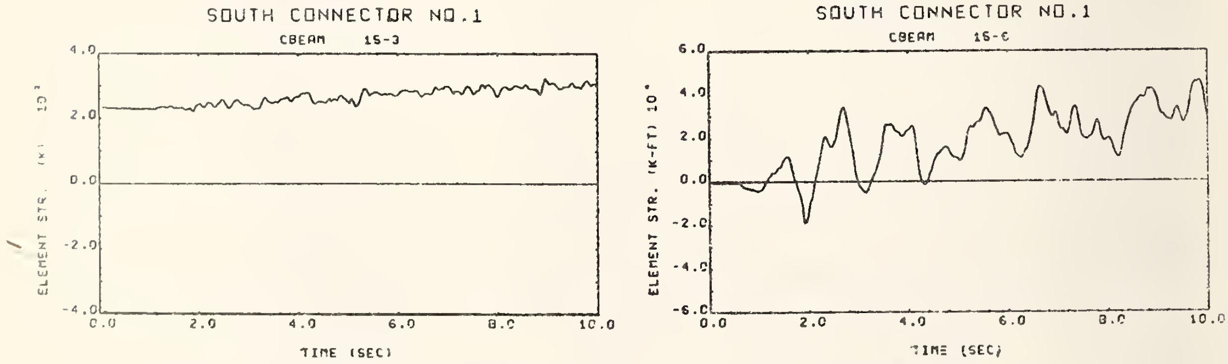
Axial Force

Torsional Moment



Shear in the Local y-Direction

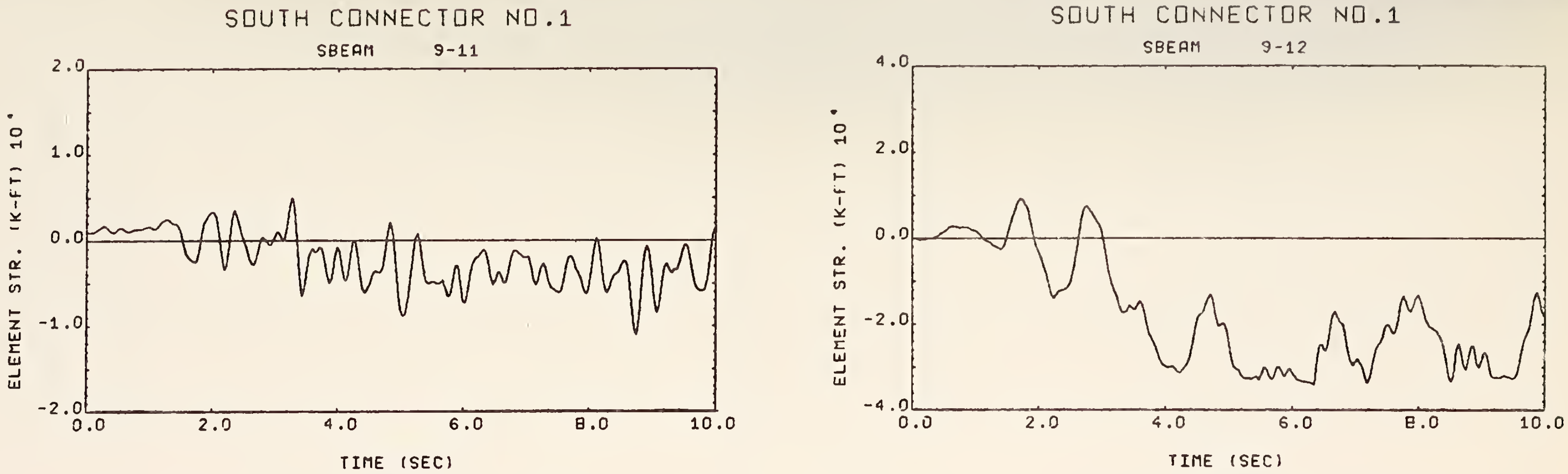
Bending Moment about the Local y Axis



Shear in the Local z-Direction

Bending Moment about the Local z Axis

Fig. 24 Generalized Forces in the Girder at the Center of Span # 4



About the Local y - Axis

About the Local z - Axis

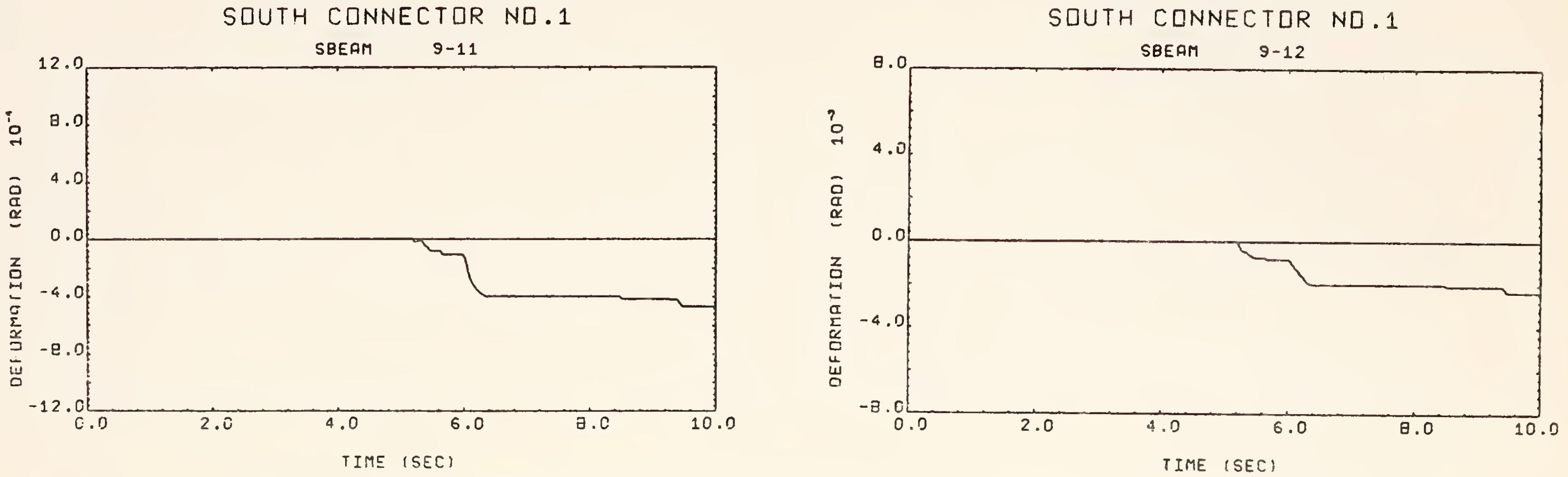
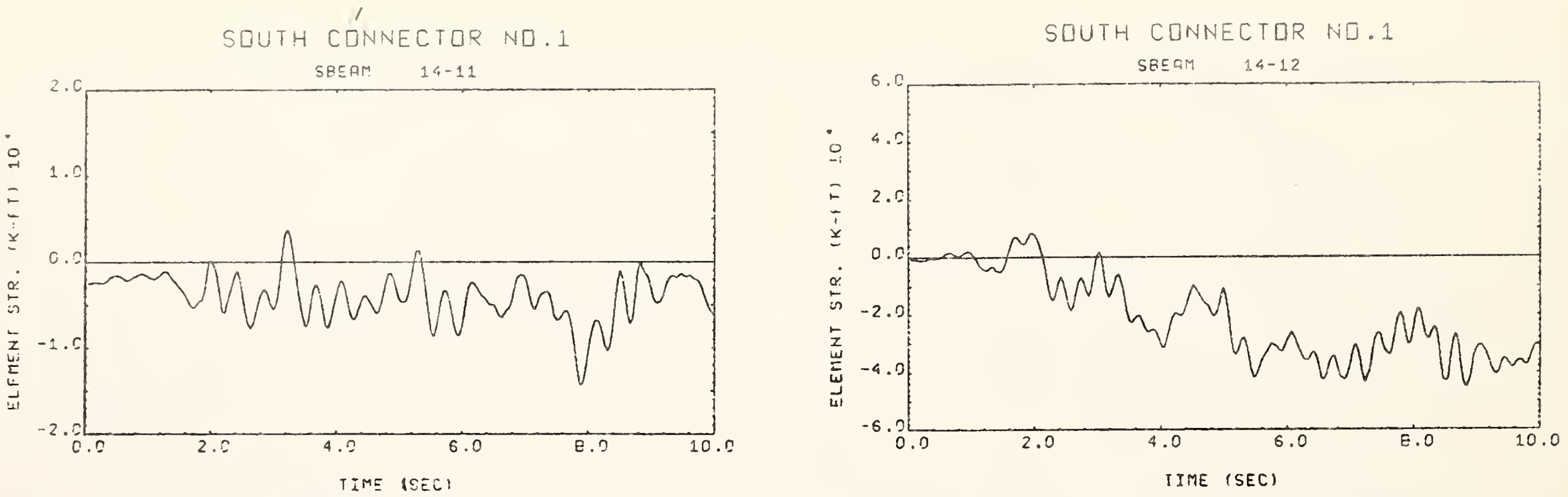


Fig. 25 Bending Moments and the Corresponding Plastic Rotations at the Base of Column # 3



About the Local y - Axis

About the Local z - Axis

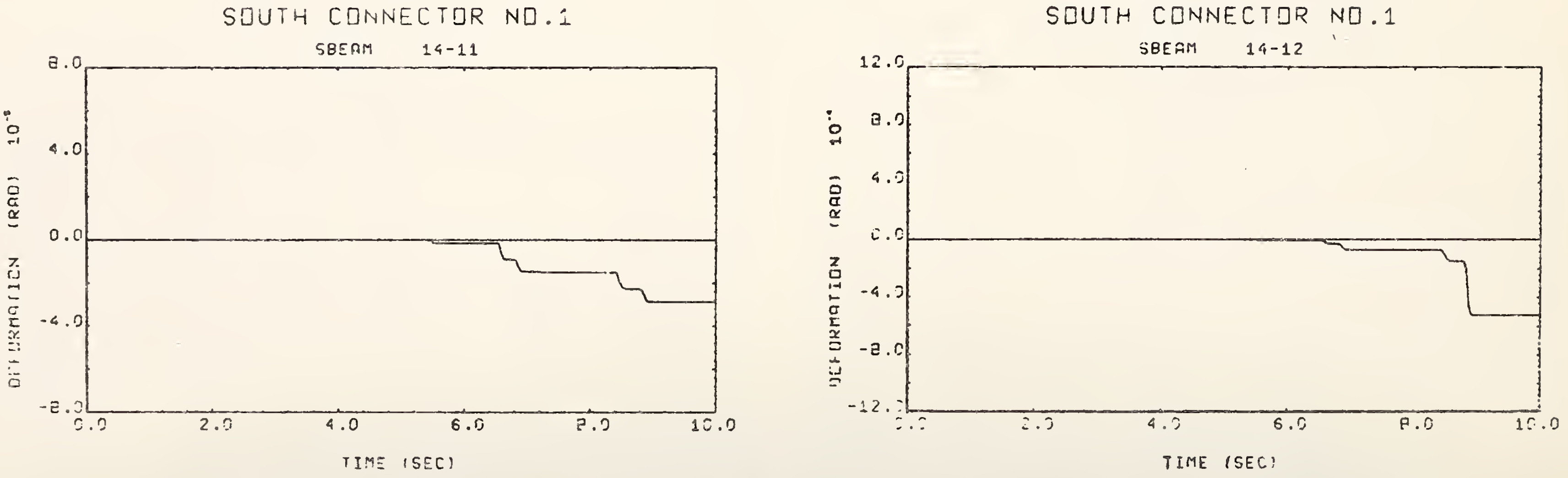
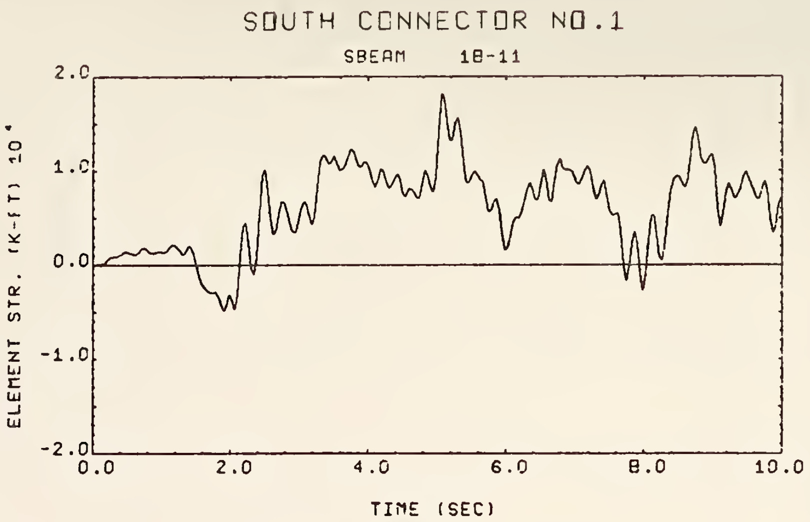


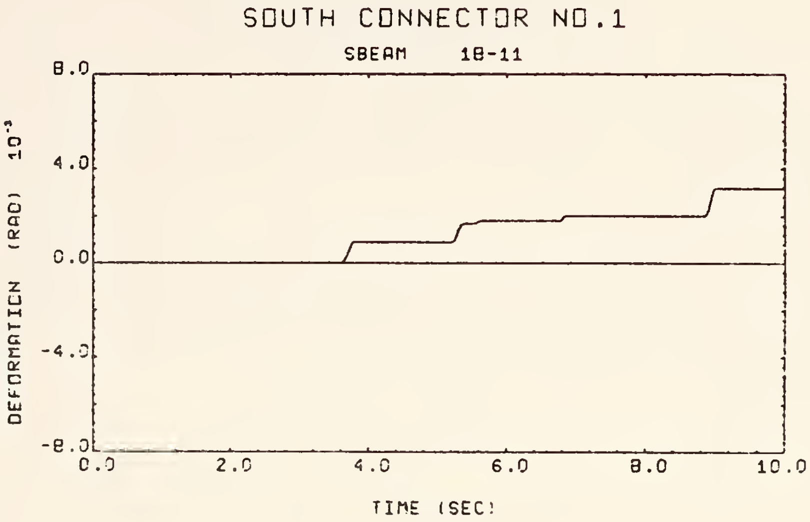
Fig. 26 Bending Moments and the Corresponding Plastic Rotations at the Base of Column # 4



SOUTH CONNECTOR NO.1

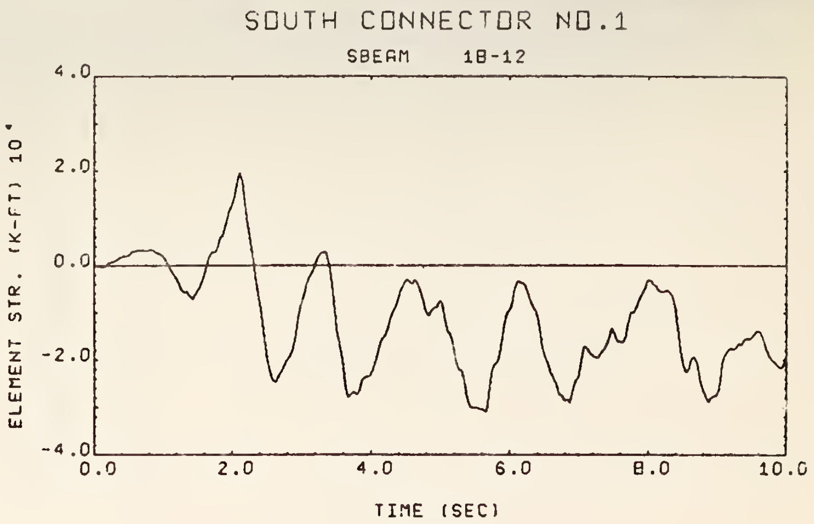
SBEAM 18-11

About the Local y - Axis



SOUTH CONNECTOR NO.1

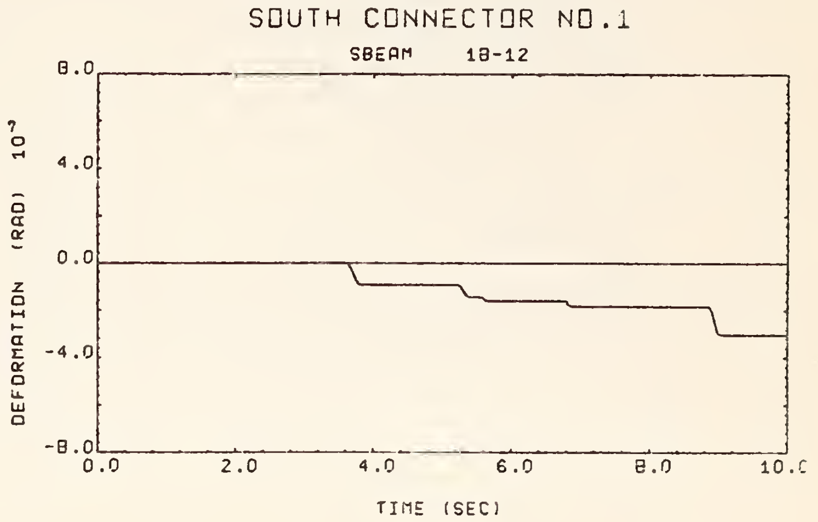
SBEAM 18-11



SOUTH CONNECTOR NO.1

SBEAM 18-12

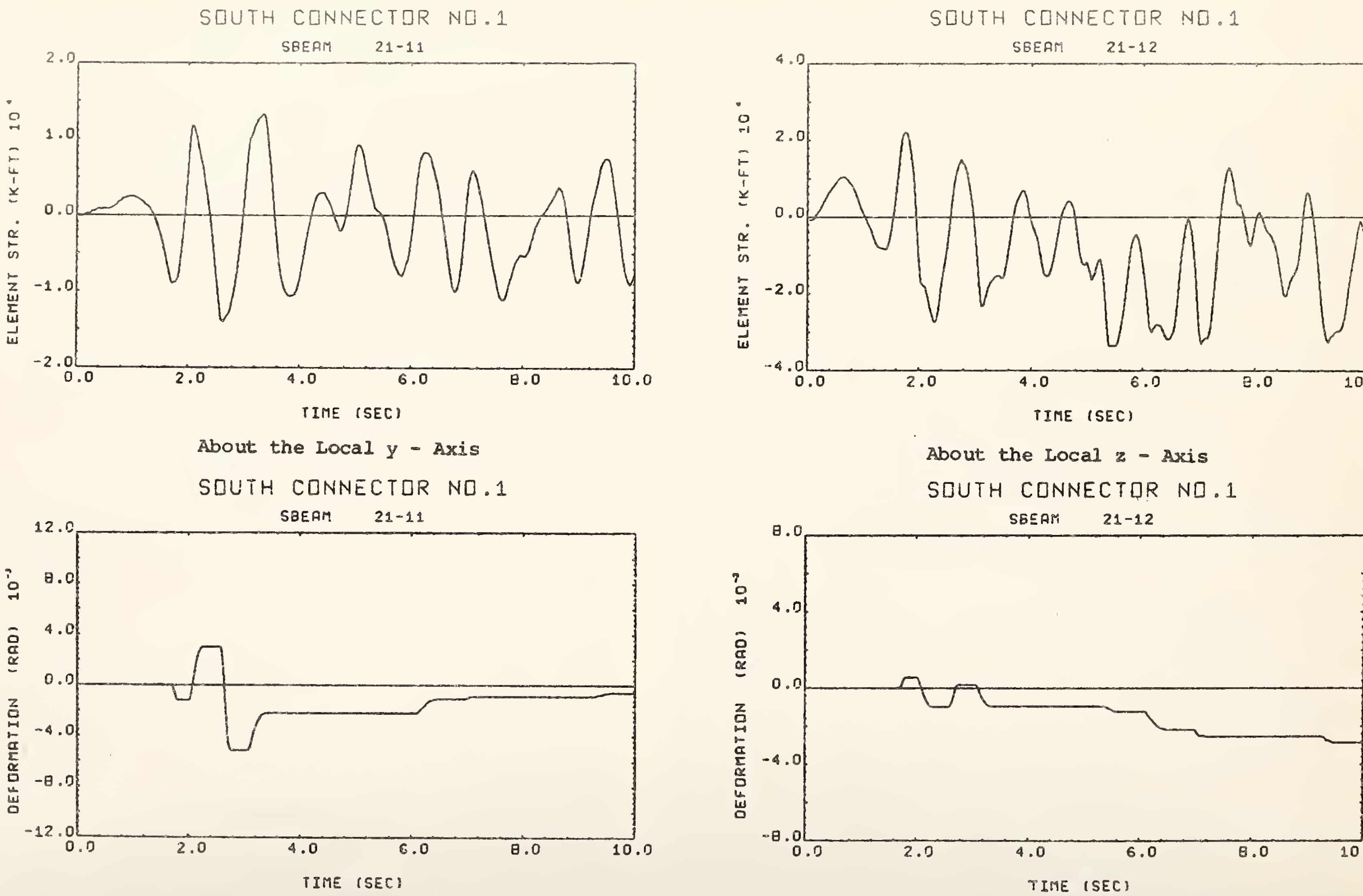
About the Local z - Axis



SOUTH CONNECTOR NO.1

SBEAM 18-12

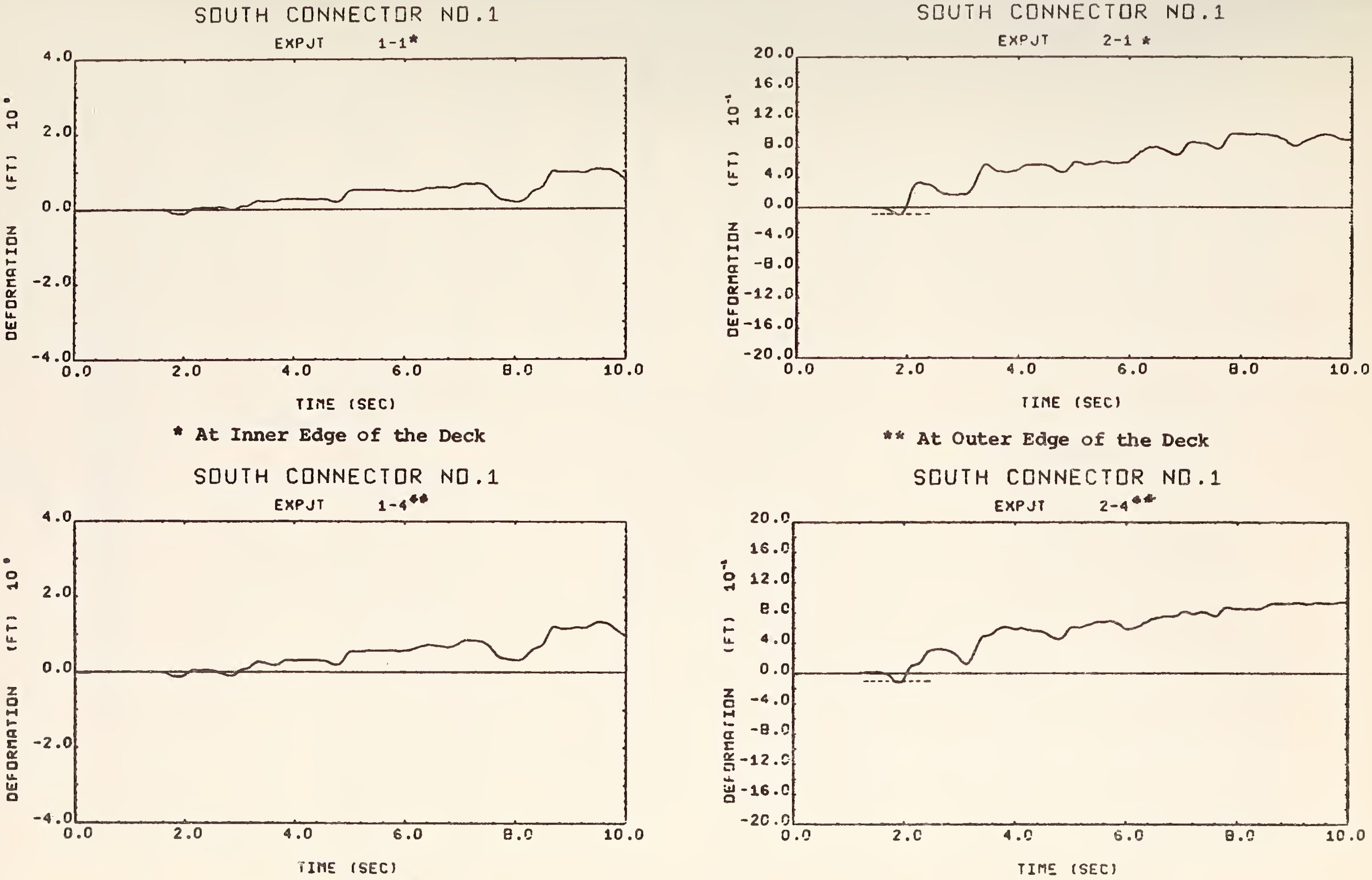
Fig. 27 Bending Moments and the Corresponding Plastic Rotations at the Base of Column # 5



About the Local y - Axis

About the Local z - Axis

Fig. 28 Bending Moments and the Corresponding Plastic Rotations at the Base of Column # 6


SOUTH CONNECTOR NO.1 EXPJT 1-1*

SOUTH CONNECTOR NO.1 EXPJT 2-1 *

* At Inner Edge of the Deck

** At Outer Edge of the Deck

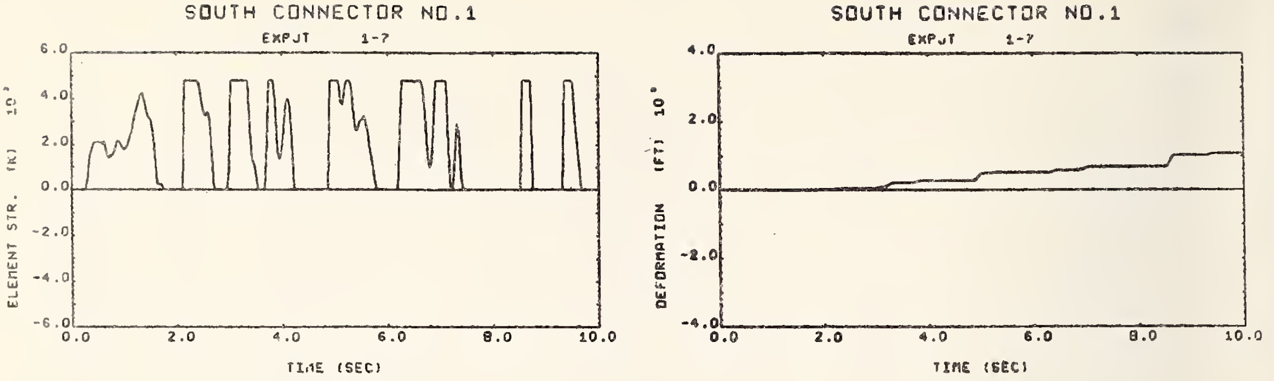
SOUTH CONNECTOR NO.1 EXPJT 1-4**

SOUTH CONNECTOR NO.1 EXPJT 2-4**

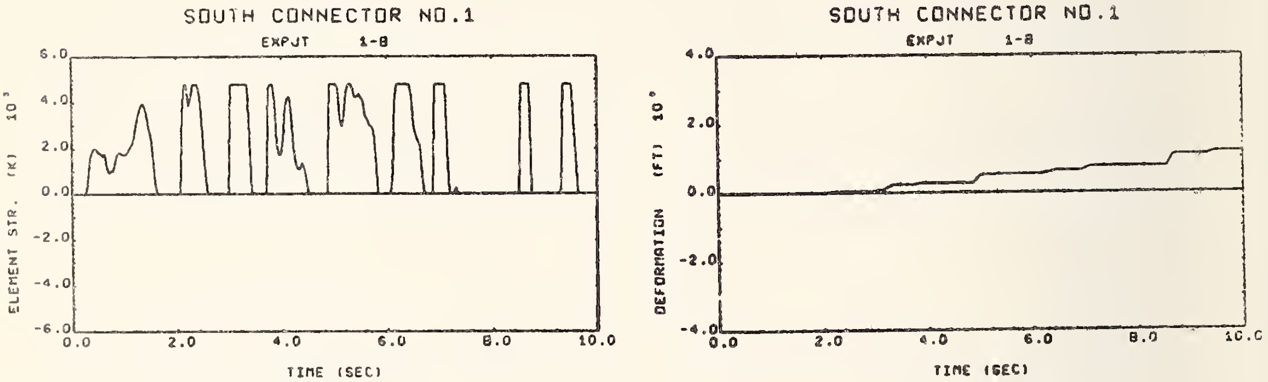
Expansion Joint # 1

Expansion Joint # 2

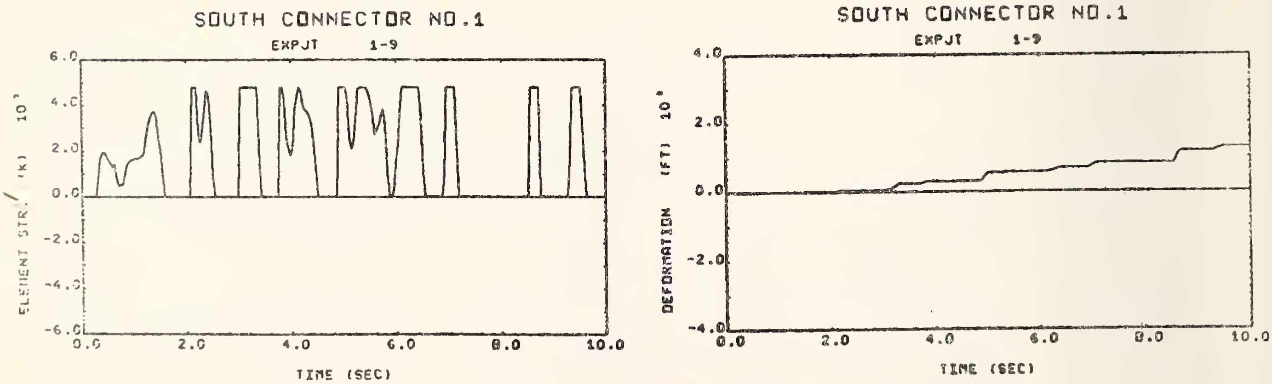
Fig. 29 Longitudinal Joint Separations at Expansion Joints # 1 and # 2



Tie Bar # 1 **Tie Bar # 1**

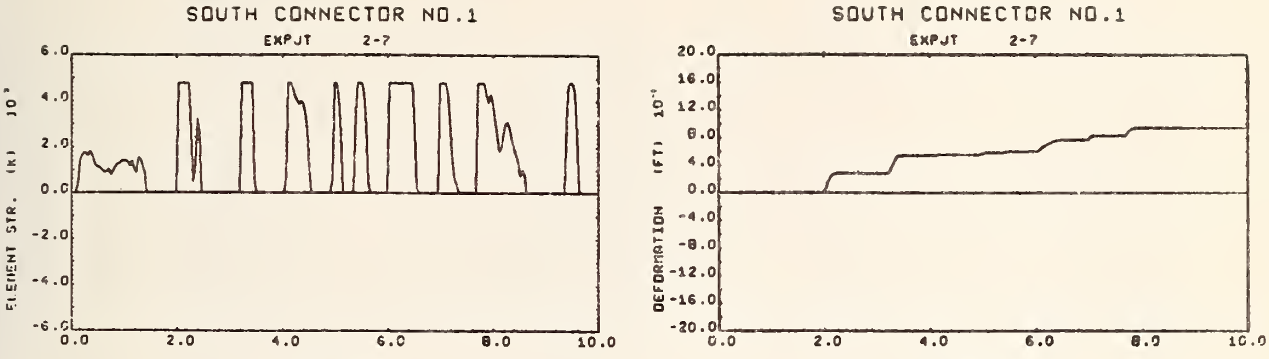


Tie Bar # 2 **Tie Bar # 2**



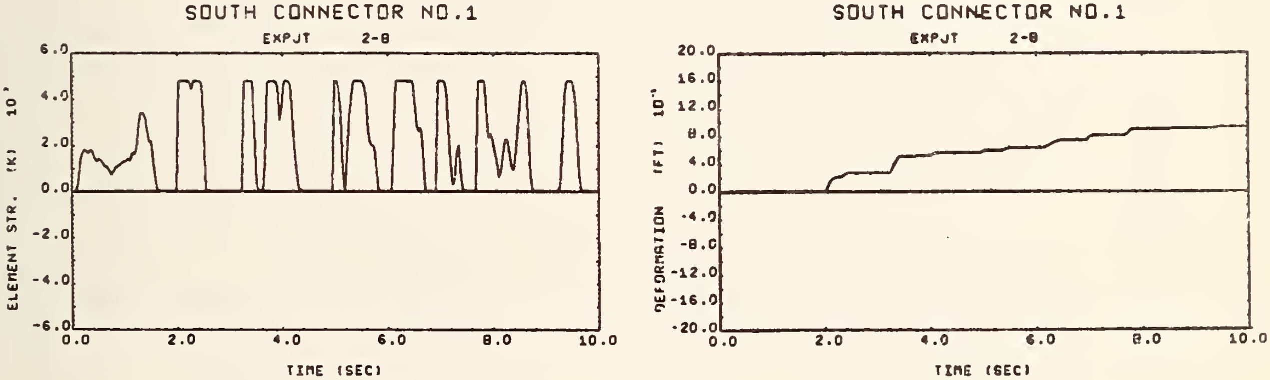
Tie Bar # 3 **Tie Bar # 3**

Fig. 30 Longitudinal Tie Bar Forces and the Corresponding Plastic Elongations at Expansion Joint # 1



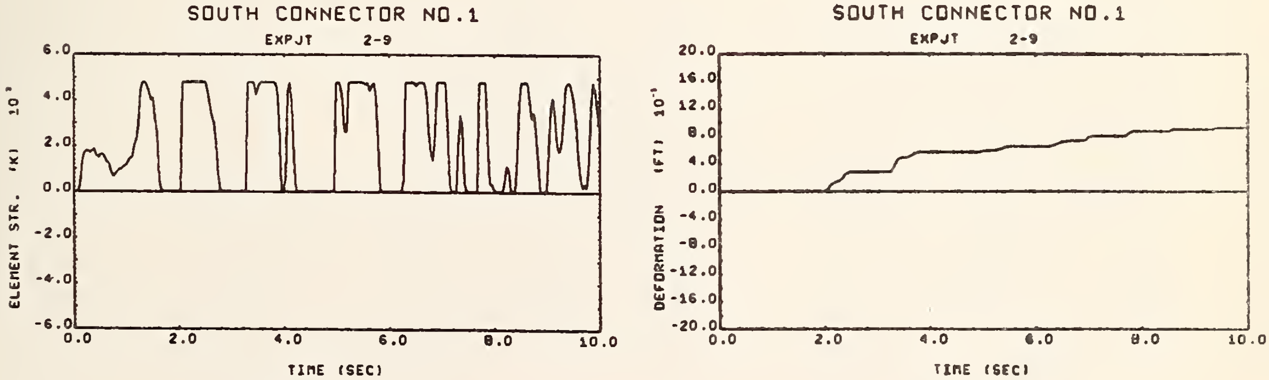
Tie Bar # 1

Tie Bar # 1



Tie Bar # 2

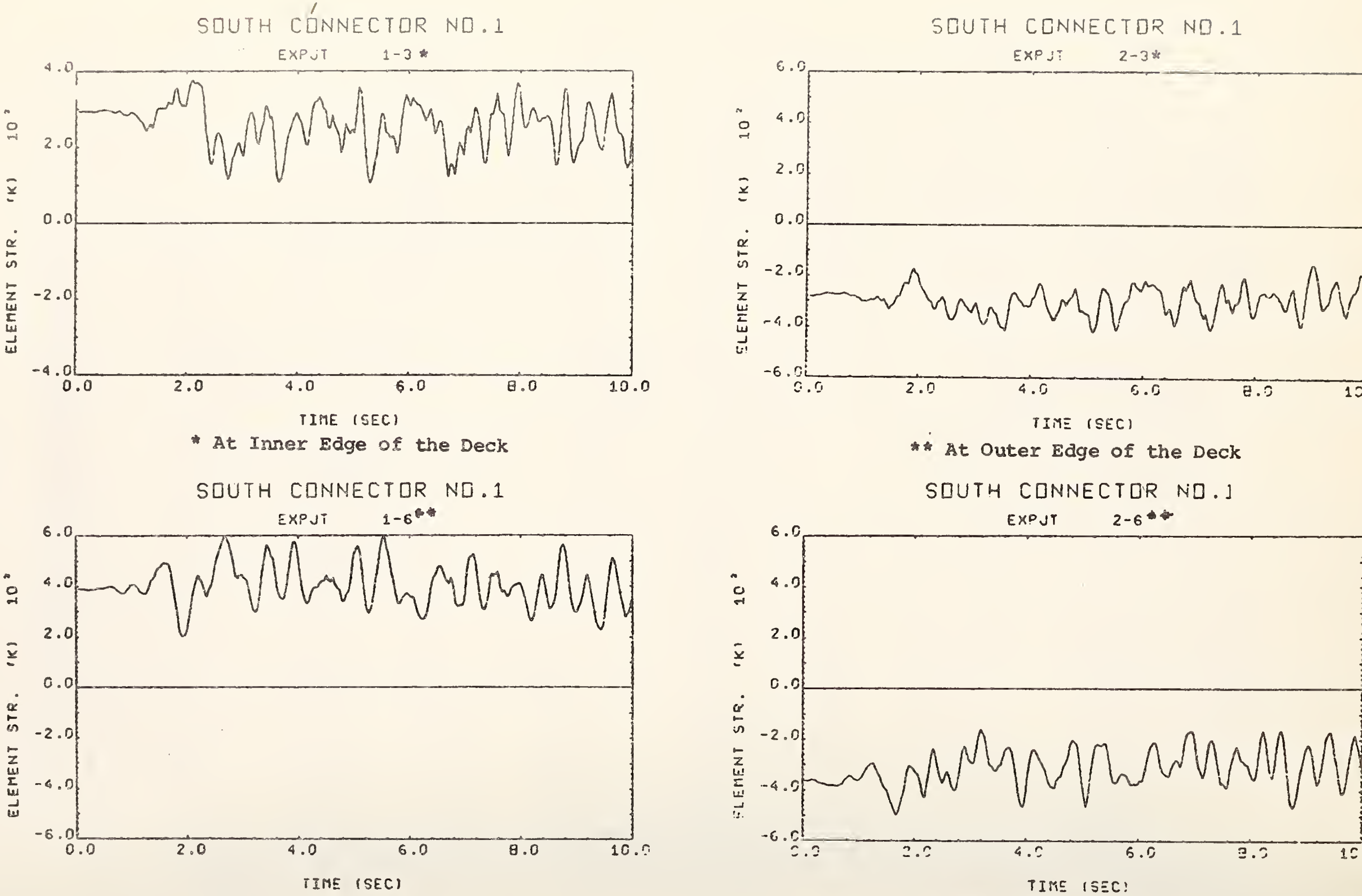
Tie Bar # 2



Tie Bar # 3

Tie Bar # 3

Fig. 31 Longitudinal Tie Bar Forces and the Corresponding Plastic Elongations at Expansion Joint # 2



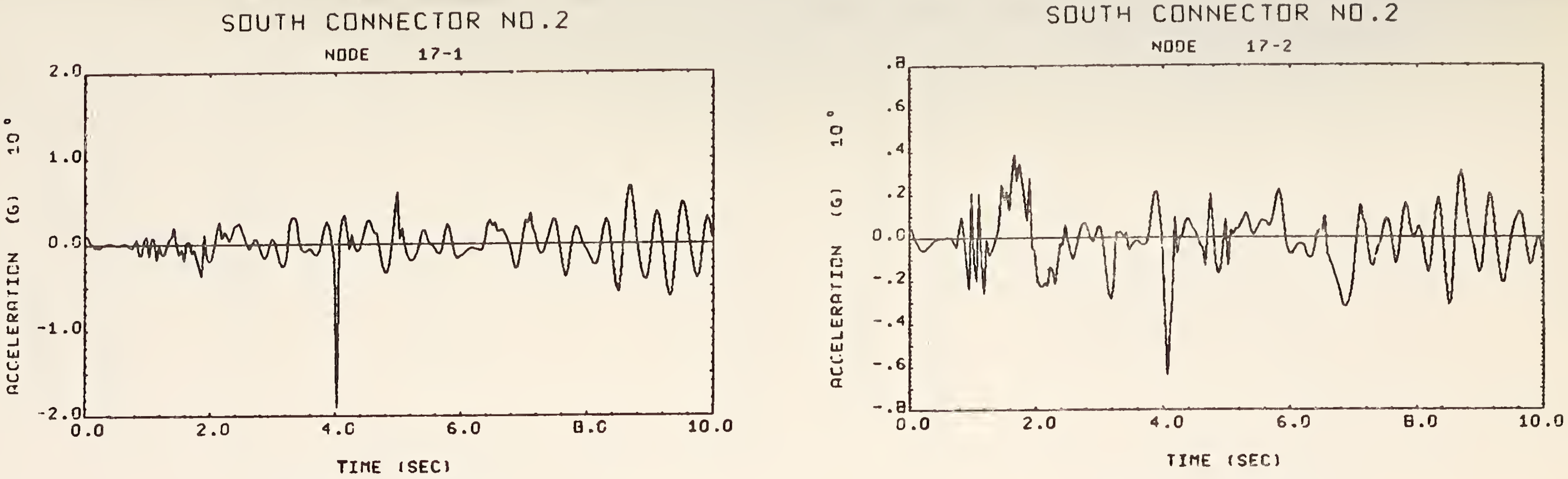
* At Inner Edge of the Deck

** At Outer Edge of the Deck

Expansion Joint # 1

Expansion Joint # 2

Fig. 32 Forces in the Vertical Restrainers at Expansion Joints # 1 and # 2



Global X - Component

Global Y - Component

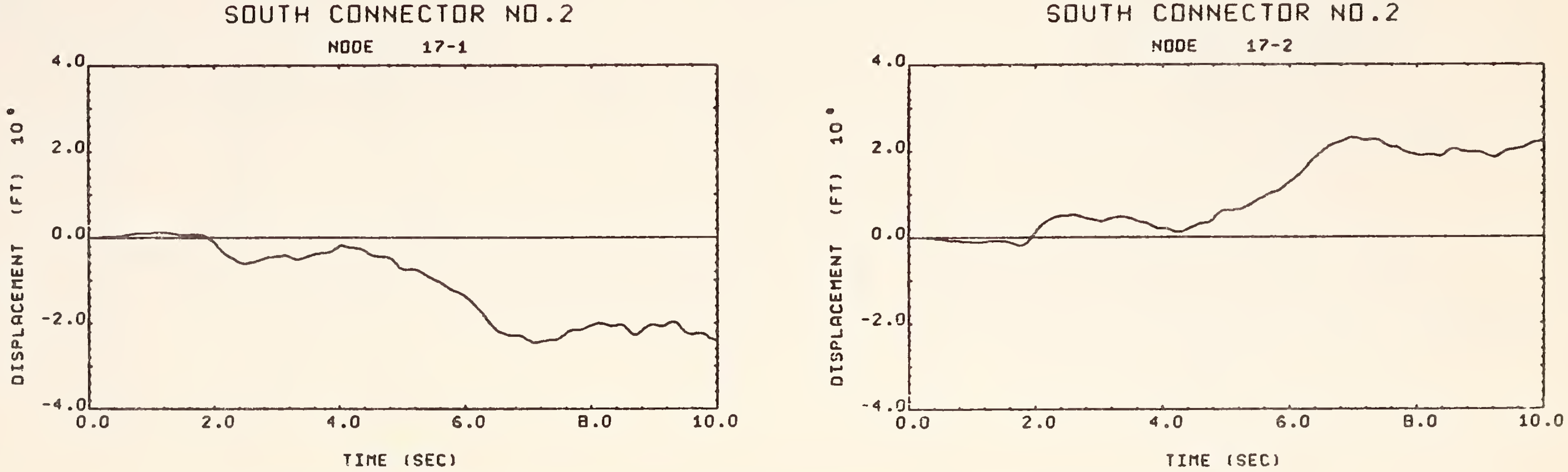
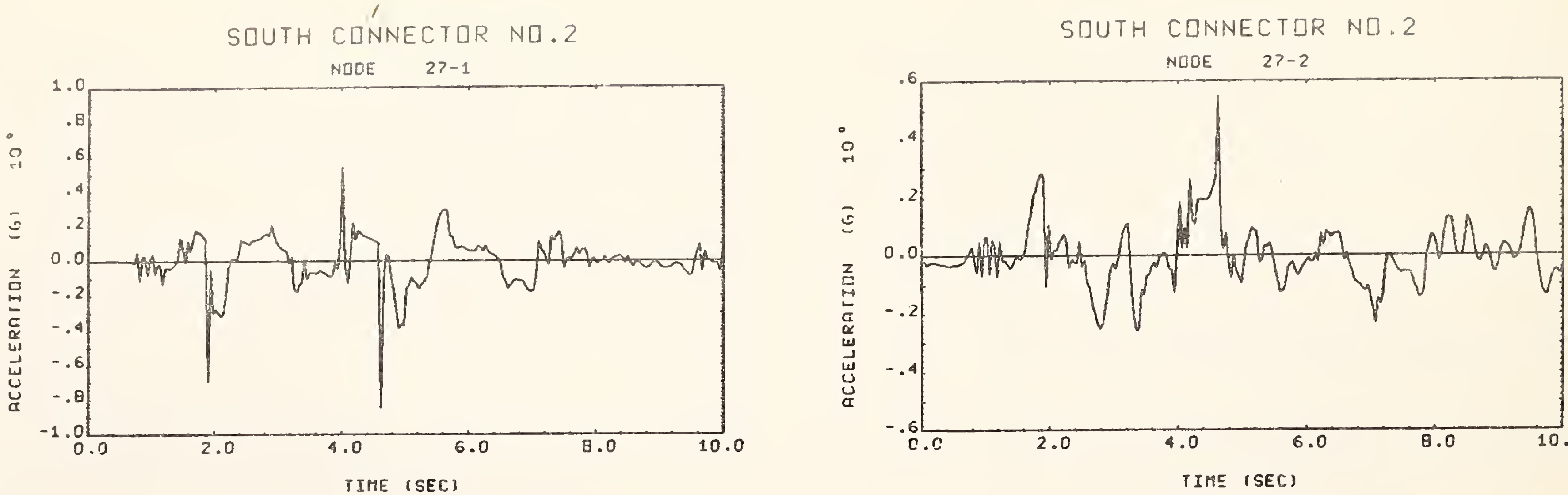


Fig. 33 Horizontal Accelerations and Displacements at the Top of Column # 3



Global X - Component

Global Y - Component

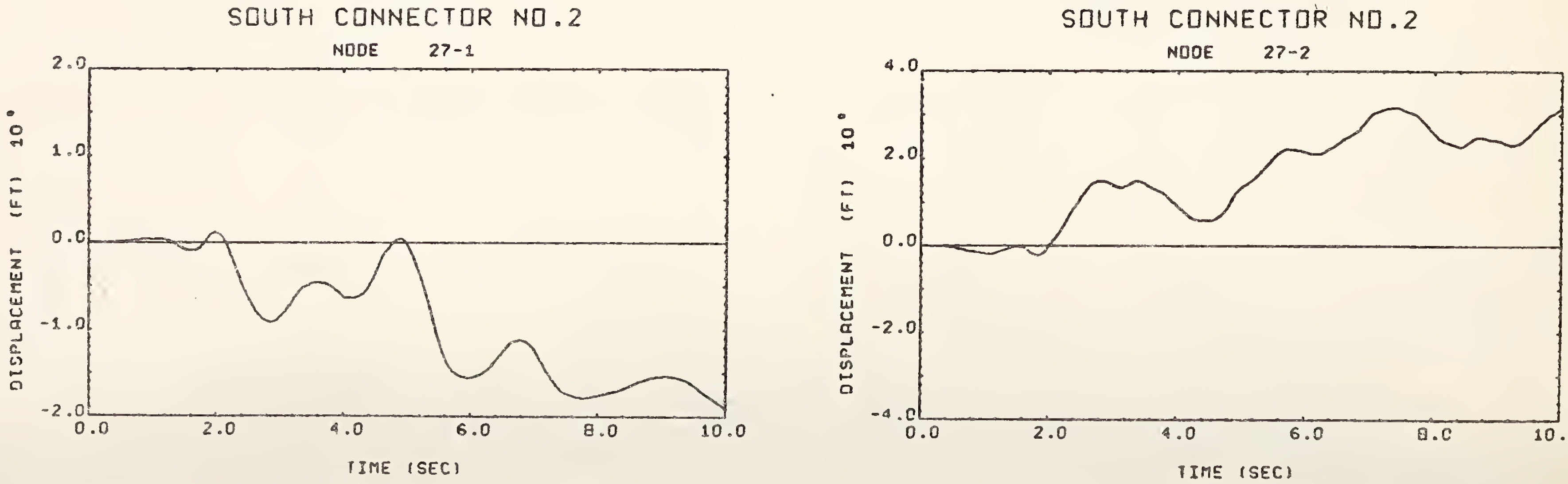
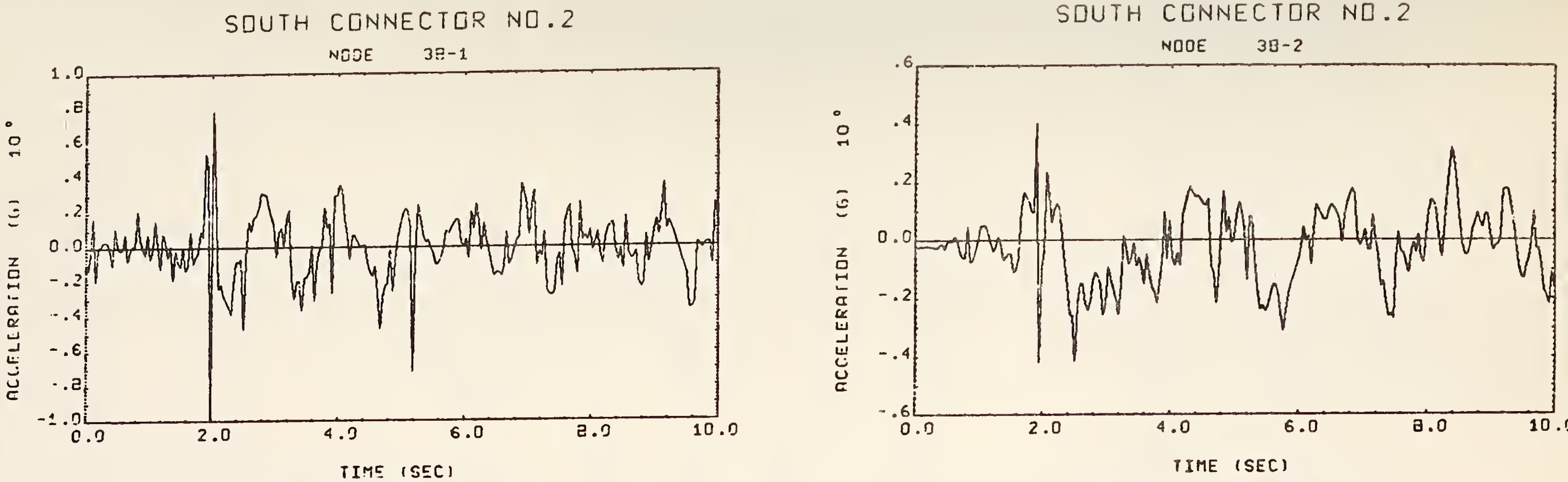


Fig. 34 Horizontal Accelerations and Displacements at the Top of Column # 4



Global X - Component **Global - Y Component**

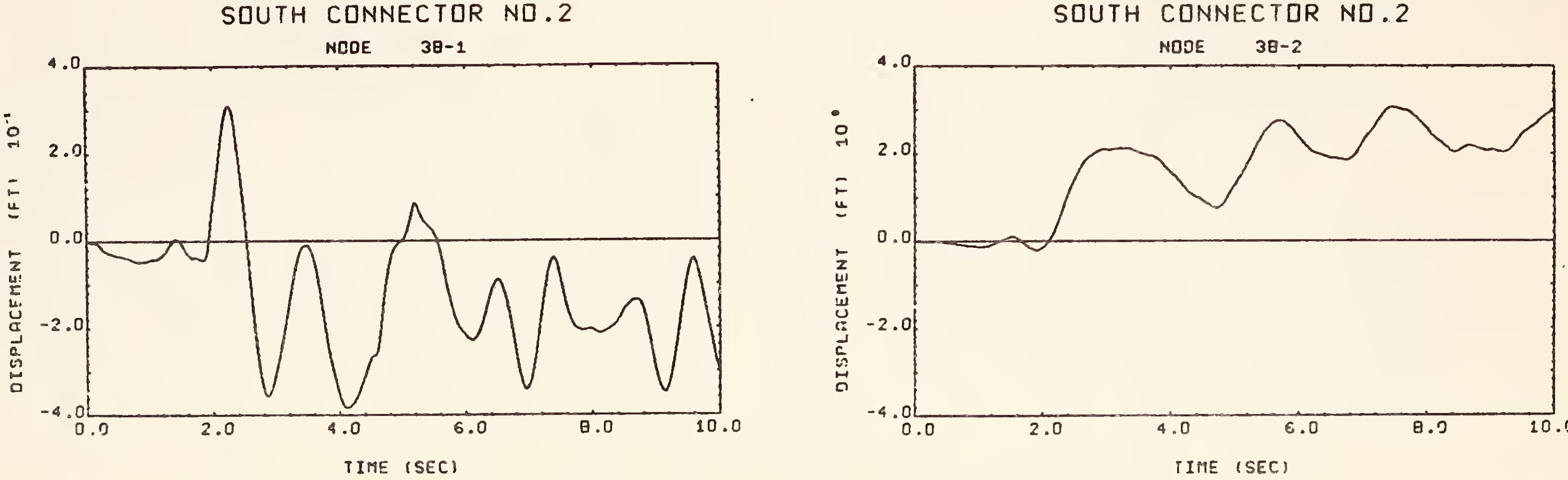
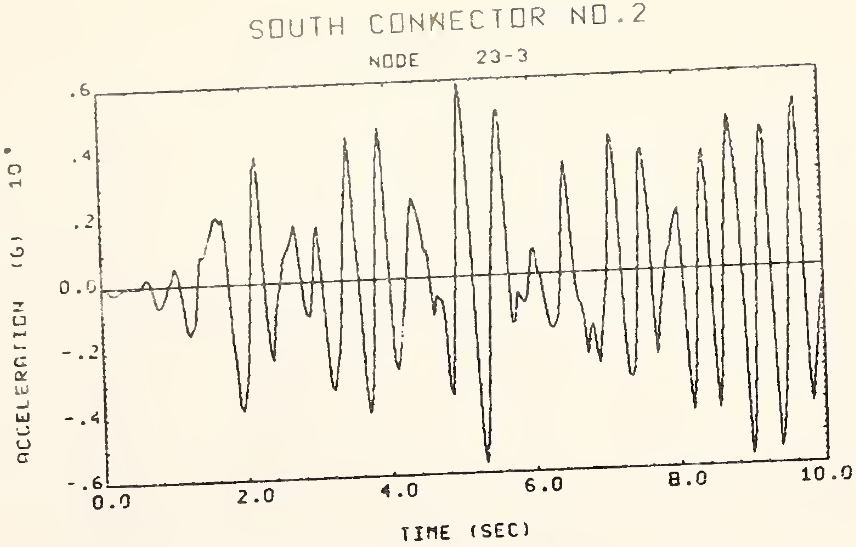
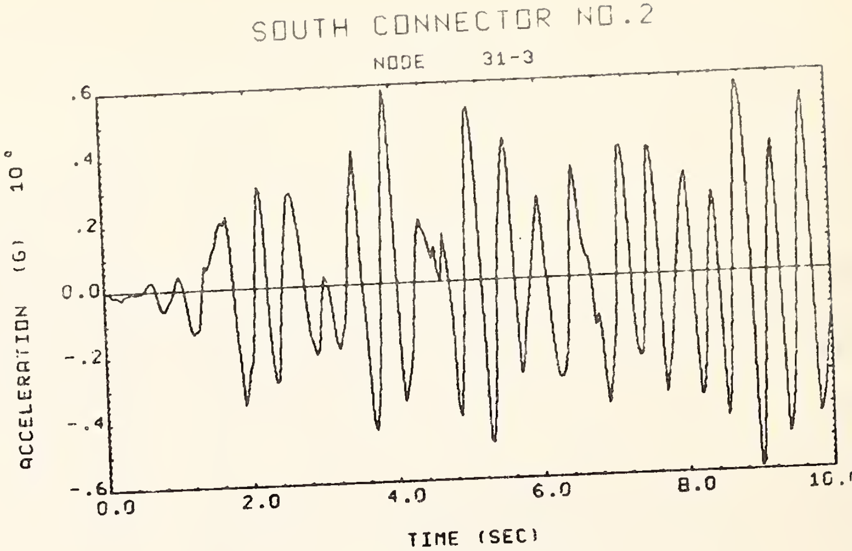


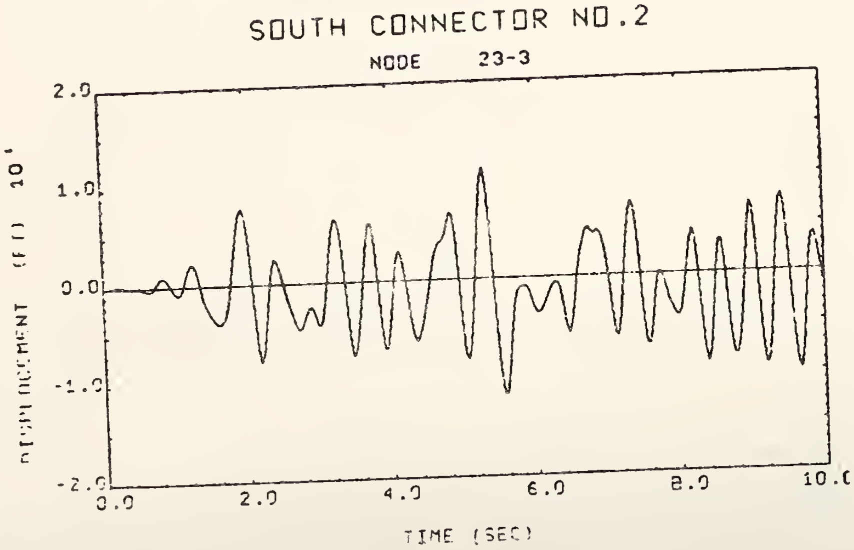
Fig. 35 Horizontal Accelerations and Displacements at the Top of Column # 5



Center of Span # 3



Center of Span # 4



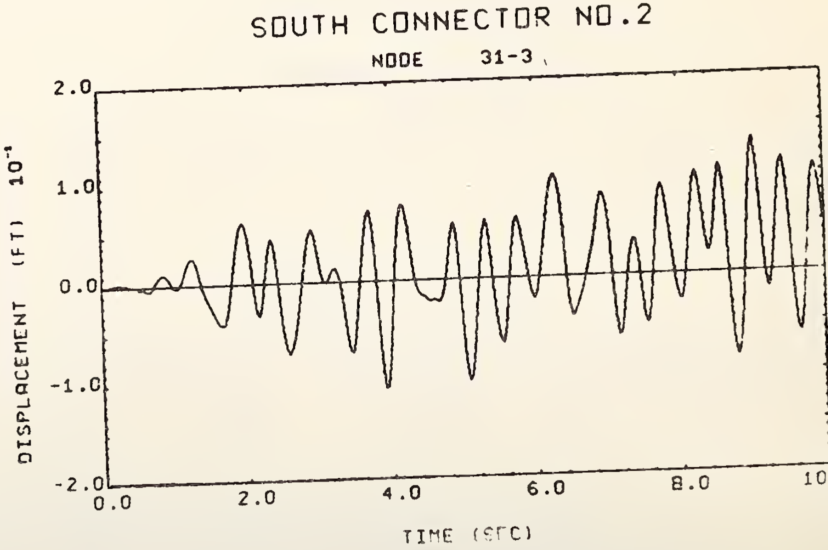
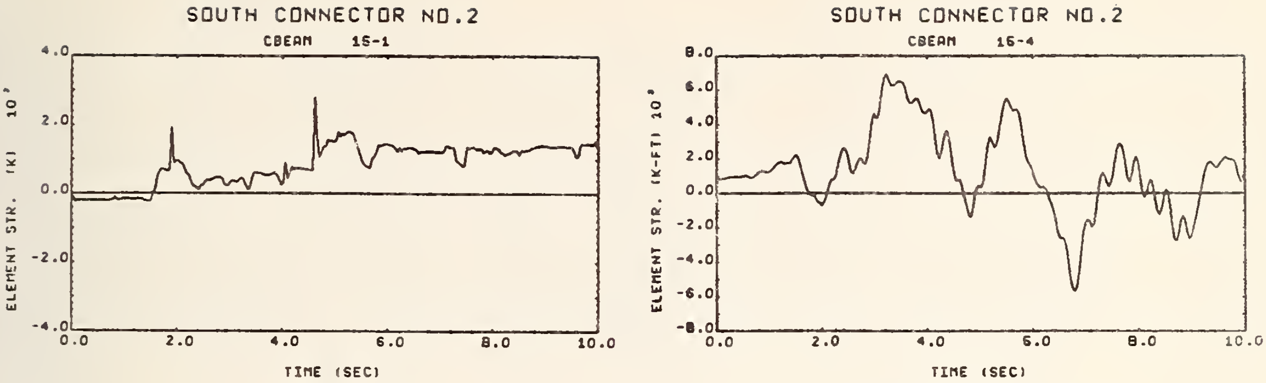
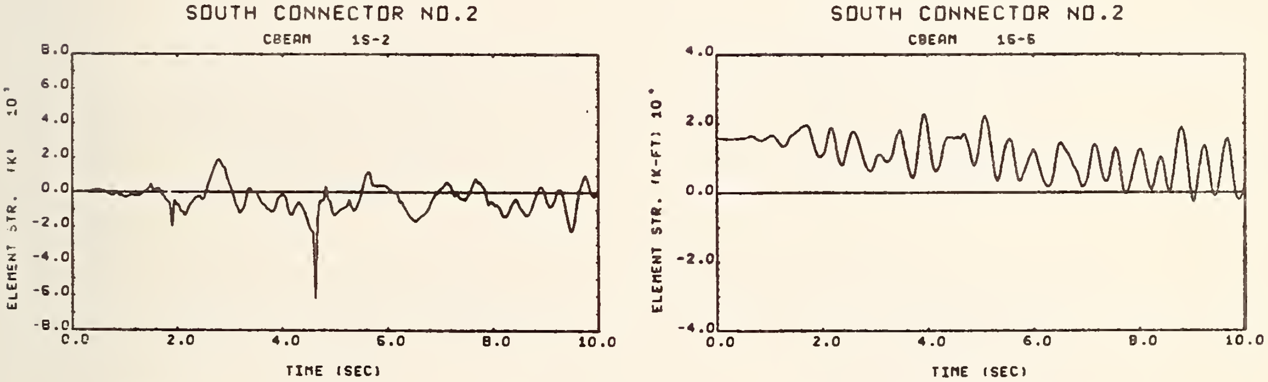


Fig. 36 Vertical Accelerations and Displacements at the Center of Spans # 3 and # 4



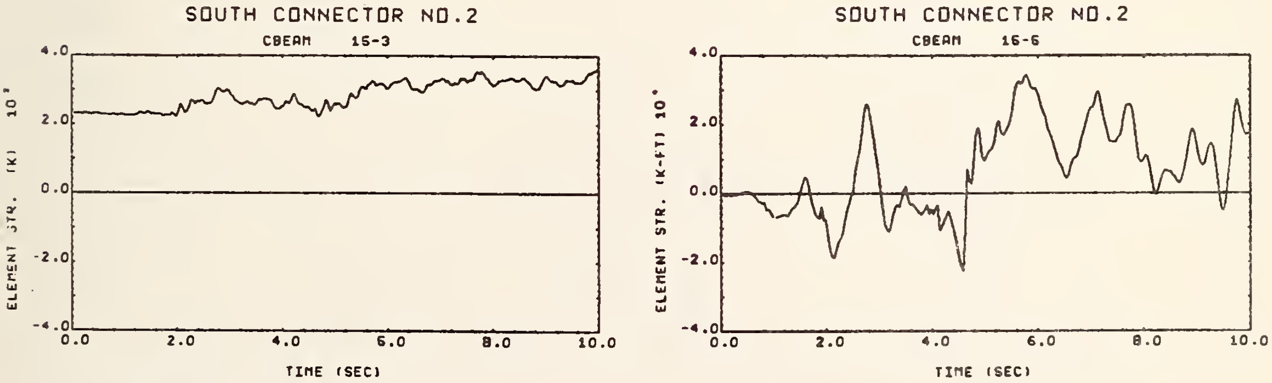
Axial Force

Torsional Moment



Shear in the Local y - Direction

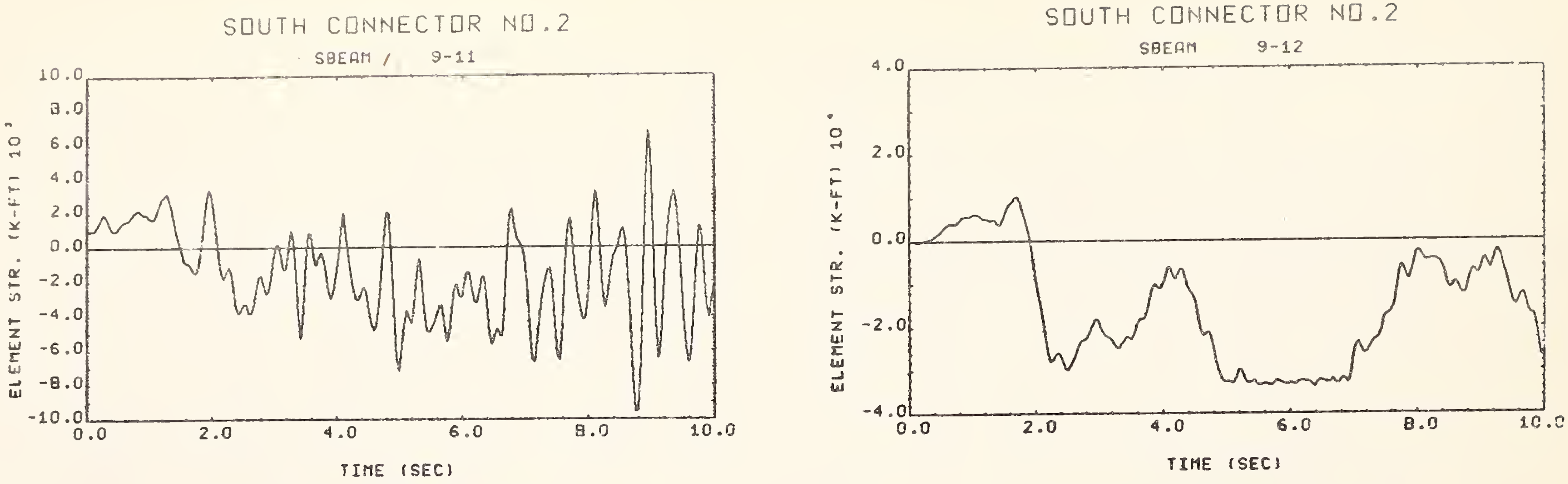
Moment about the Local y - Axis



Shear in the Local z - Direction

Moment about the Local z - Axis

Fig. 37 Generalized Forces in the Girder at the Center of Span # 4



About the Local y - Axis

About the Local z - Axis

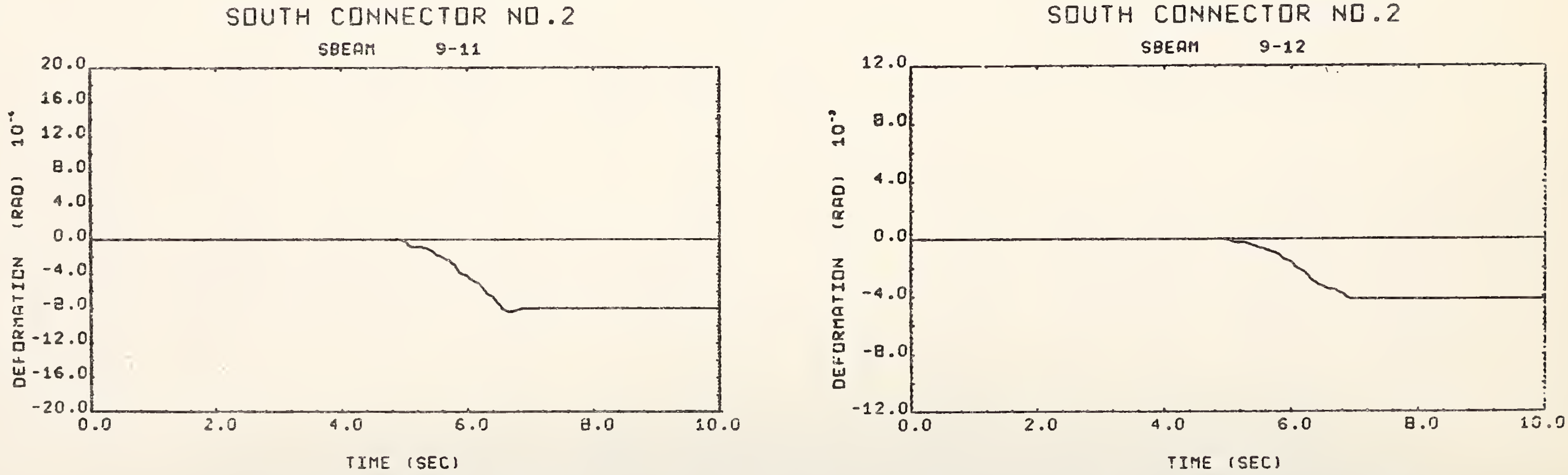
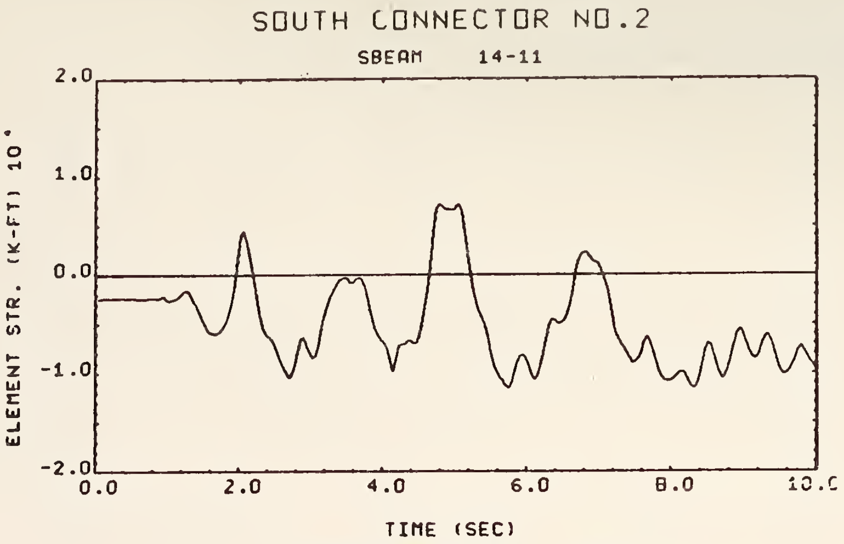
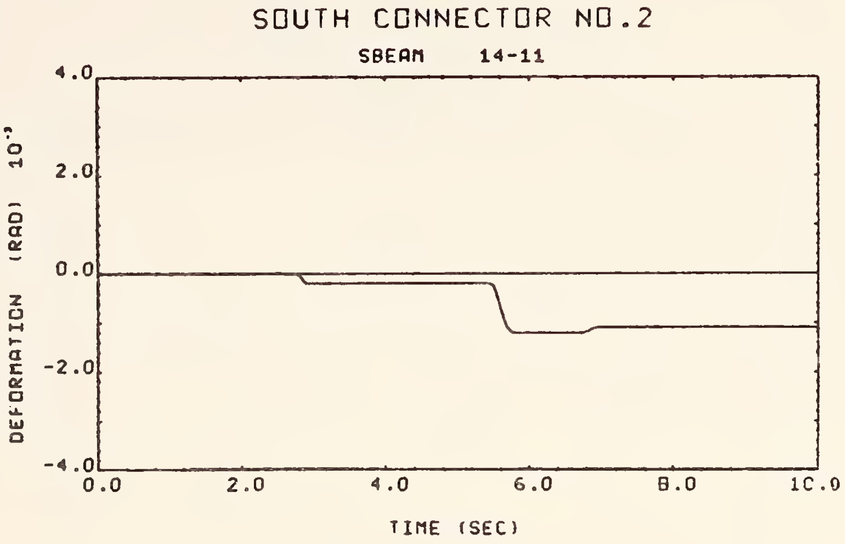
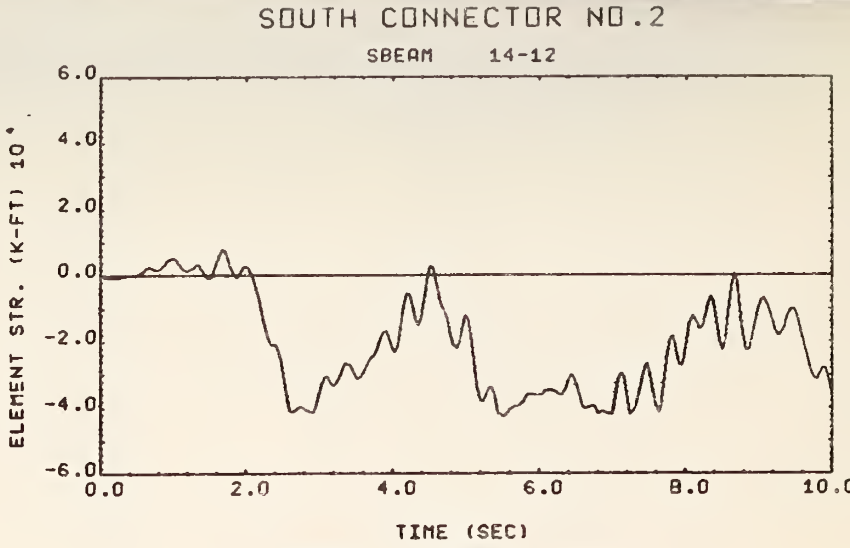


Fig. 38 Bending Moments and the Corresponding Plastic Rotations at the Base of Column # 3



About the Local y - Axis





About the Local z - Axis

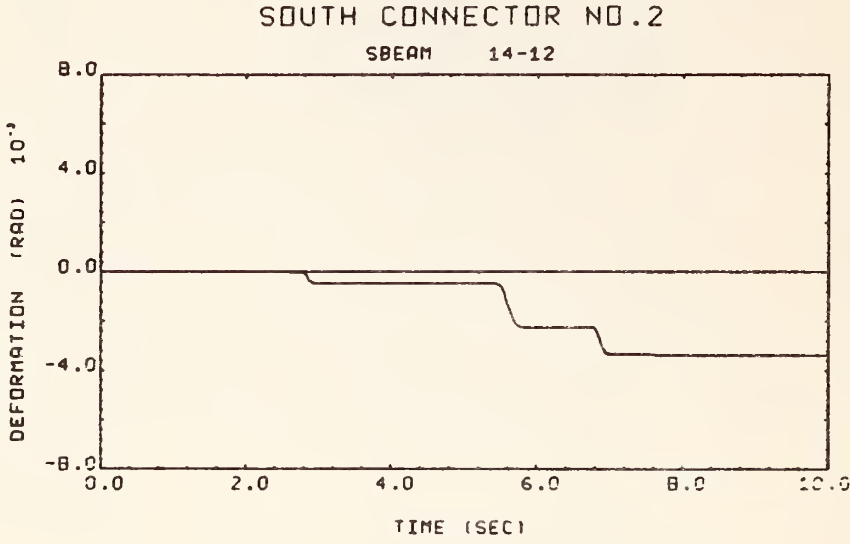
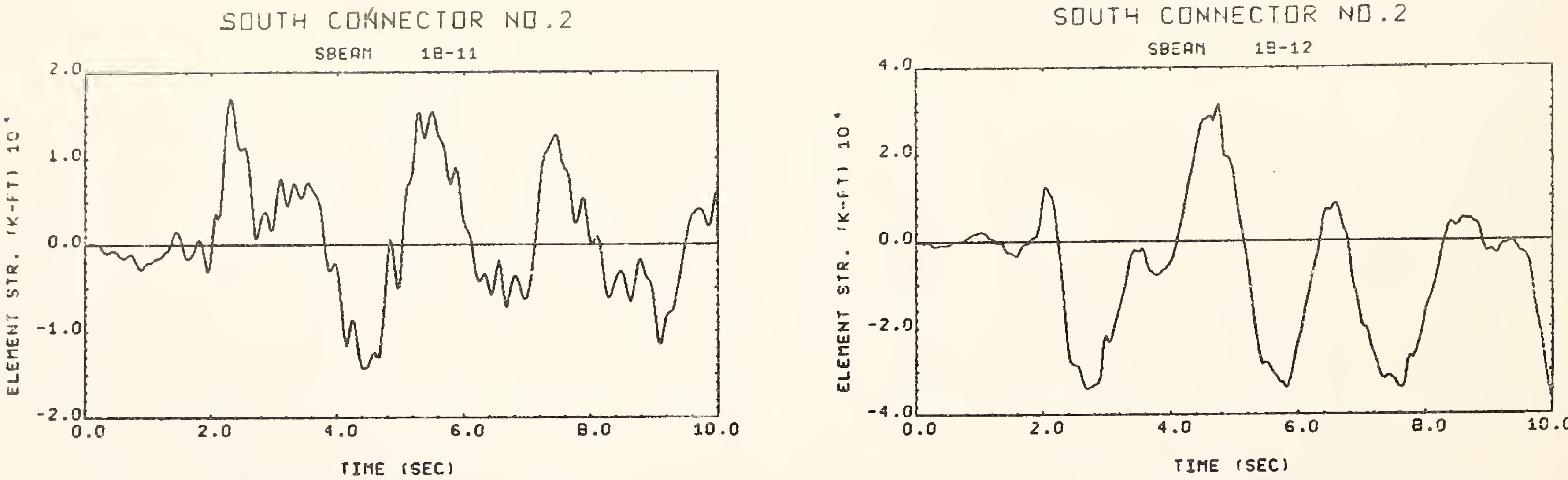


Fig. 39 Bending Moments and the Corresponding Plastic Rotations at the Base of Column # 4



About the Local y - Axis

About the Local z - Axis

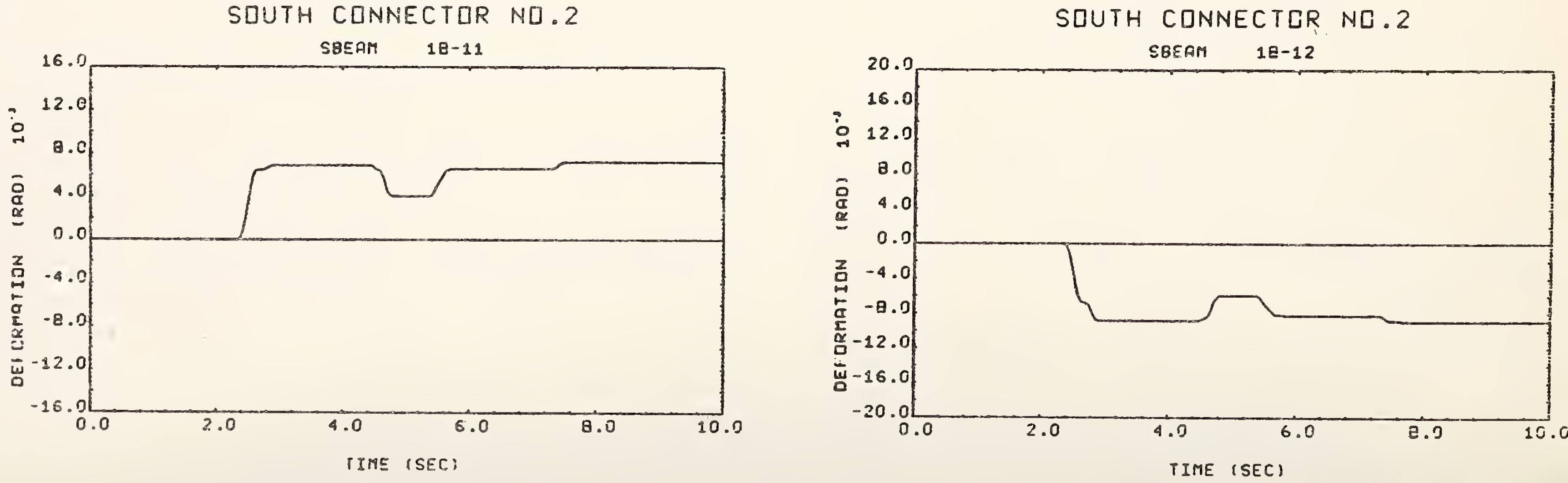
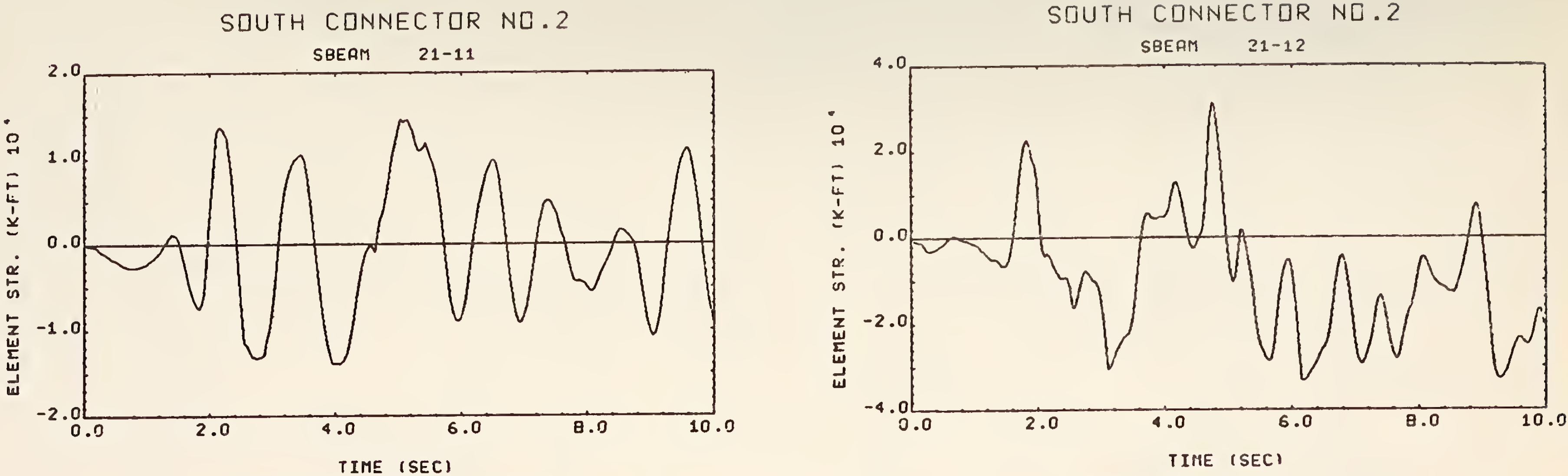


Fig. 40 Bending Moments and the Corresponding Plastic Rotations at the Base of Column # 5



About the Local y - Axis

About the Local z - Axis

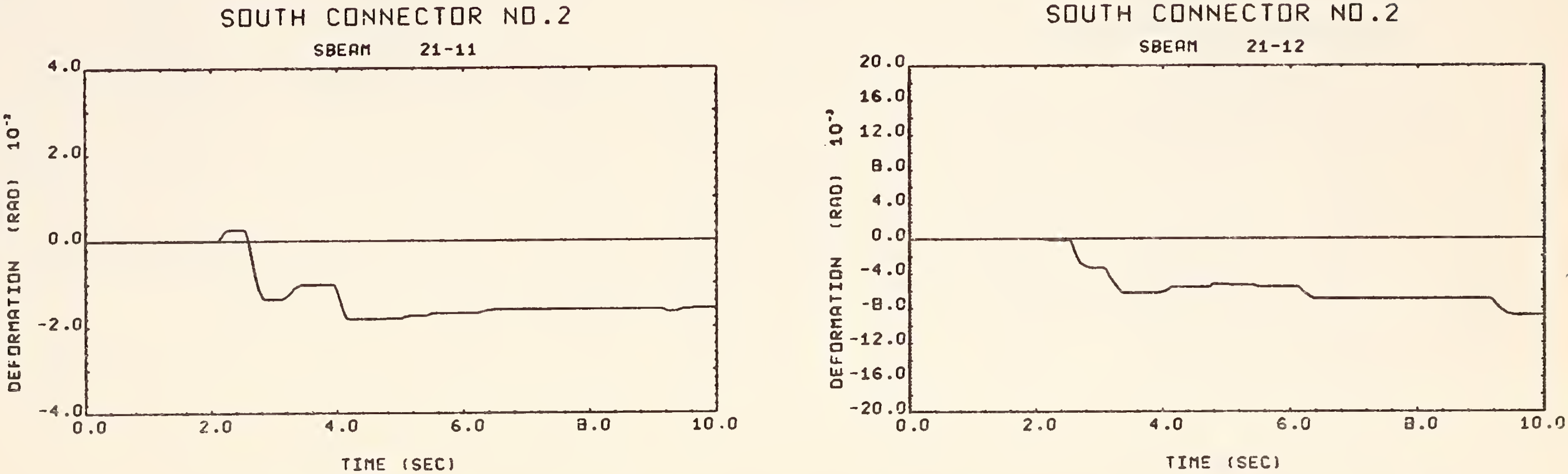
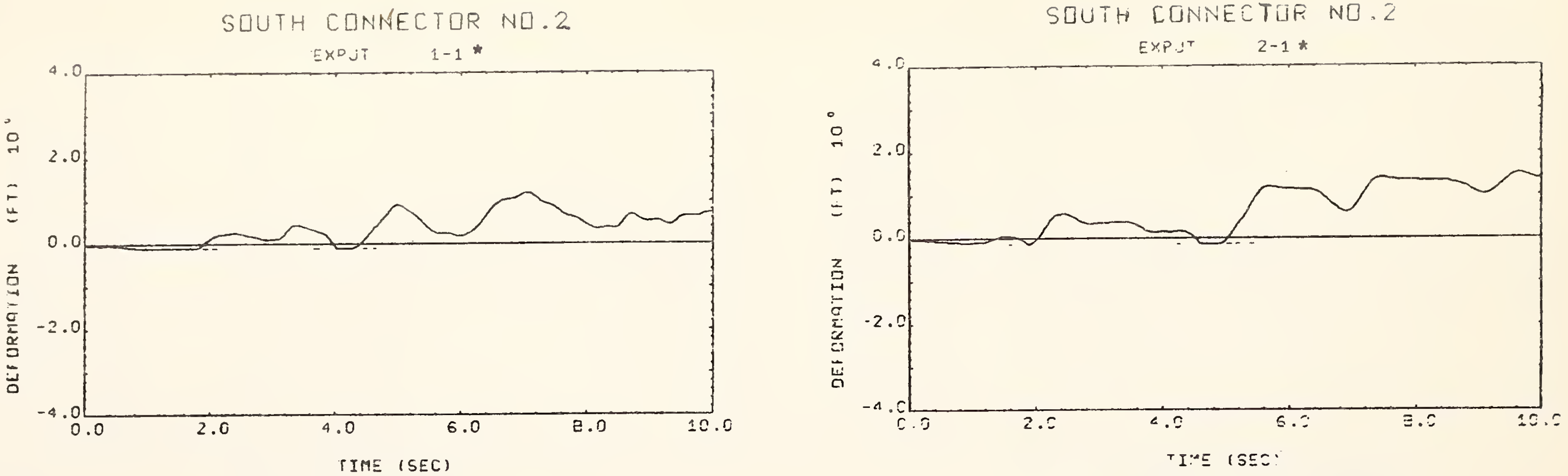
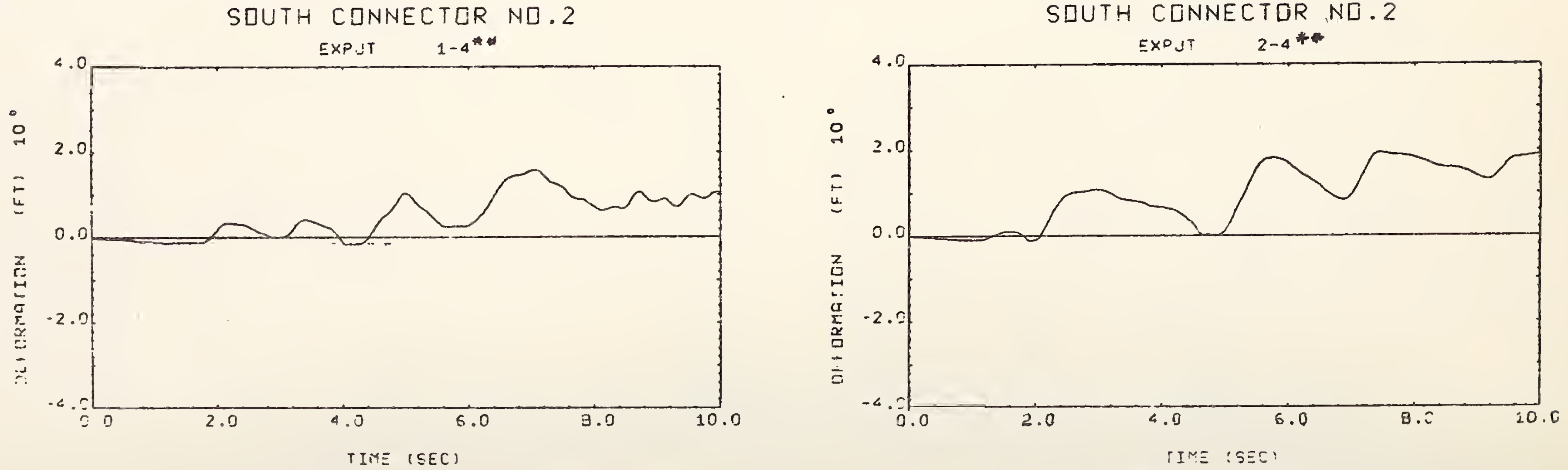


Fig. 41 Bending Moments and the Corresponding Plastic Rotations at the Base of Column # 6



* At Inner Edge of the Deck

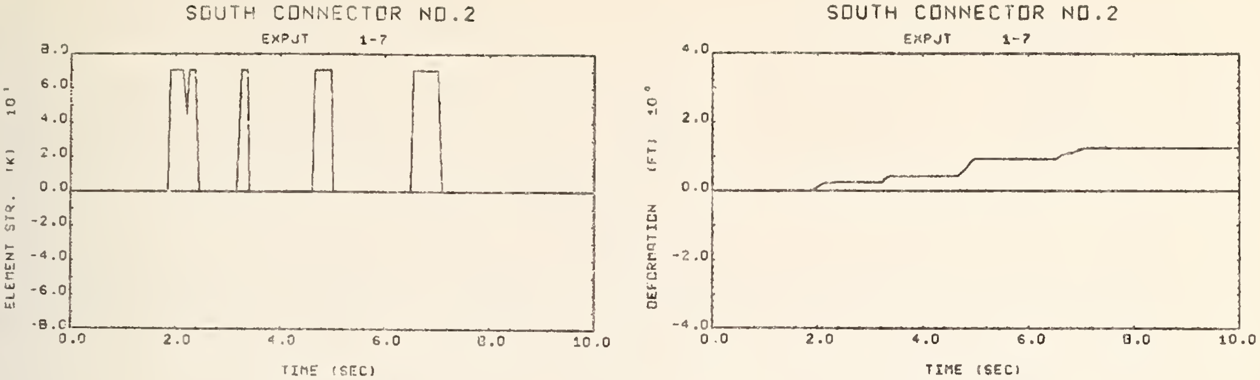
** At Outer Edge of the Deck



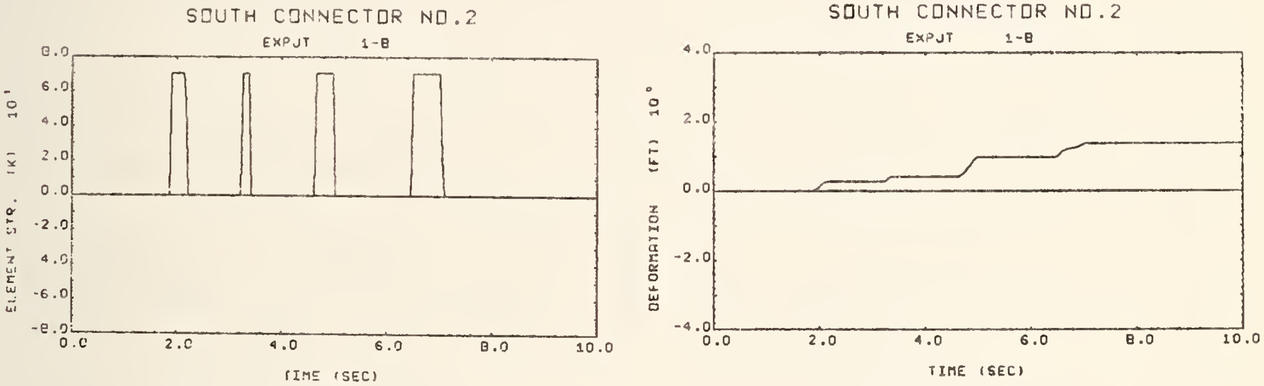
Expansion Joint # 1

Expansion Joint # 2

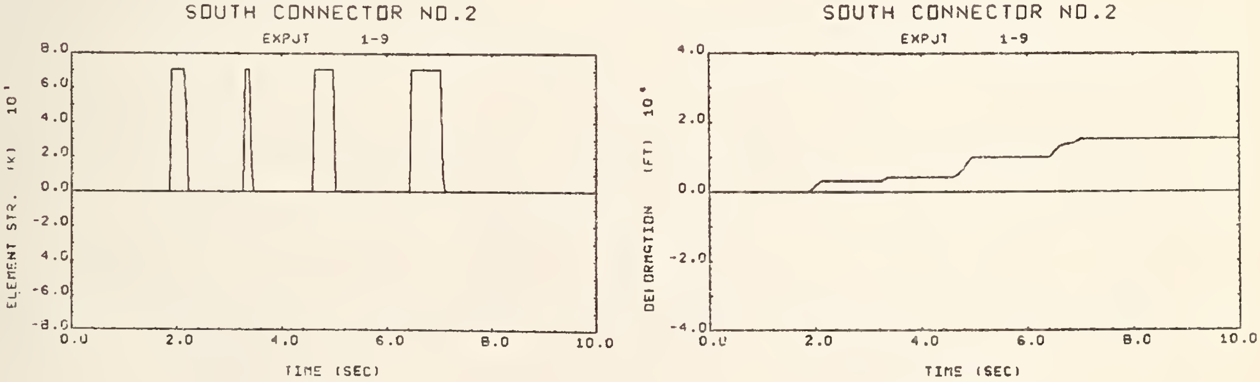
Fig. 42 Longitudinal Joint Separations at Expansion Joints # 1 and # 2



Tie Bar # 1 Tie Bar # 1

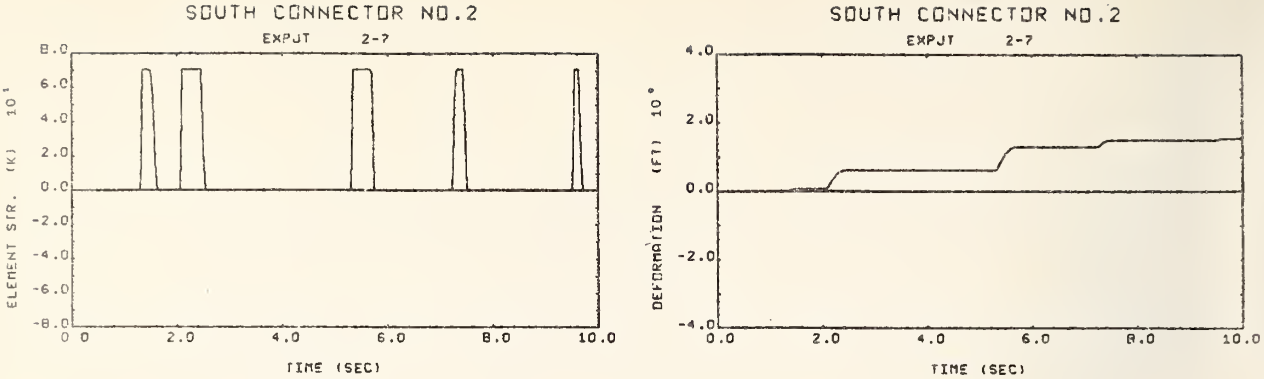


Tie Bar # 2 Tie Bar # 2



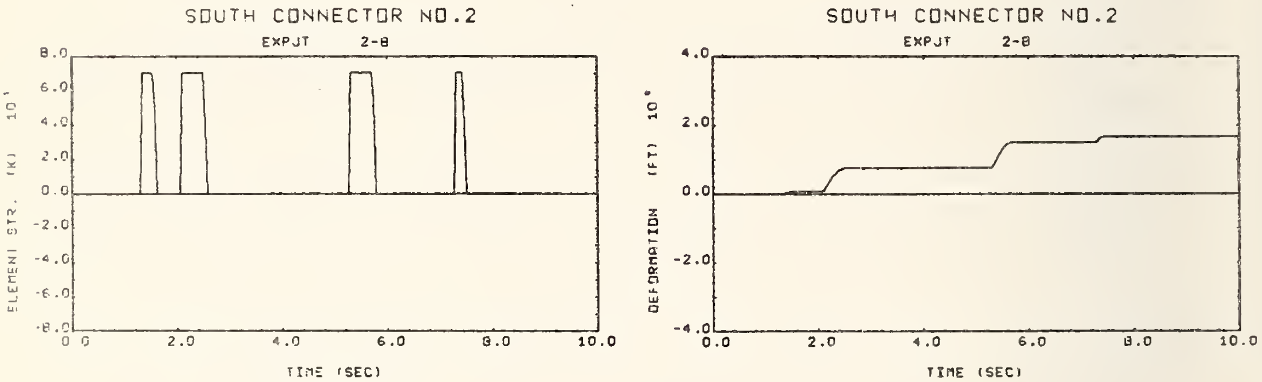
Tie Bar # 3 Tie Bar # 3

Fig. 43 Longitudinal Tie Bar Forces and the Corresponding Plastic Elongations at Expansion Joint # 1



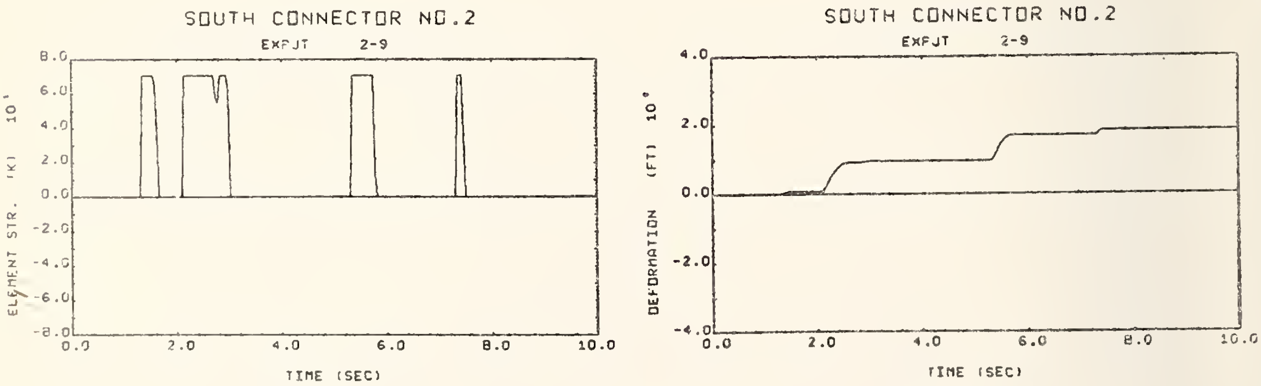
Tie Bar # 1

Tie Bar # 1



Tie Bar # 2

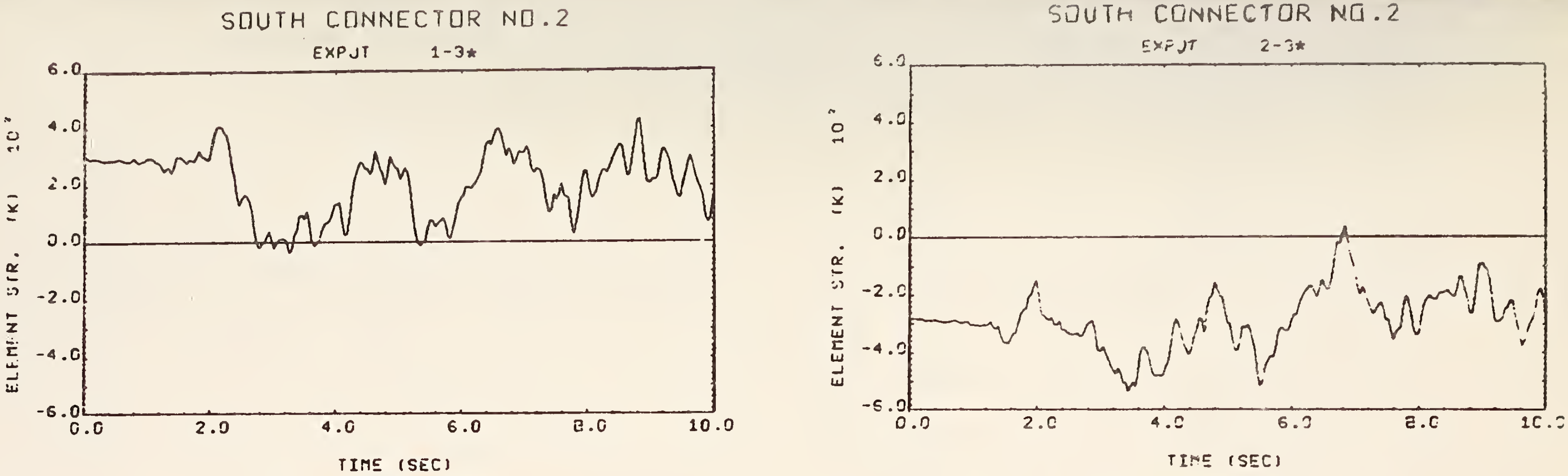
Tie Bar # 2



Tie Bar # 3

Tie Bar # 3

Fig. 44 Longitudinal Tie Bar Forces and the Corresponding Plastic Elongations at Expansion Joint # 2

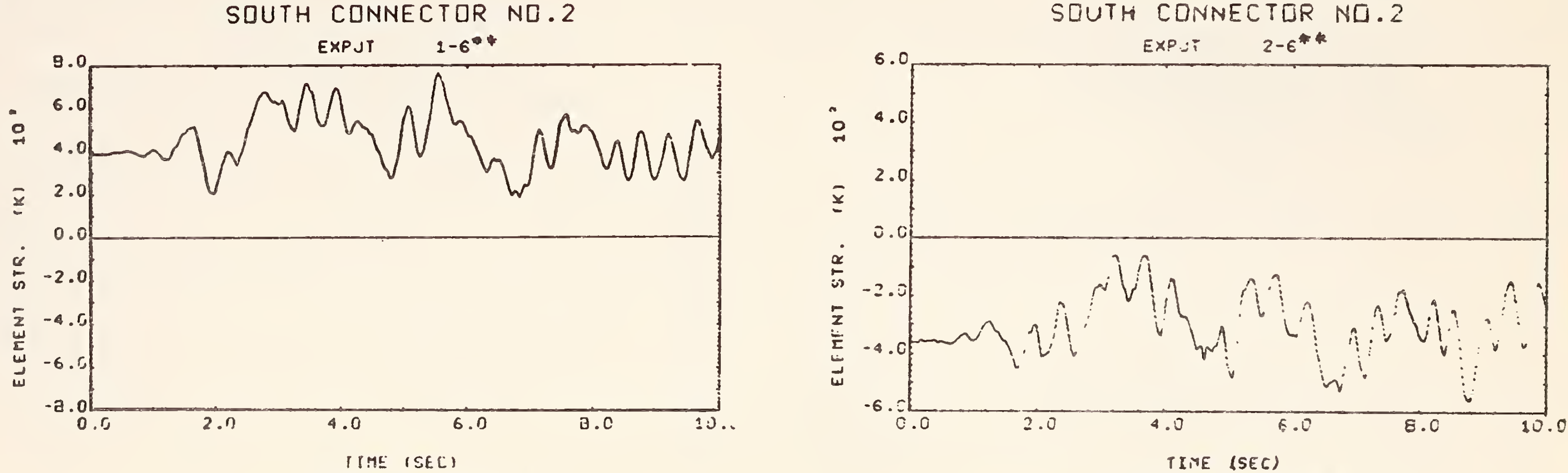


SOUTH CONNECTOR NO.2 EXPJT 1-3*

SOUTH CONNECTOR NO.2 EXPJT 2-3*

* At Inner Edge of the Deck

** At Outer Edge of the Deck



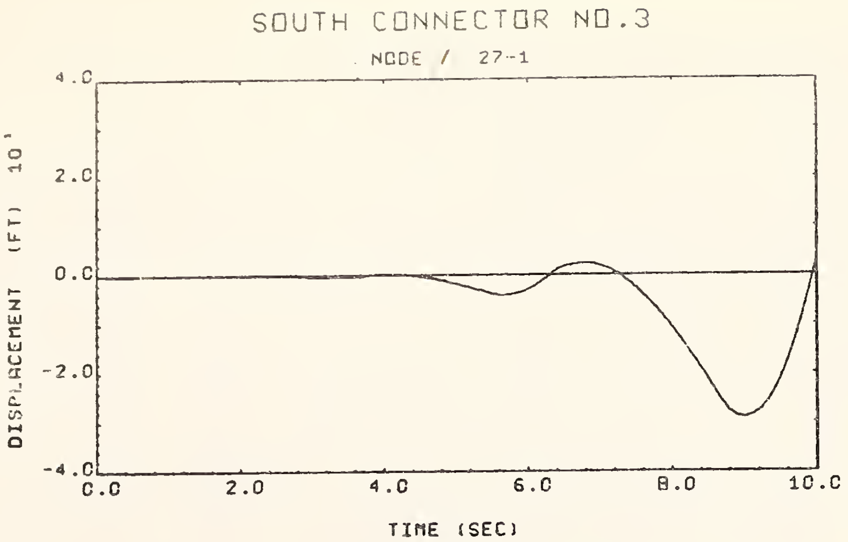
SOUTH CONNECTOR NO.2 EXPJT 1-6**

SOUTH CONNECTOR NO.2 EXPJT 2-6**

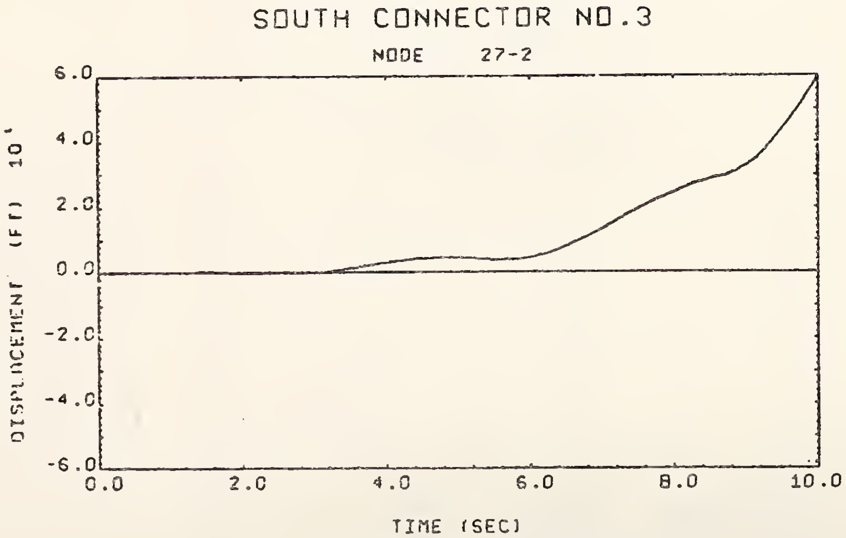
Expansion Joint # 1

Expansion Joint # 2

Fig. 45 Forces in Vertical Restrainers at Expansion Joint # 1 and # 2

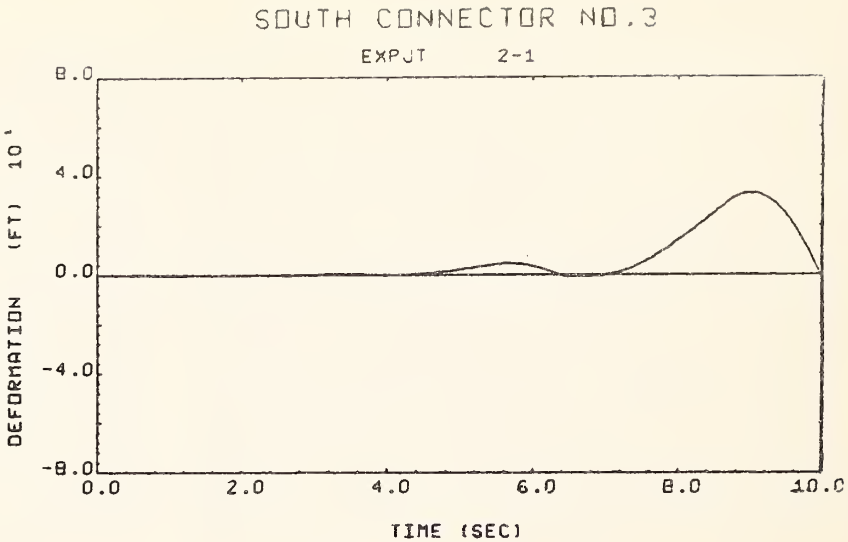


Global X - Component

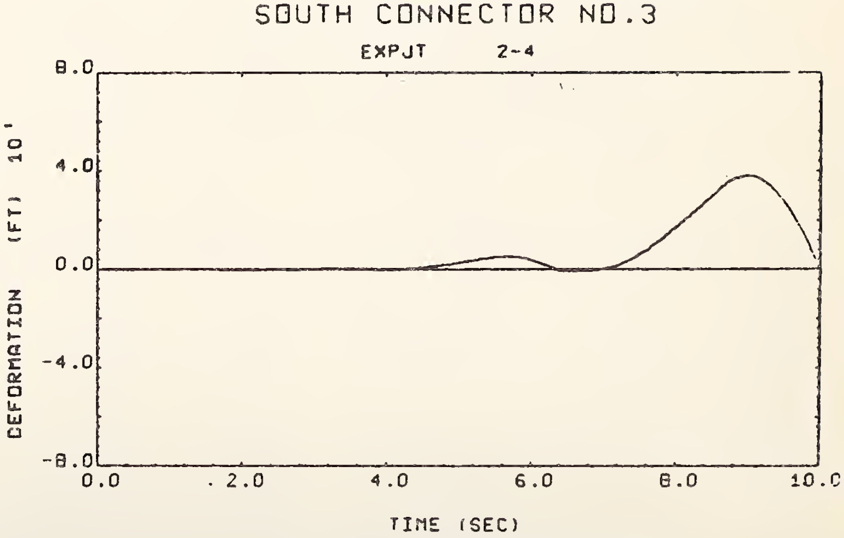


Global Y - Component

Fig. 46 Horizontal Displacements at the Top of Column # 4

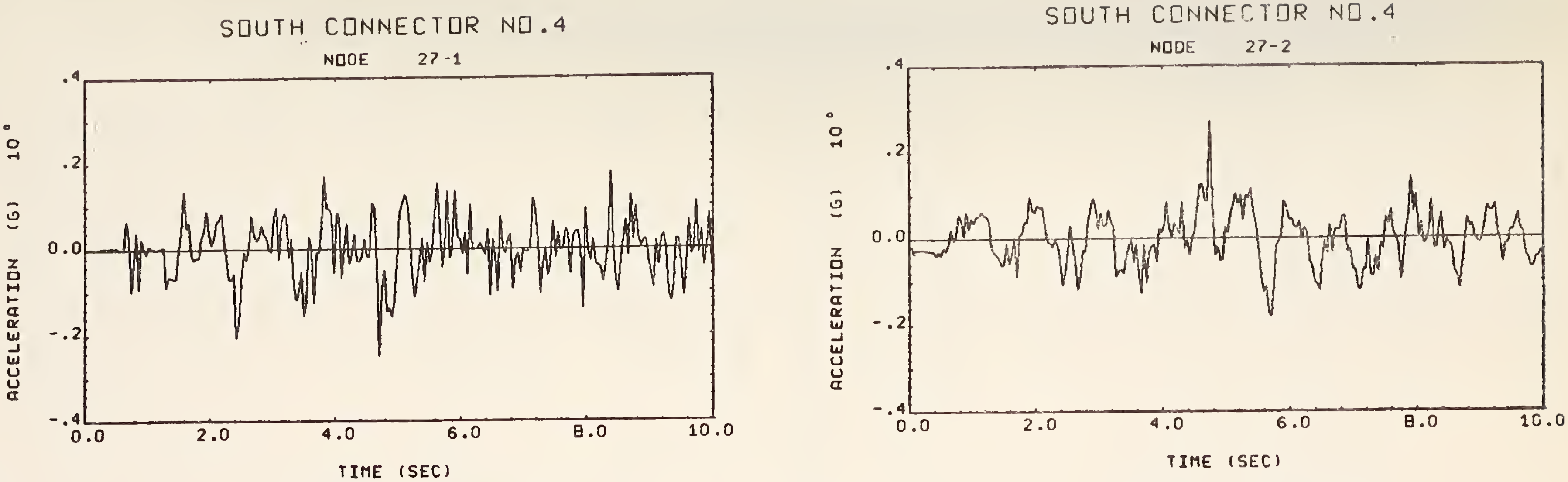


At Inner Edge of the Deck



At Outer Edge of the Deck

Fig. 47 Longitudinal Joint Separations at Expansion Joint # 2



Global X - Component

Global Y - Component

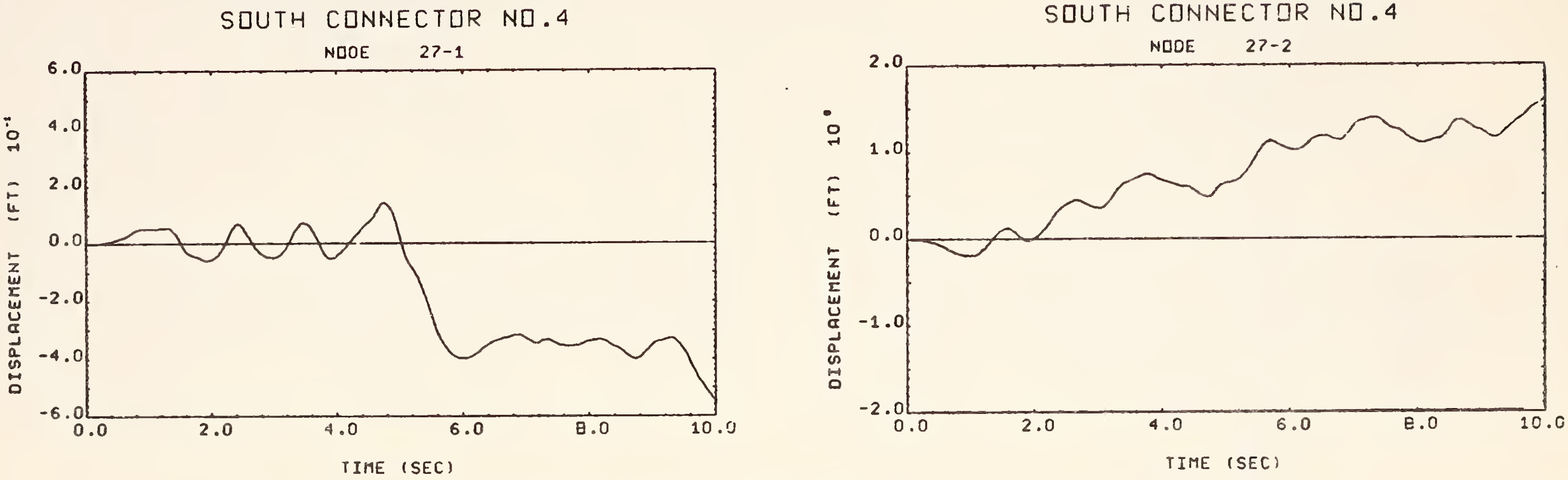
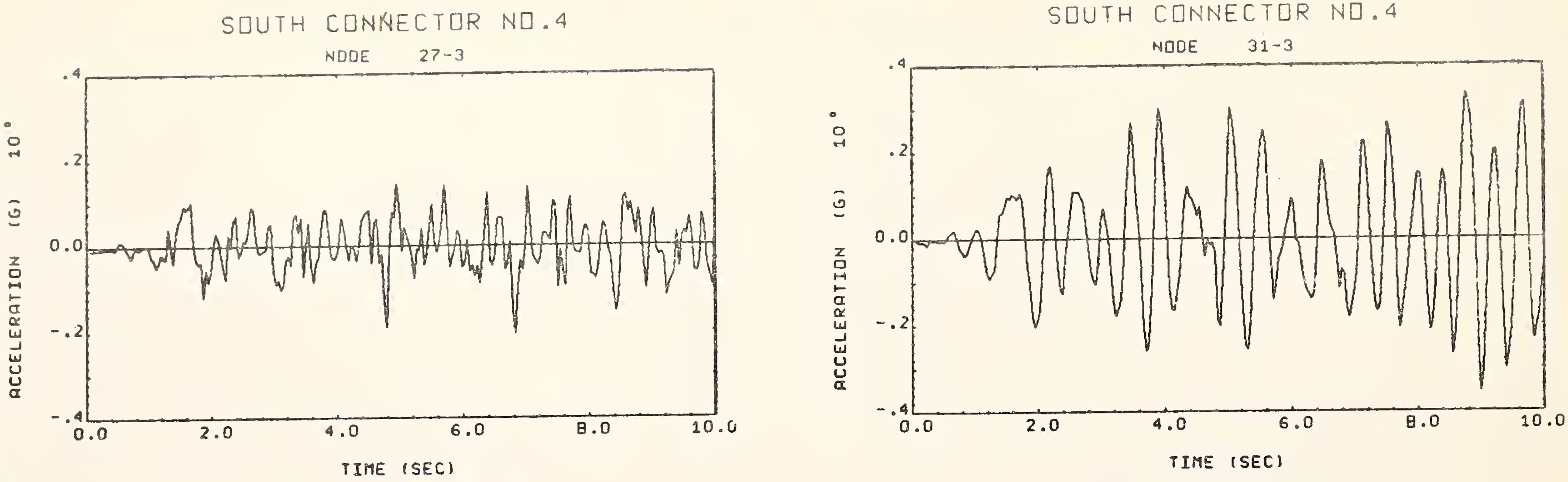


Fig. 48 Horizontal Accelerations and Displacements at the Top of Column # 4



Top of Column # 4

Center of Span # 4

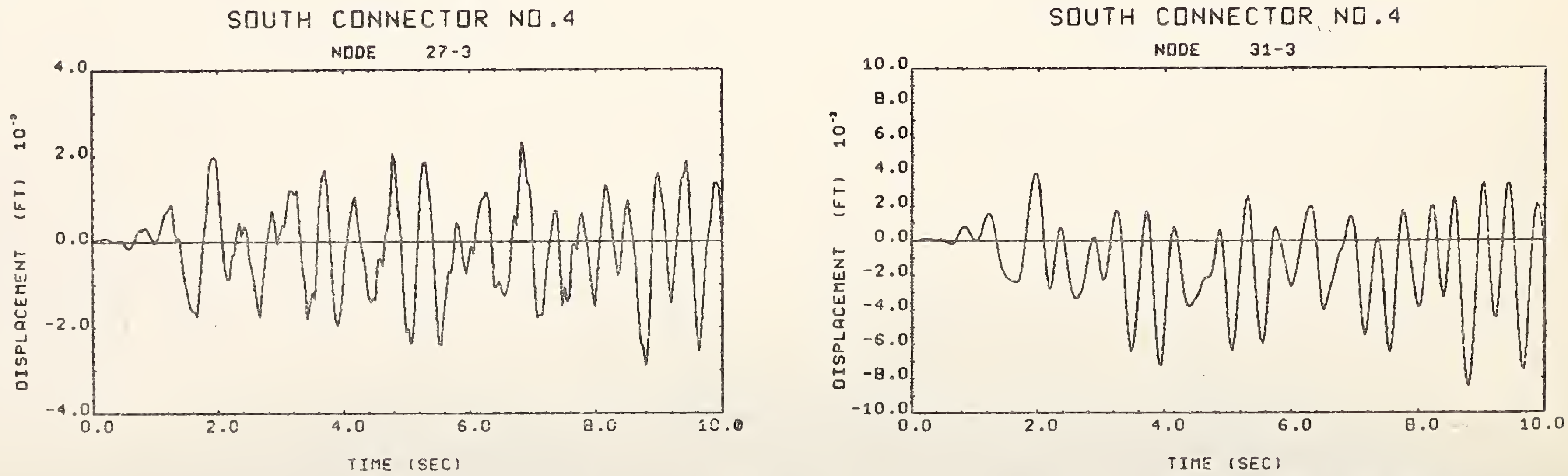
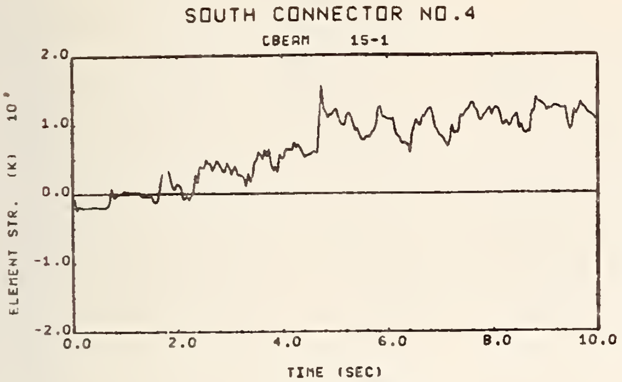
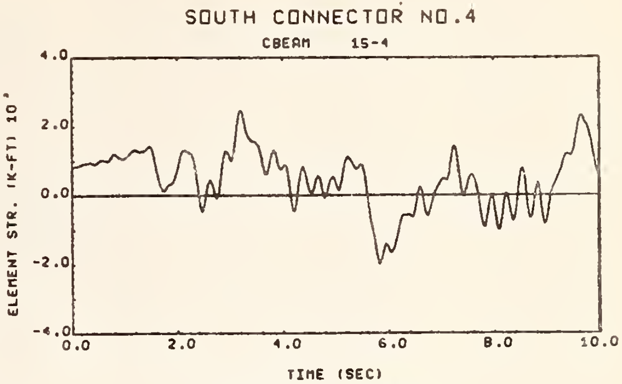


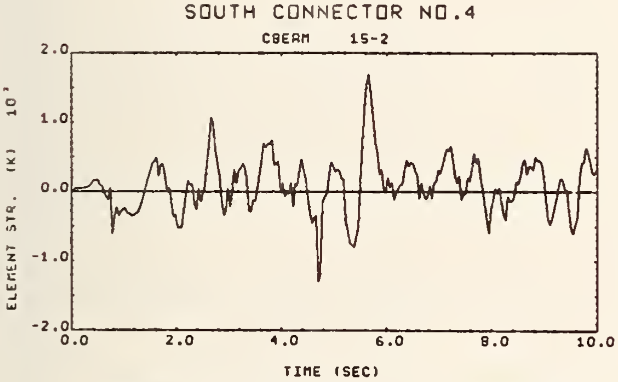
Fig. 49 Vertical Accelerations and Displacements at the Top of Column # 4 and the Center of Span # 4



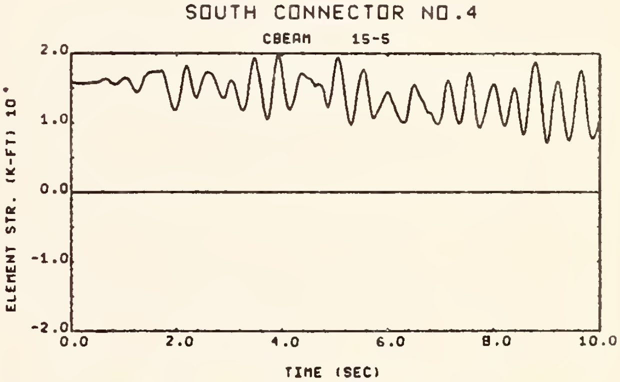
Axial Force



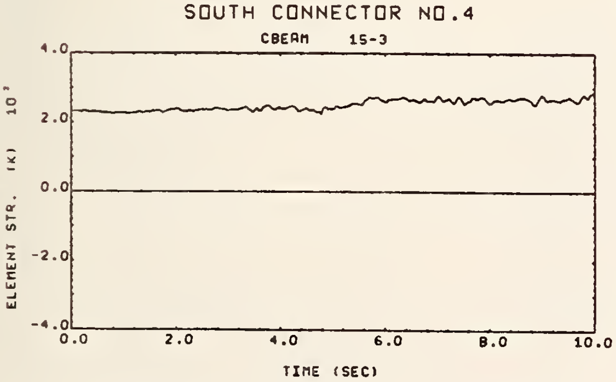
Torsional Moment



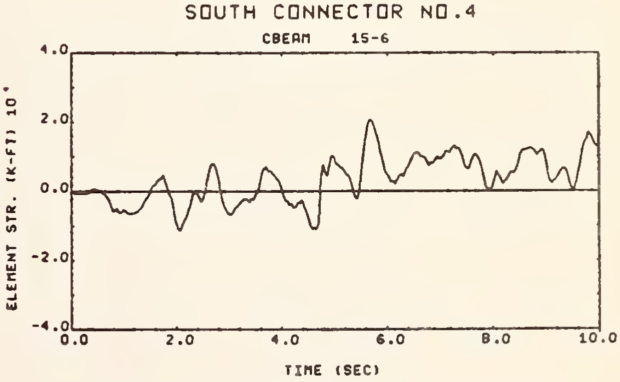
Shear in the Local y - Direction



Moment about the Local y - Axis

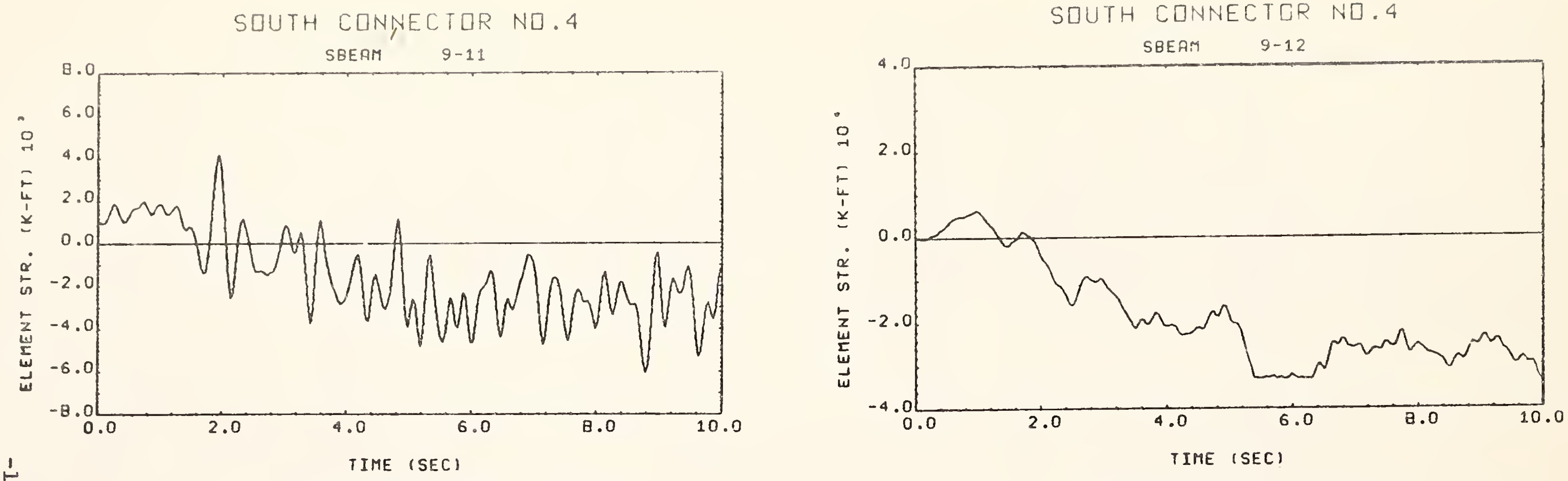


Shear in the Local z - Direction



Moment about the Local z - Axis

Fig. 50 Generalized Forces in the Girder at the Center of Span # 4



About the Local y - Axis

About the Local z - Axis

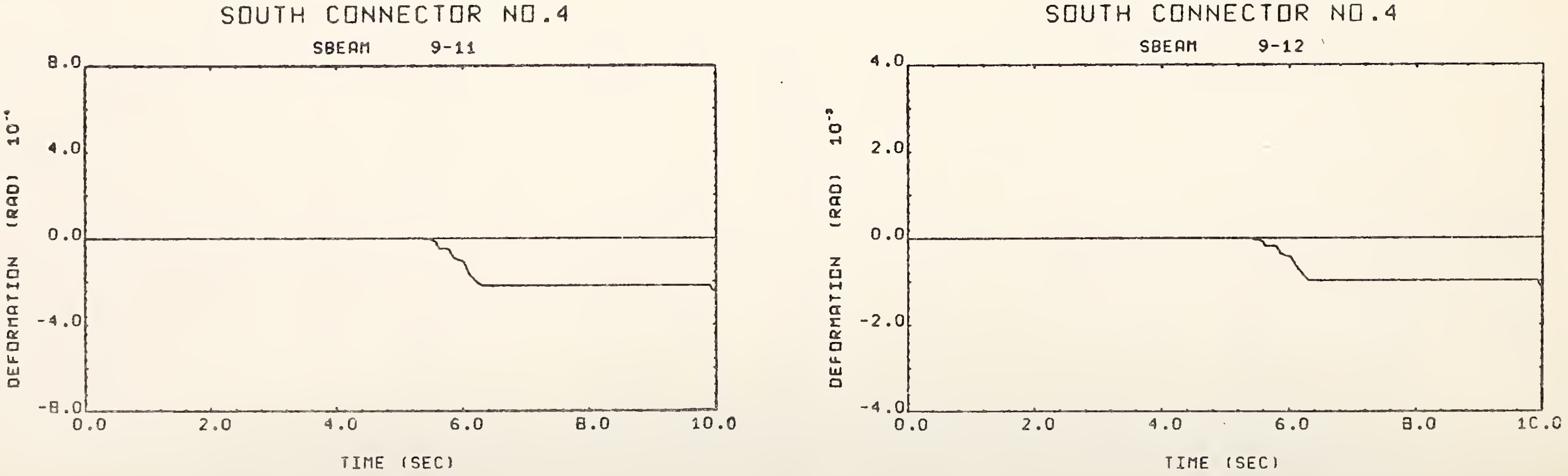
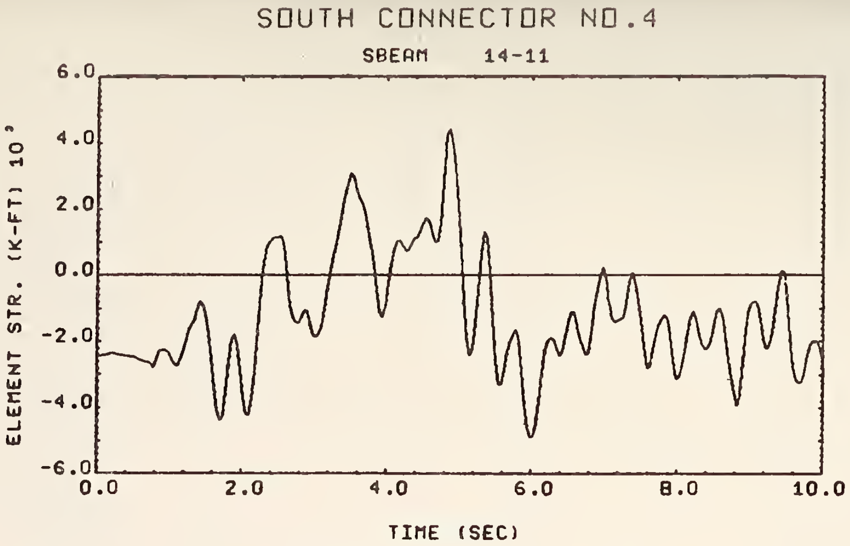
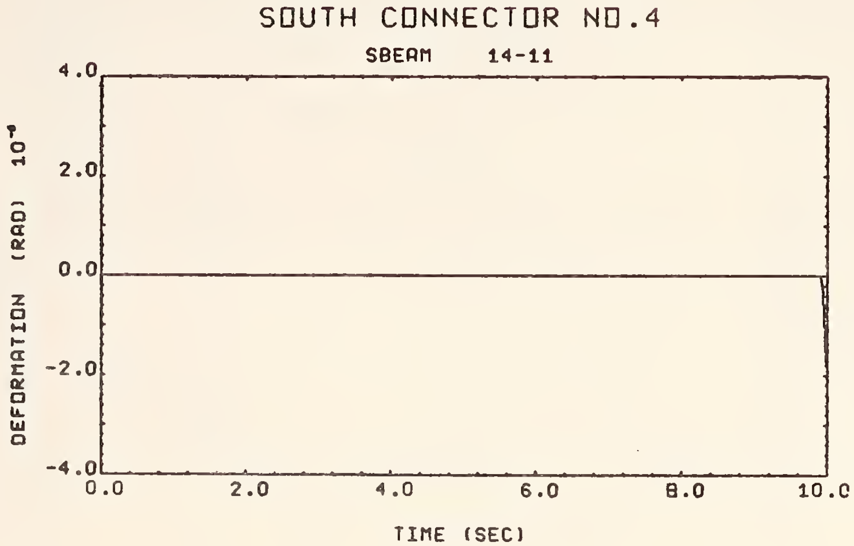
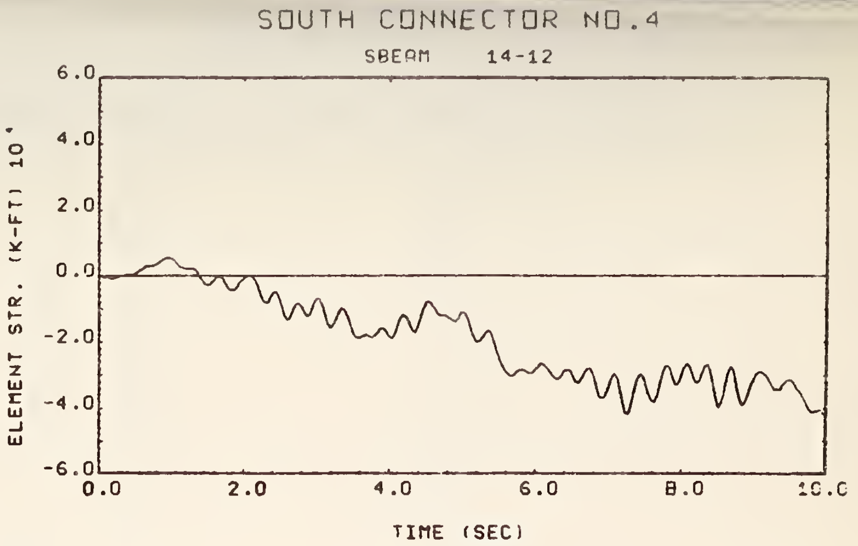


Fig. 51 Bending Moments and the Corresponding Plastic Rotations at the Base of Column # 3



About the Local y - Axis





About the Local z - Axis

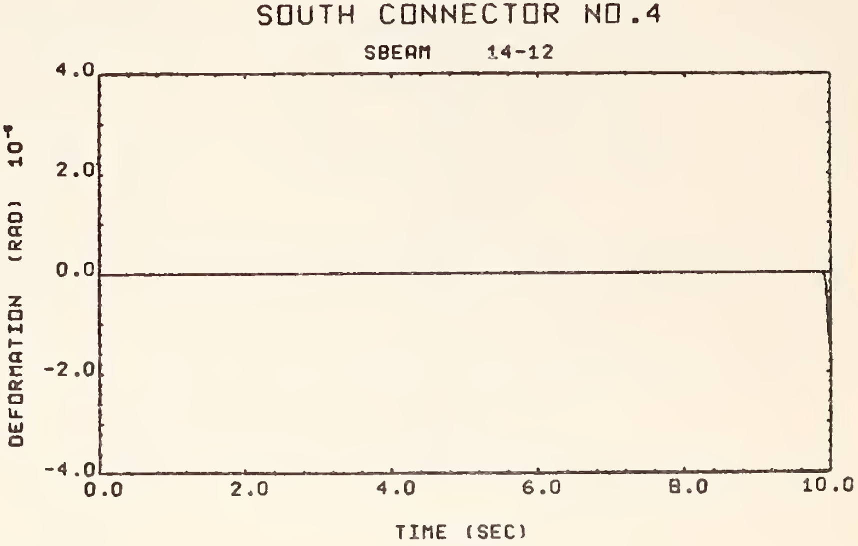
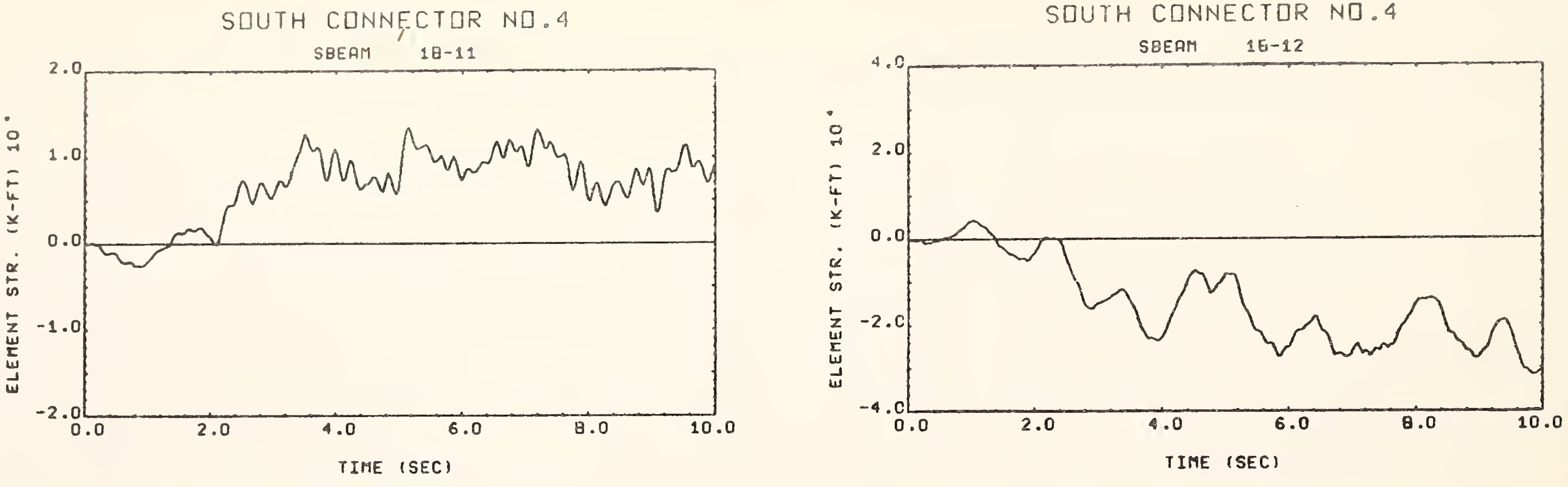


Fig. 52 Bending Moments and the Corresponding Plastic Rotations at the Base of Column # 4



About the Local y - Axis

About the Local z - Axis

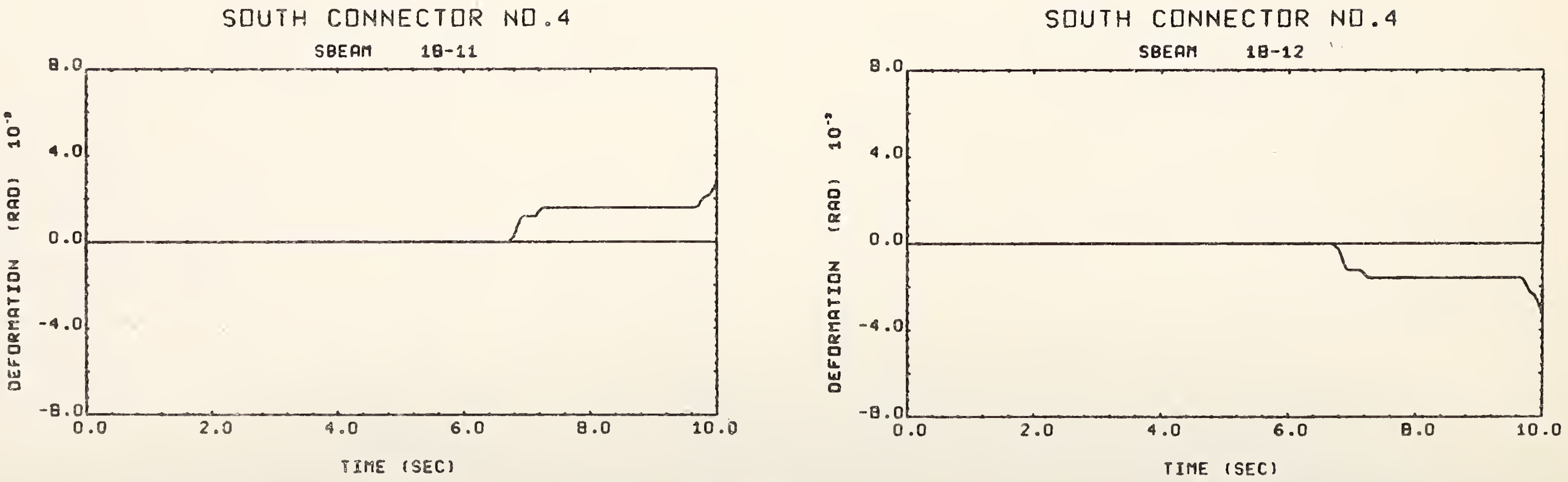
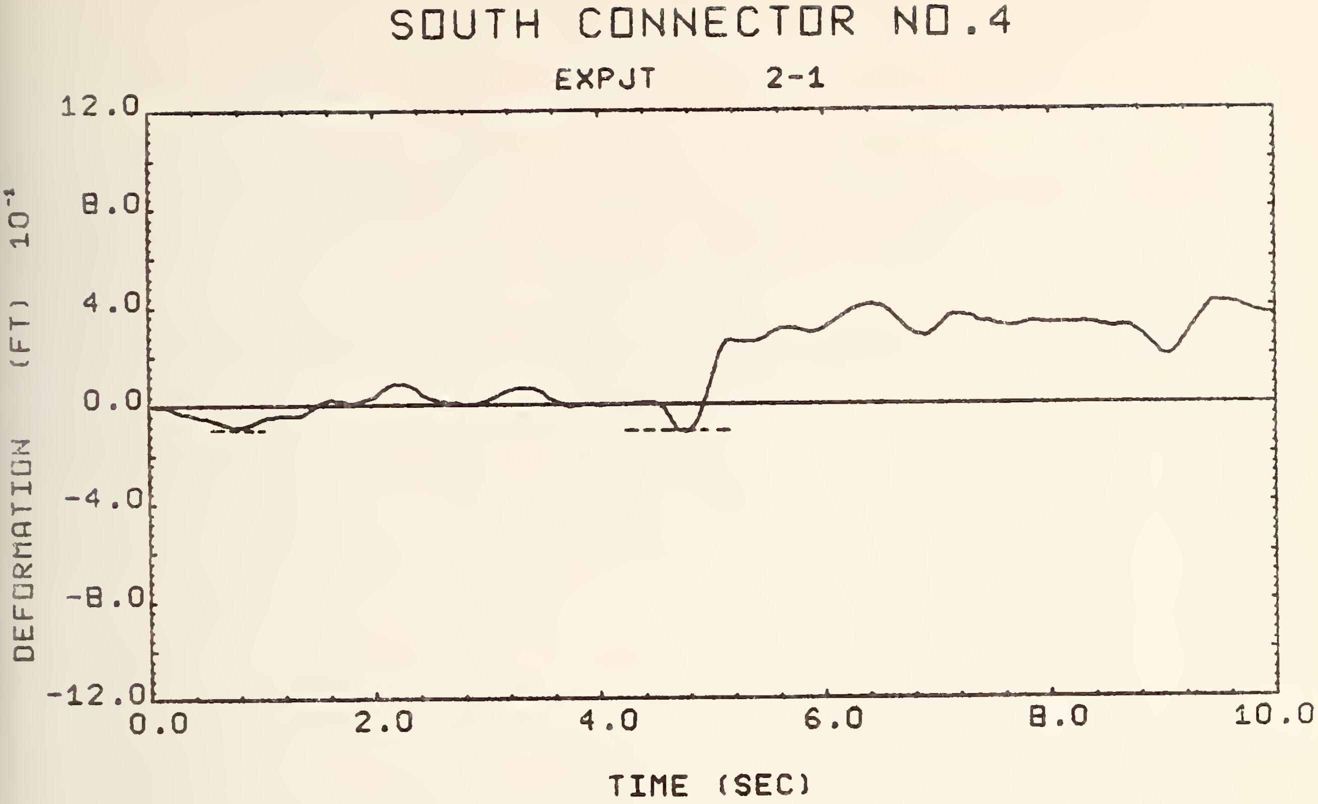
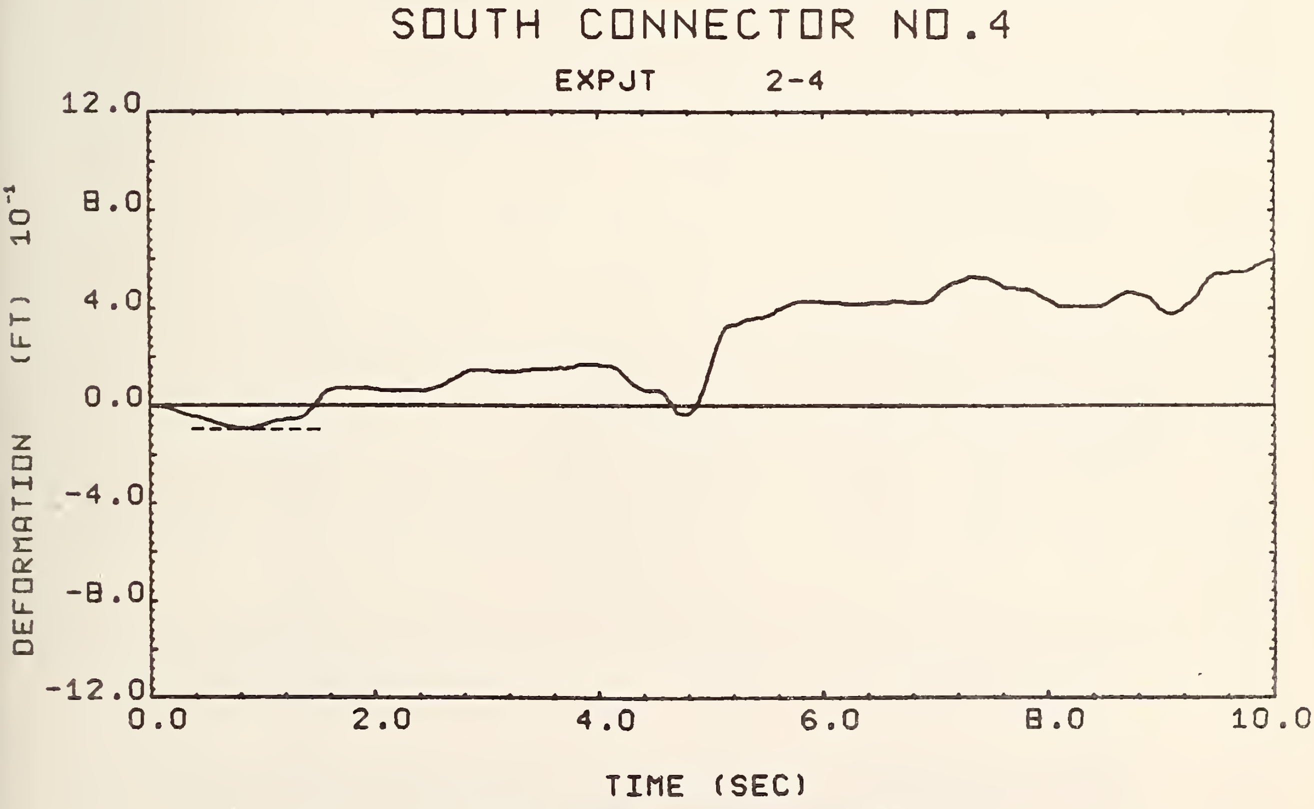


Fig. 53 Bending Moments and the Corresponding Plastic Rotations at the Base of Column # 5

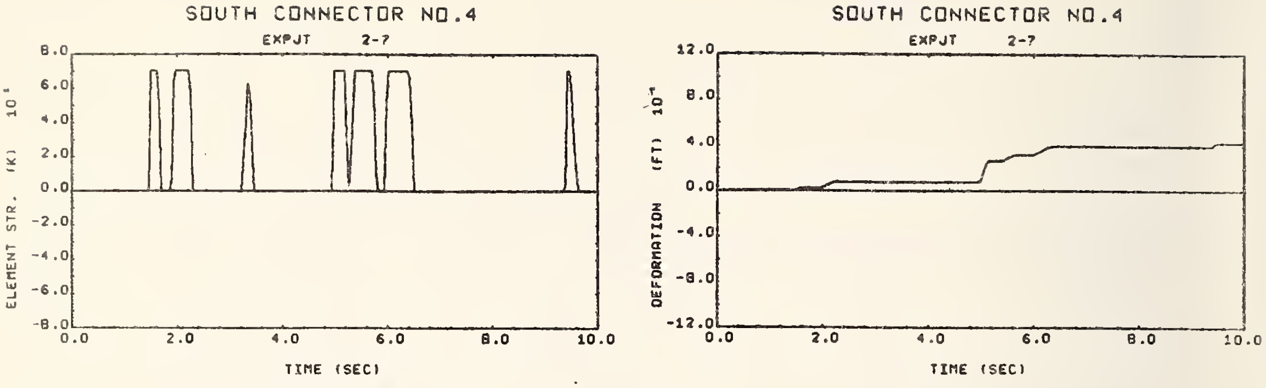


At Inner Edge of the Deck

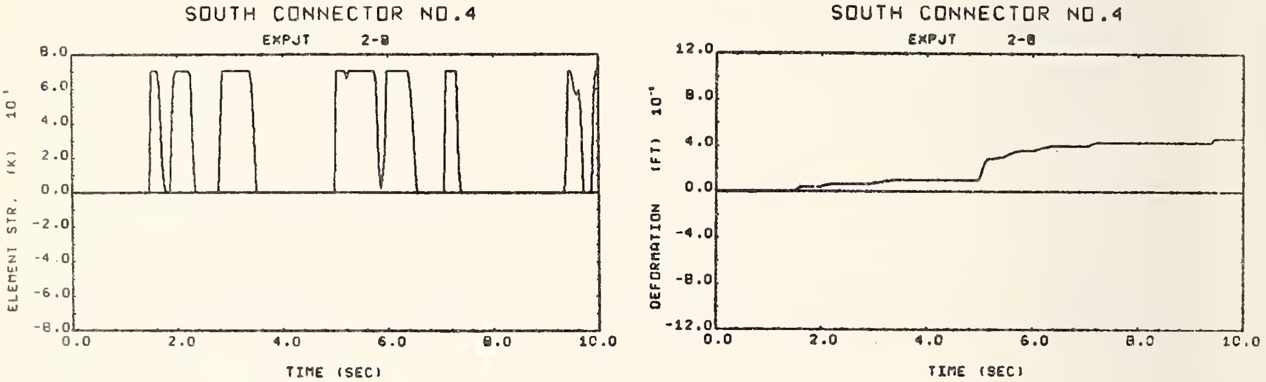


At Outer Edge of the Deck

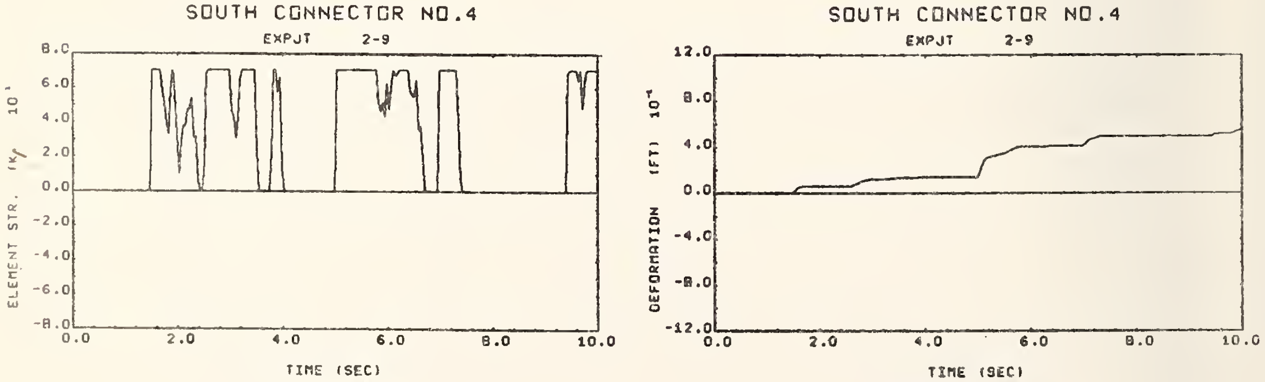
Fig. 54 Longitudinal Joint Separations at Expansion Joint # 2



Tie Bar # 1 Tie Bar # 1

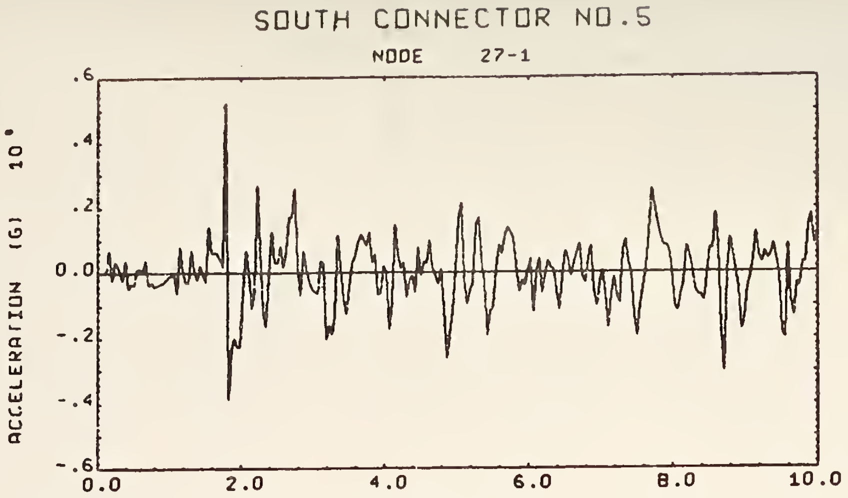


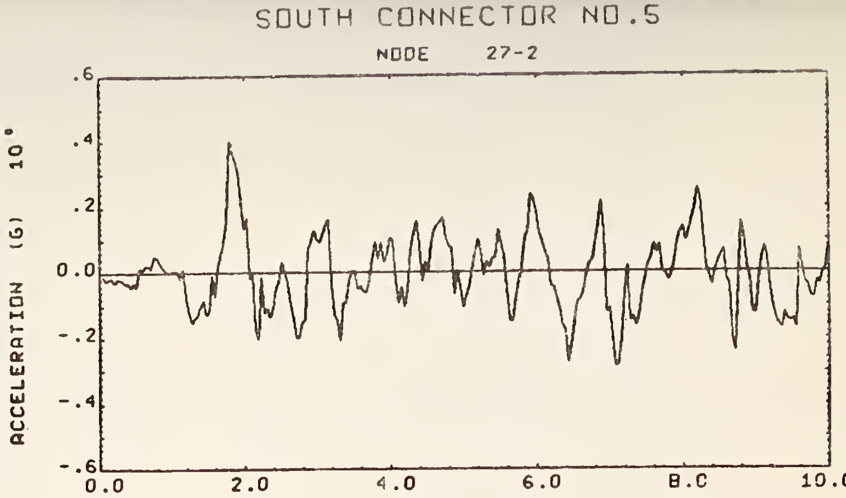
Tie Bar # 2 Tie Bar # 2



Tie Bar # 3 Tie Bar # 3

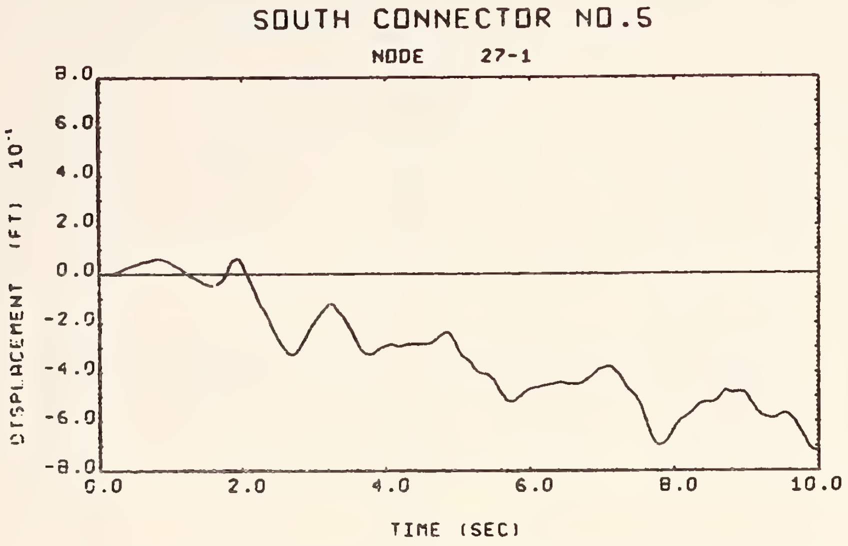
Fig. 55 Longitudinal Tie Bar Forces and the Corresponding Plastic Elongations at Expansion Joint # 2





Global X - Component

Global Y - Component



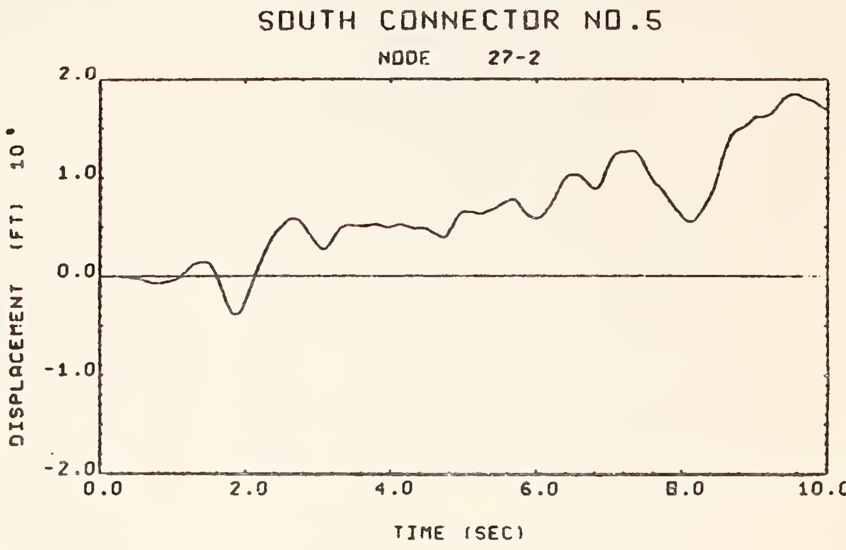
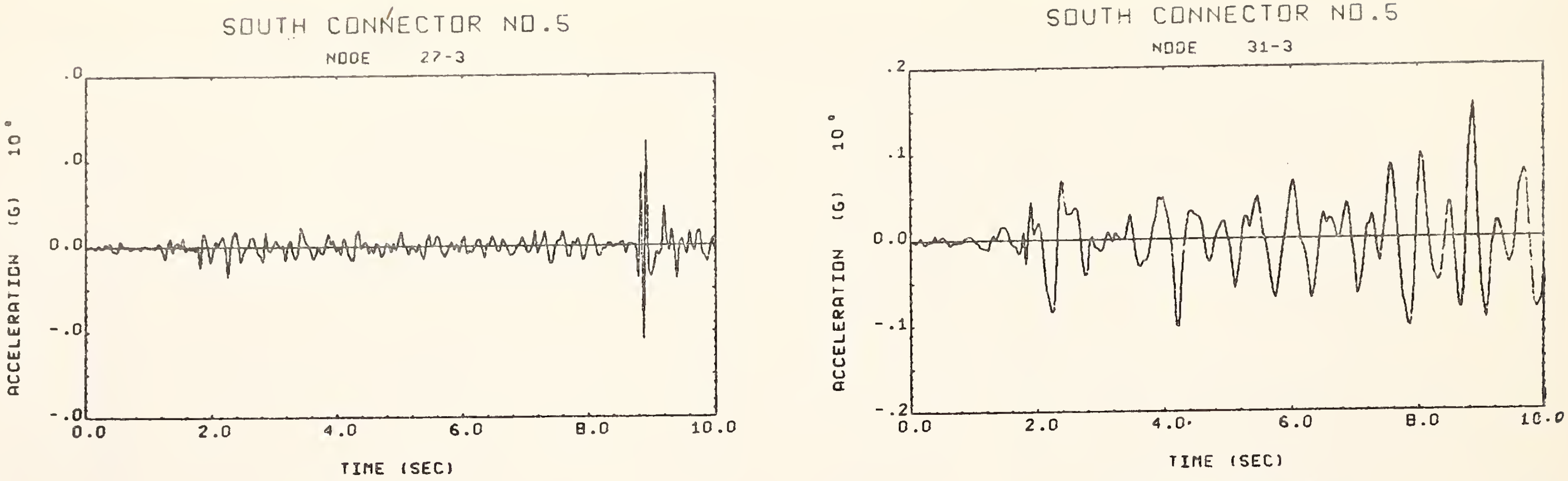


Fig. 56 Horizontal Accelerations and Displacements at the Top of Column # 4



Top of Column # 4

Center of Span # 4

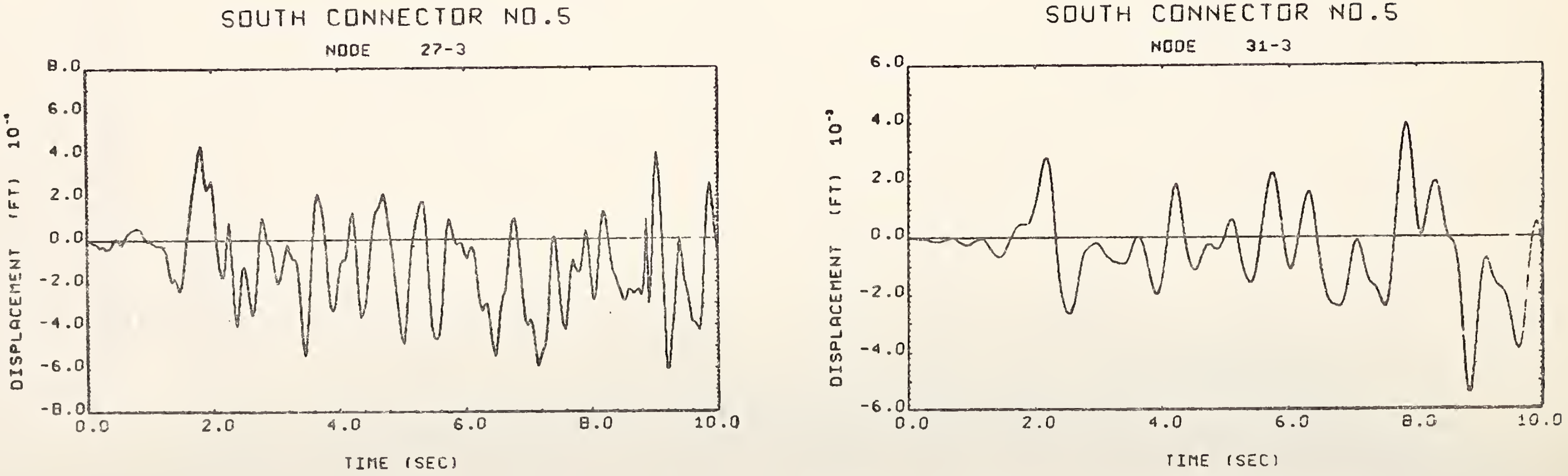
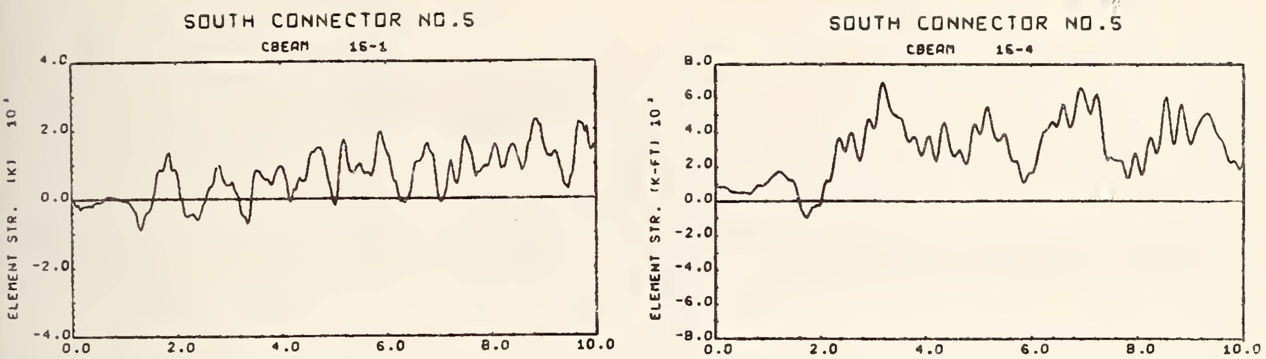
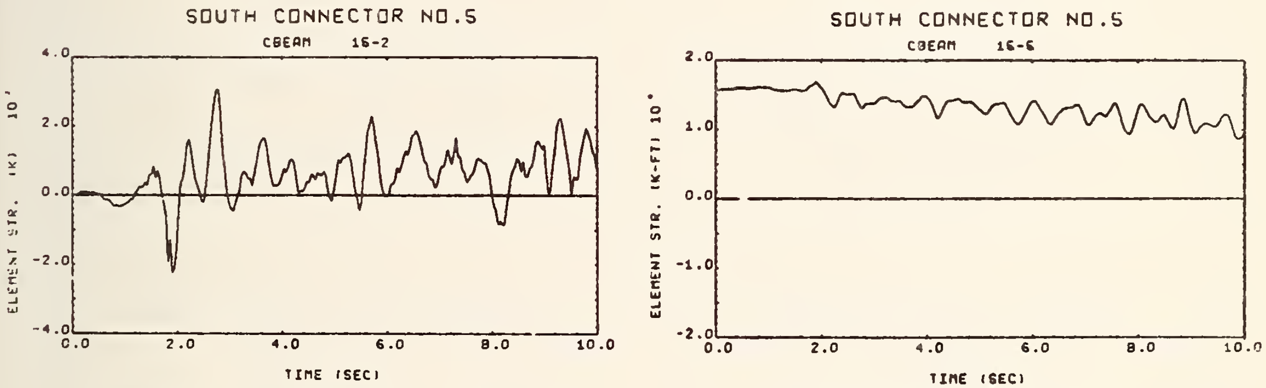


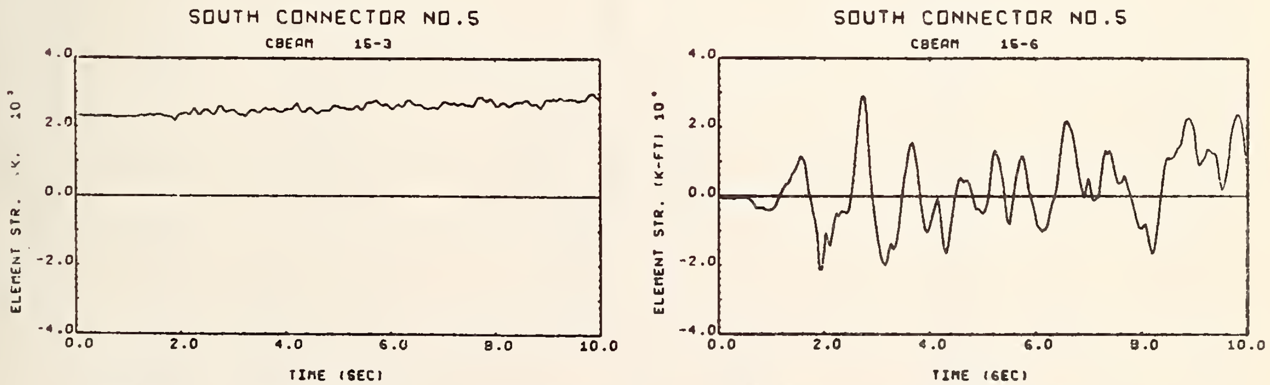
Fig. 57 Vertical Accelerations and Displacements at the Top of Column # 4 and the Center of Span # 4



Axial Force **Torsional Moment**

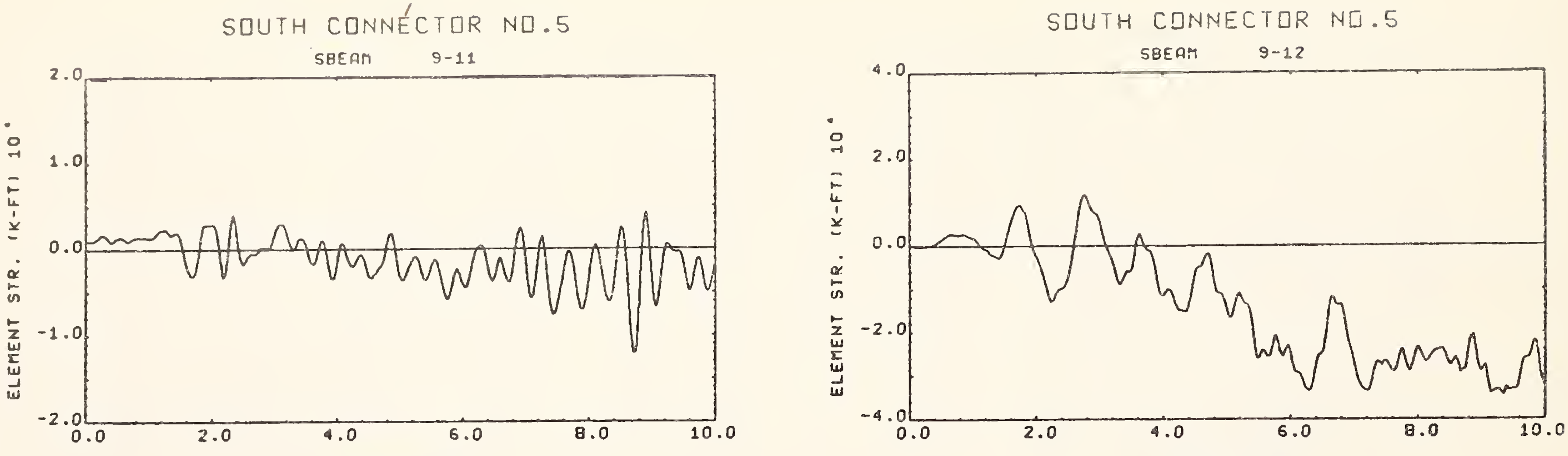


Shear in the Local y - Direction **Moment about the Local y - Axis**



Shear in the Local z - Direction **Moment about the Local z - Axis**

Fig. 58 Generalized Forces in the Girder at the Center of Span # 4



About the Local y - Axis

About the Local z - Axis

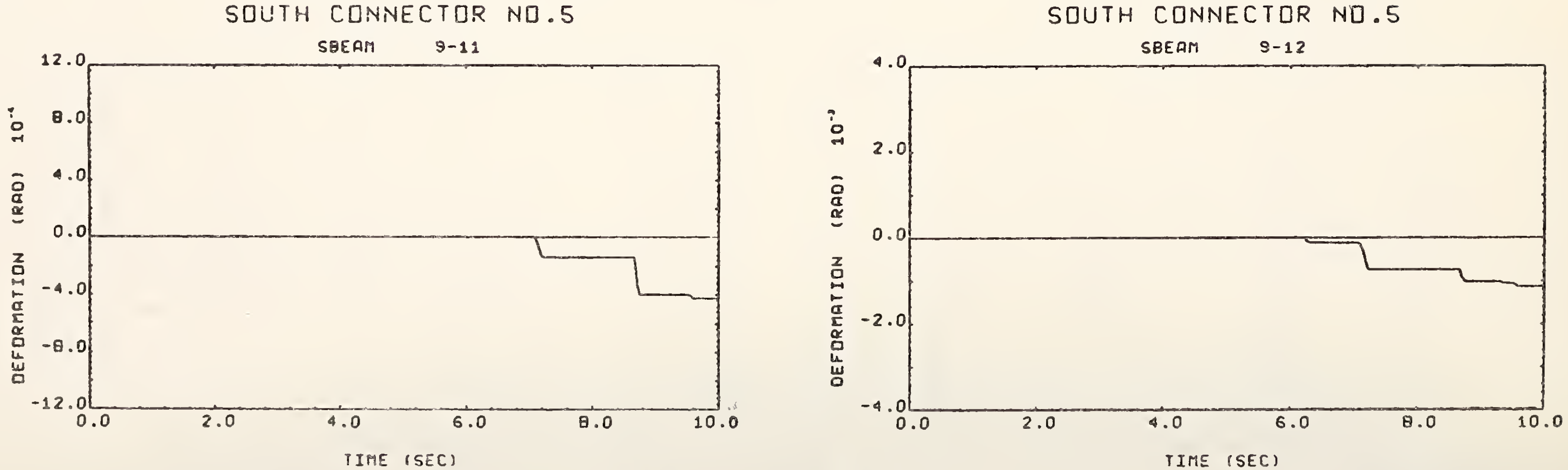
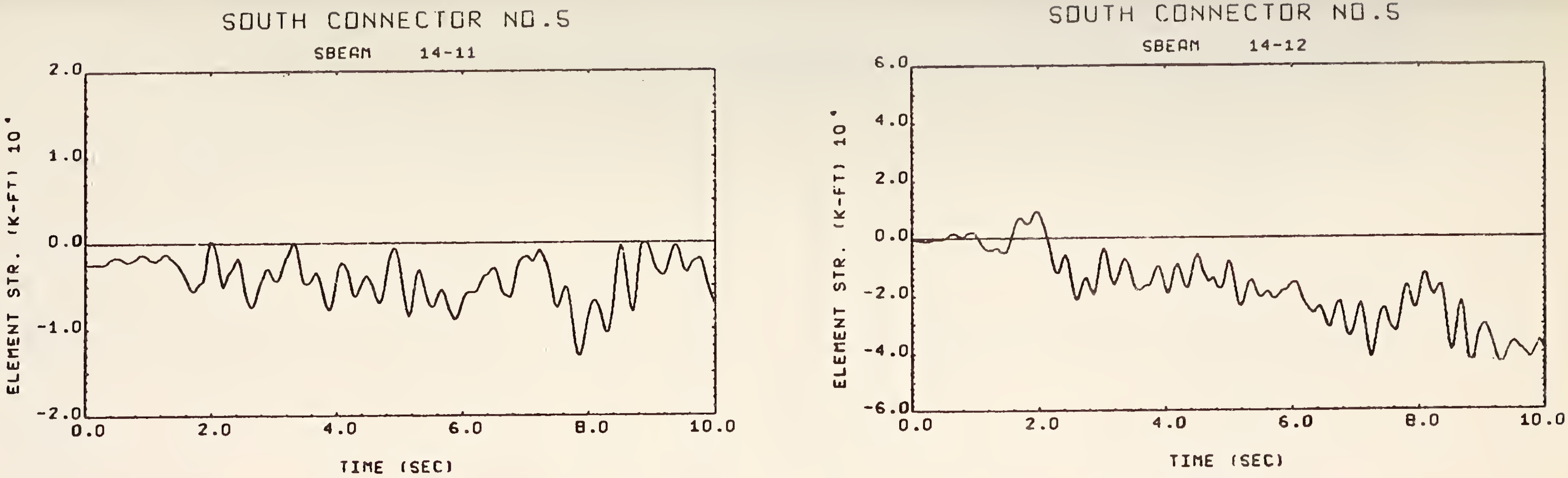


Fig. 59 Bending Moments and the Corresponding Plastic Rotations at the Base of Column # 3



About the Local y - Axis

About the Local z - Axis

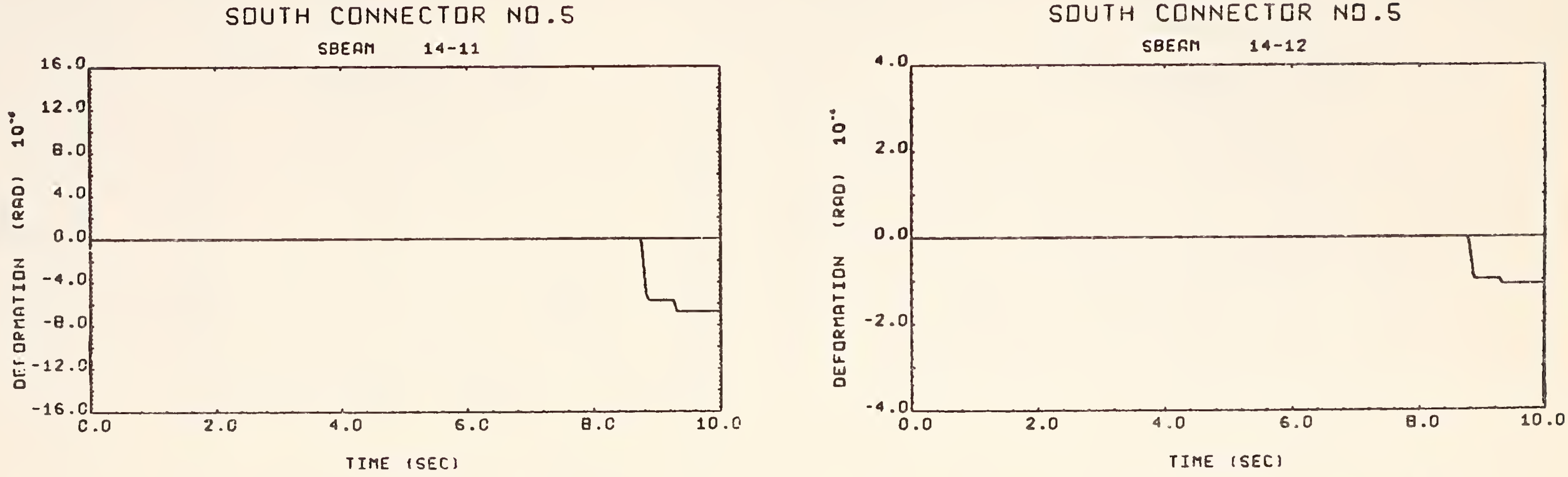
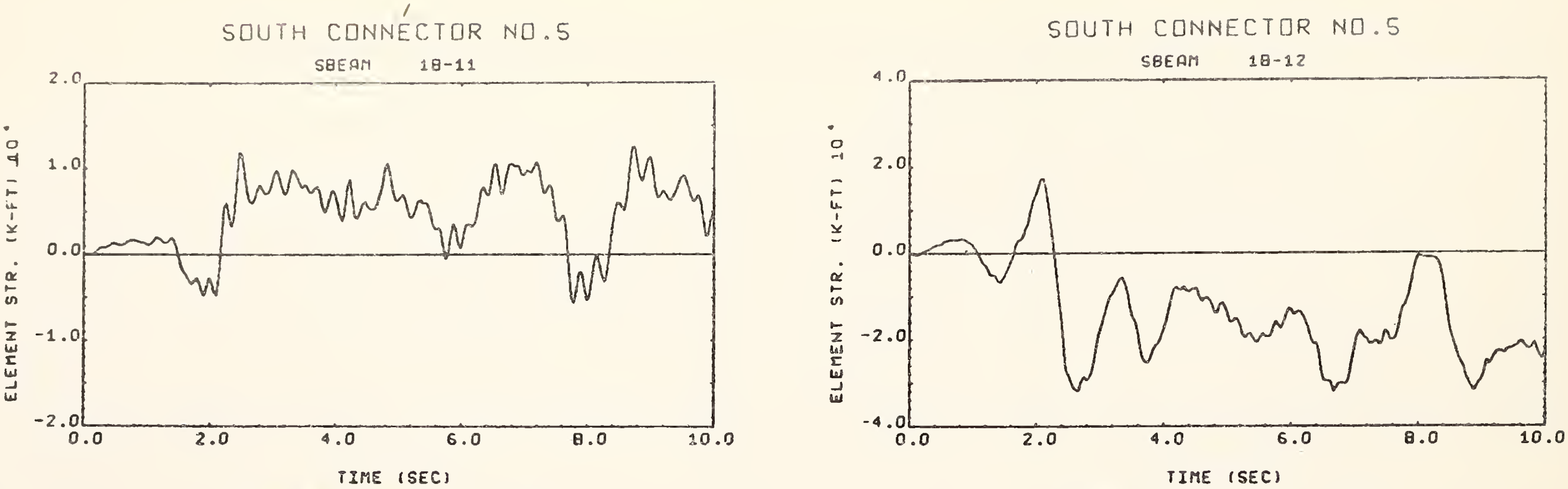


Fig. 60 Bending Moments and the Corresponding Plastic Rotations at the Base of Column # 4



About the Local y - Axis

About the Local z - Axis

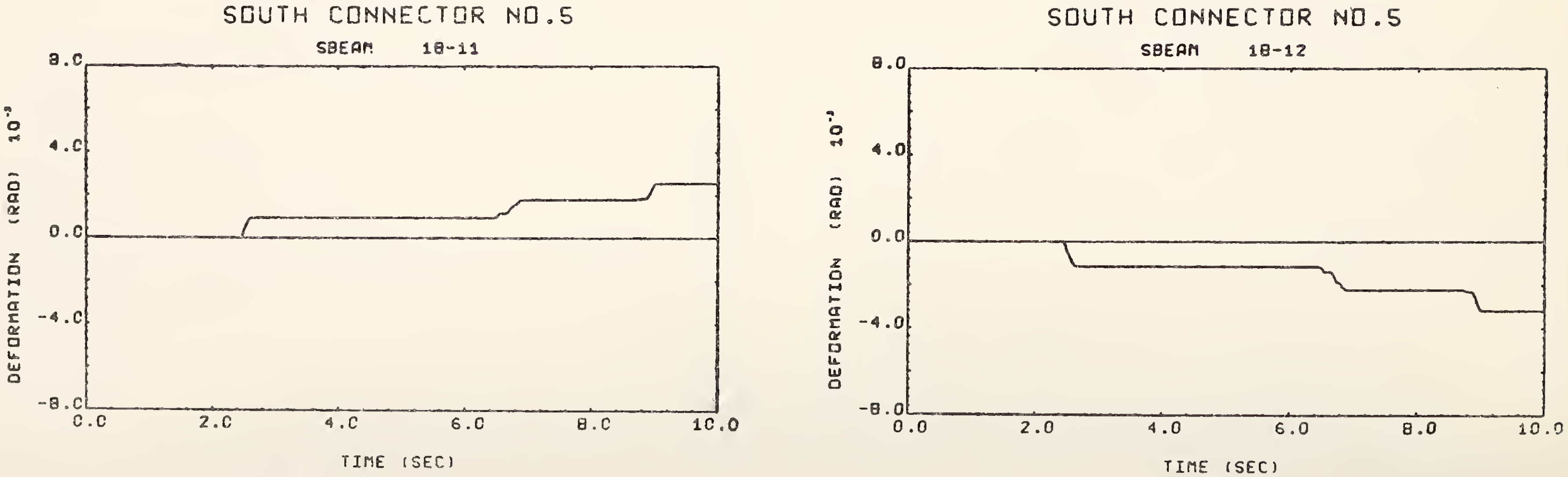
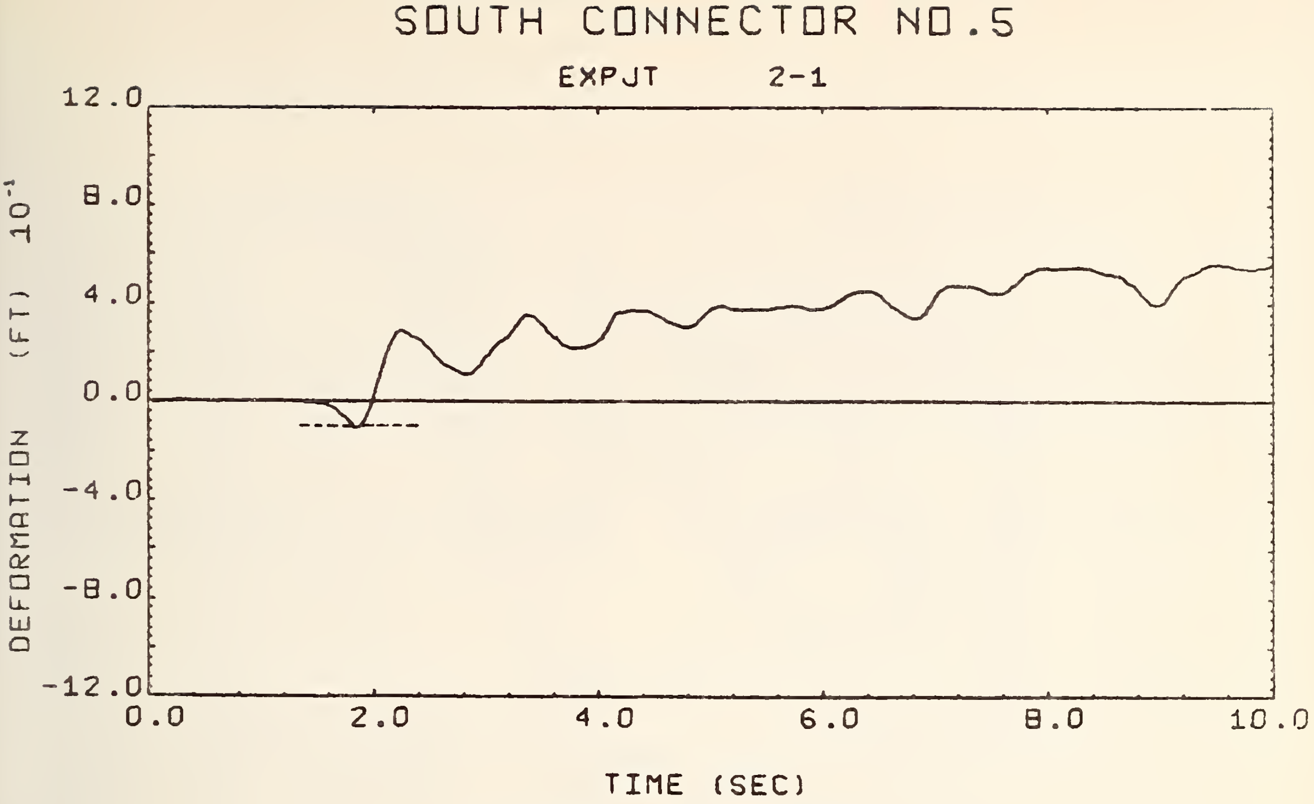
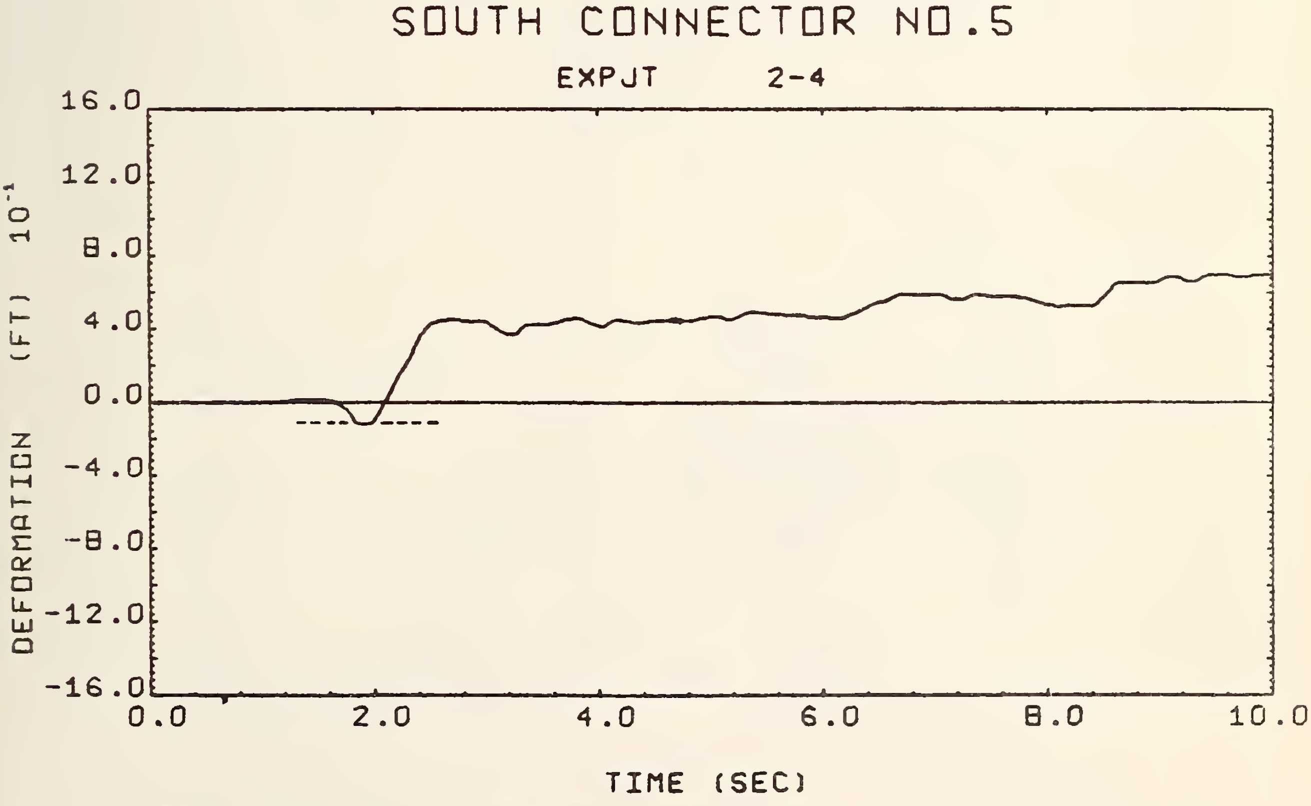


Fig. 61 Bending Moments and the Corresponding Plastic Rotations at the Base of Column # 5

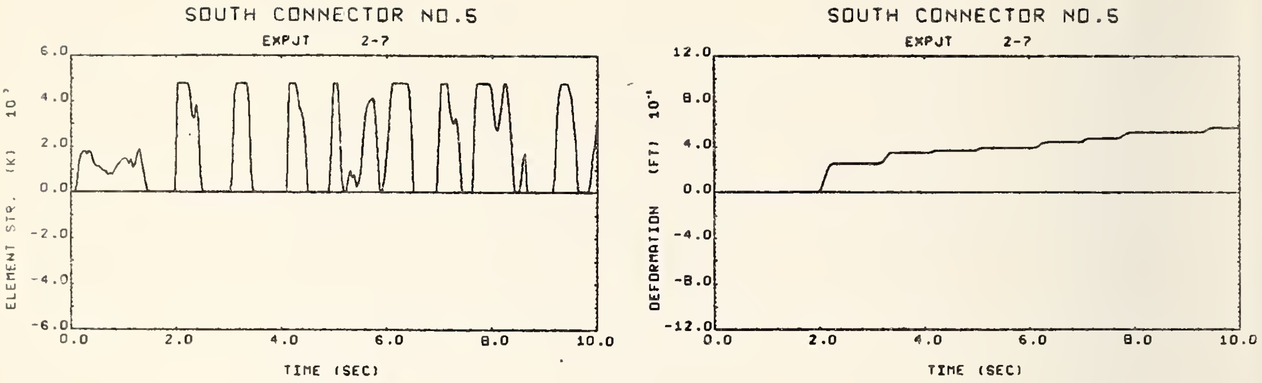


At Inner Edge of the Deck

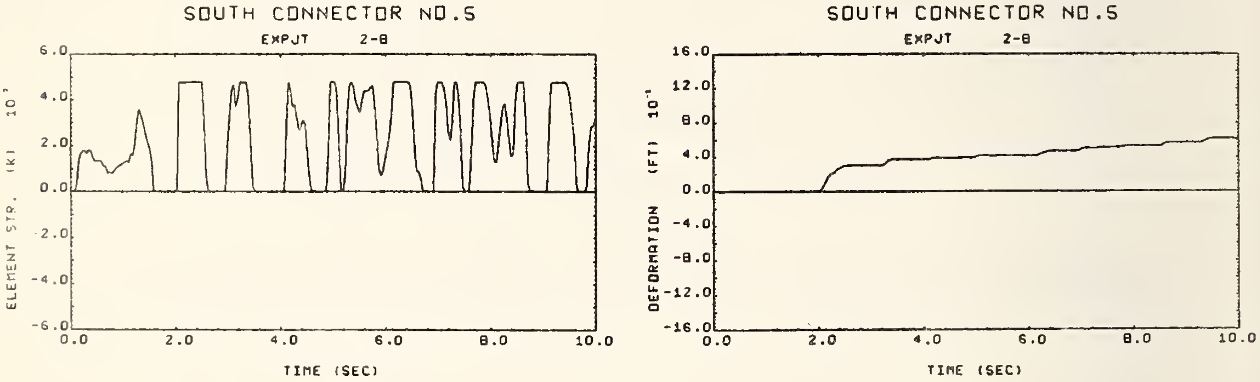


At Outer Edge of the Deck

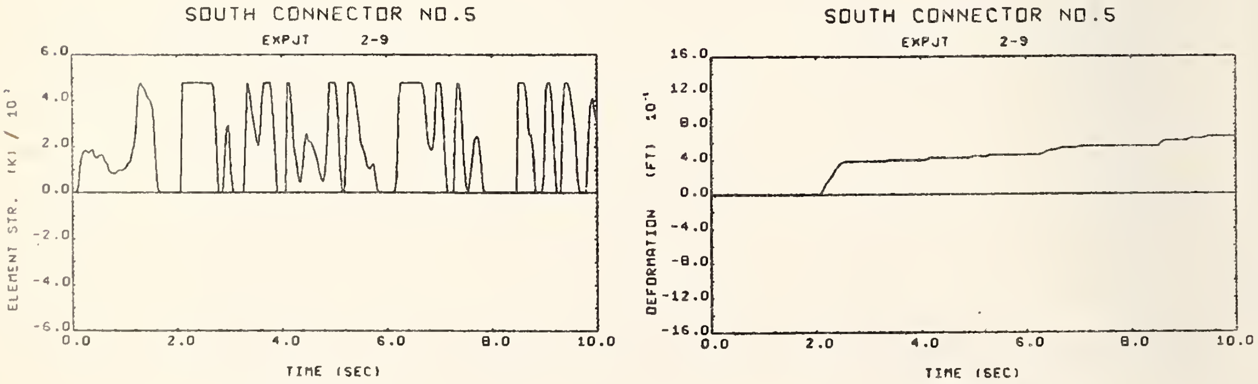
Fig. 62 Longitudinal Joint Separations at Expansion Joint # 2



Tie Bar # 1 Tie Bar # 1



Tie Bar # 2 Tie Bar # 2



Tie Bar # 3 Tie Bar # 3

Fig. 63 Longitudinal Tie Bar Forces and the Corresponding Plastic Elongations at Expansion Joint # 2

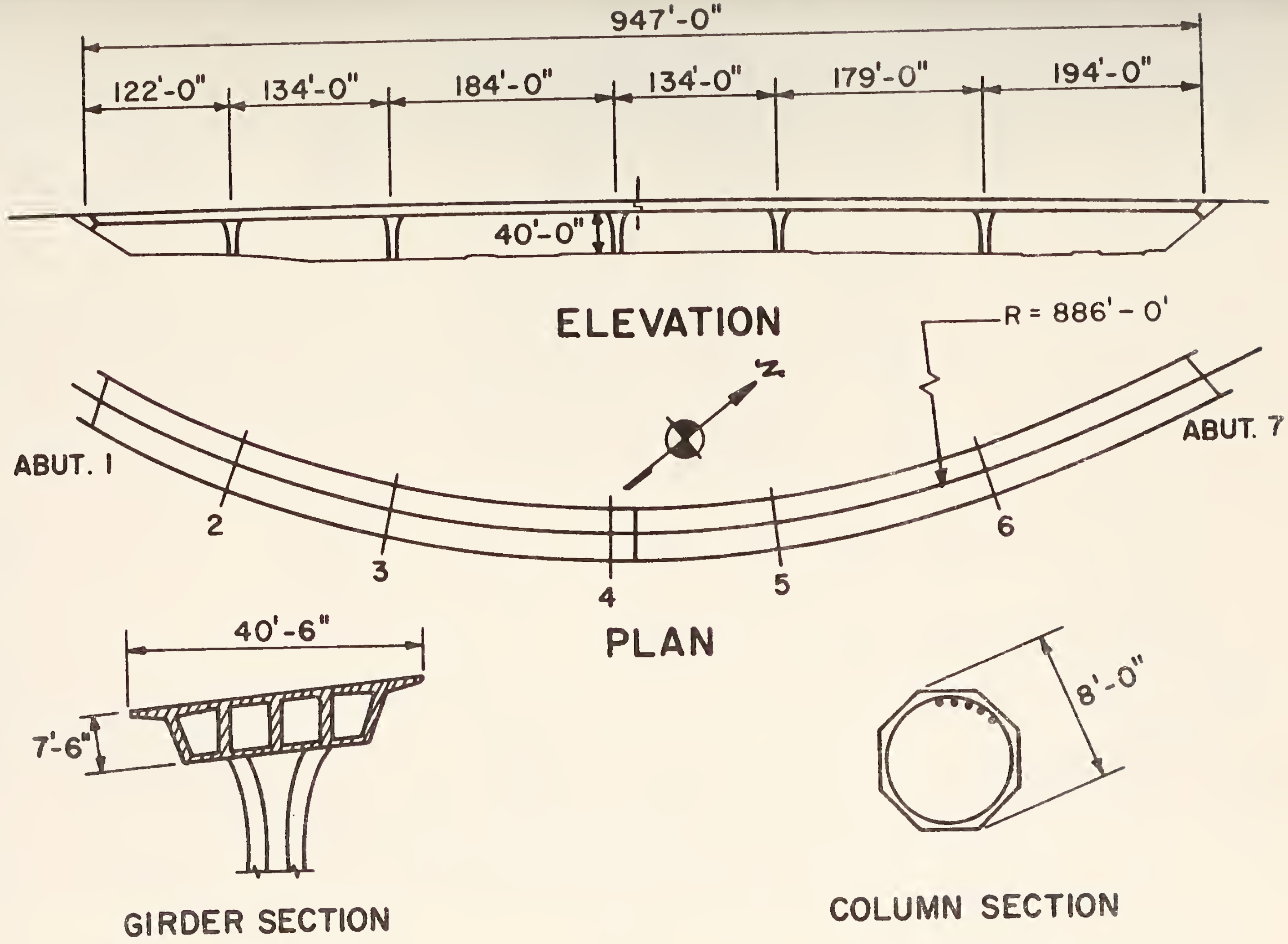
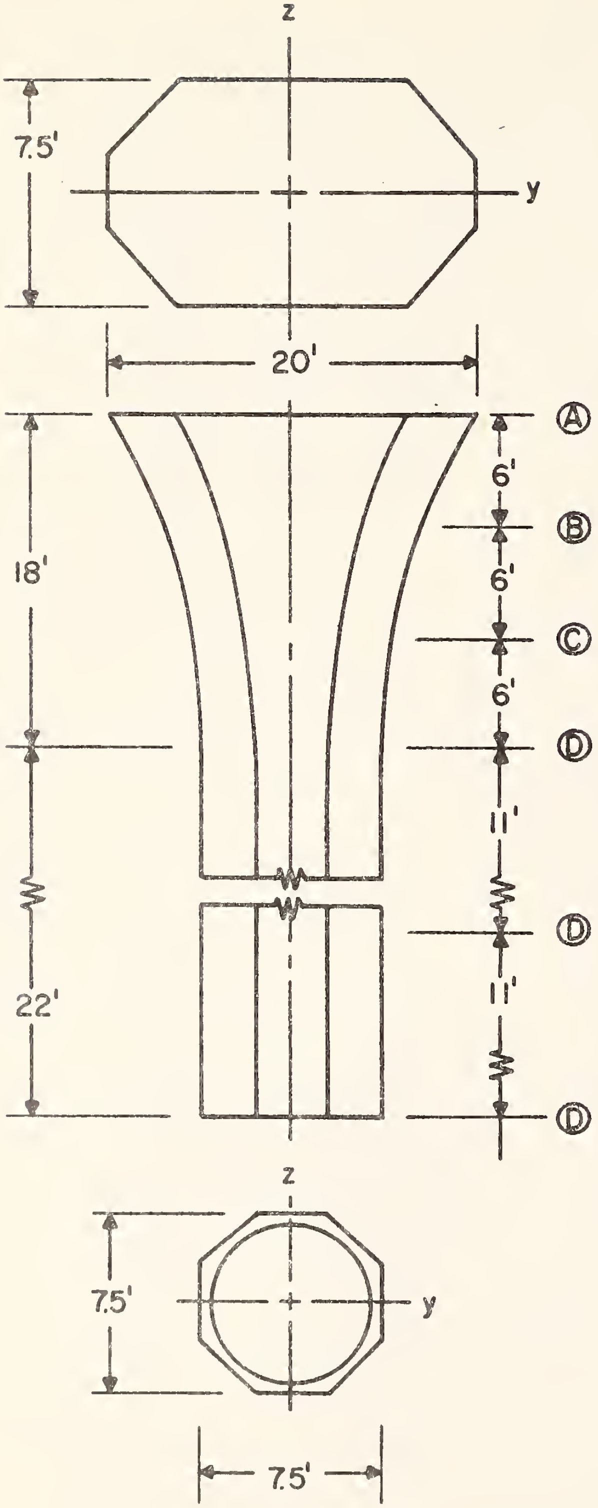


Fig. 64 The Structural System of the Curved Figueroa Street Undercrossing Connector



COLUMN CROSS-SECTION PROPERTIES

CROSS-SECTION	A (ft^2)	I_x (ft^4)	I_y (ft^4)	I_z (ft^4)
Ⓐ	114.2	1728.	758.	1436.
Ⓑ	75.1	779.	373.	545.
Ⓒ	48.3	333.	182.	191.
Ⓓ	45.6	295.	164.	164.

Fig. 65 Column Cross-Sections of the Curved Figueroa Street Undercrossing Connector

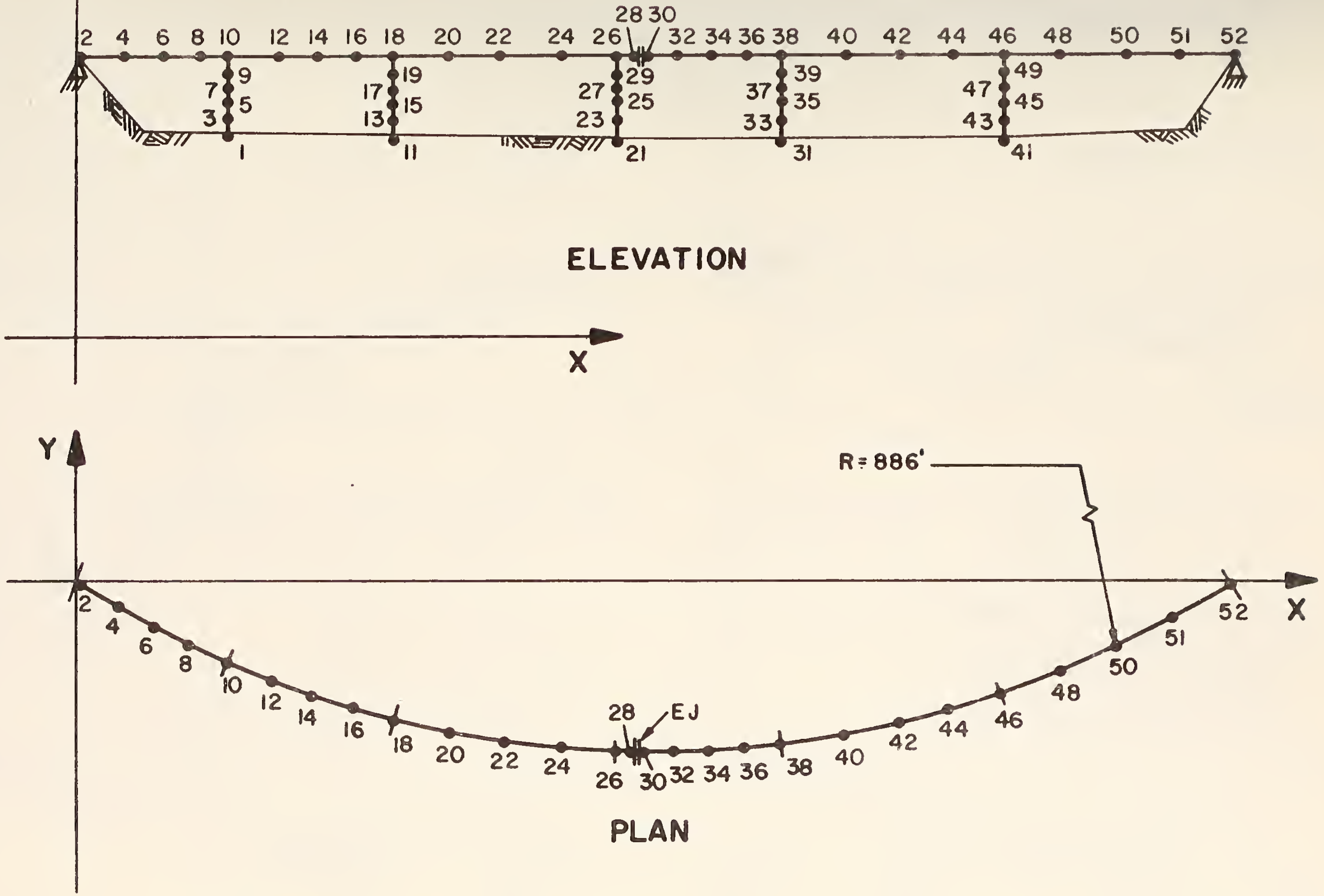


Fig. 66 Lumped Parameter System of the Curved Figueroa Street Undercrossing Connector

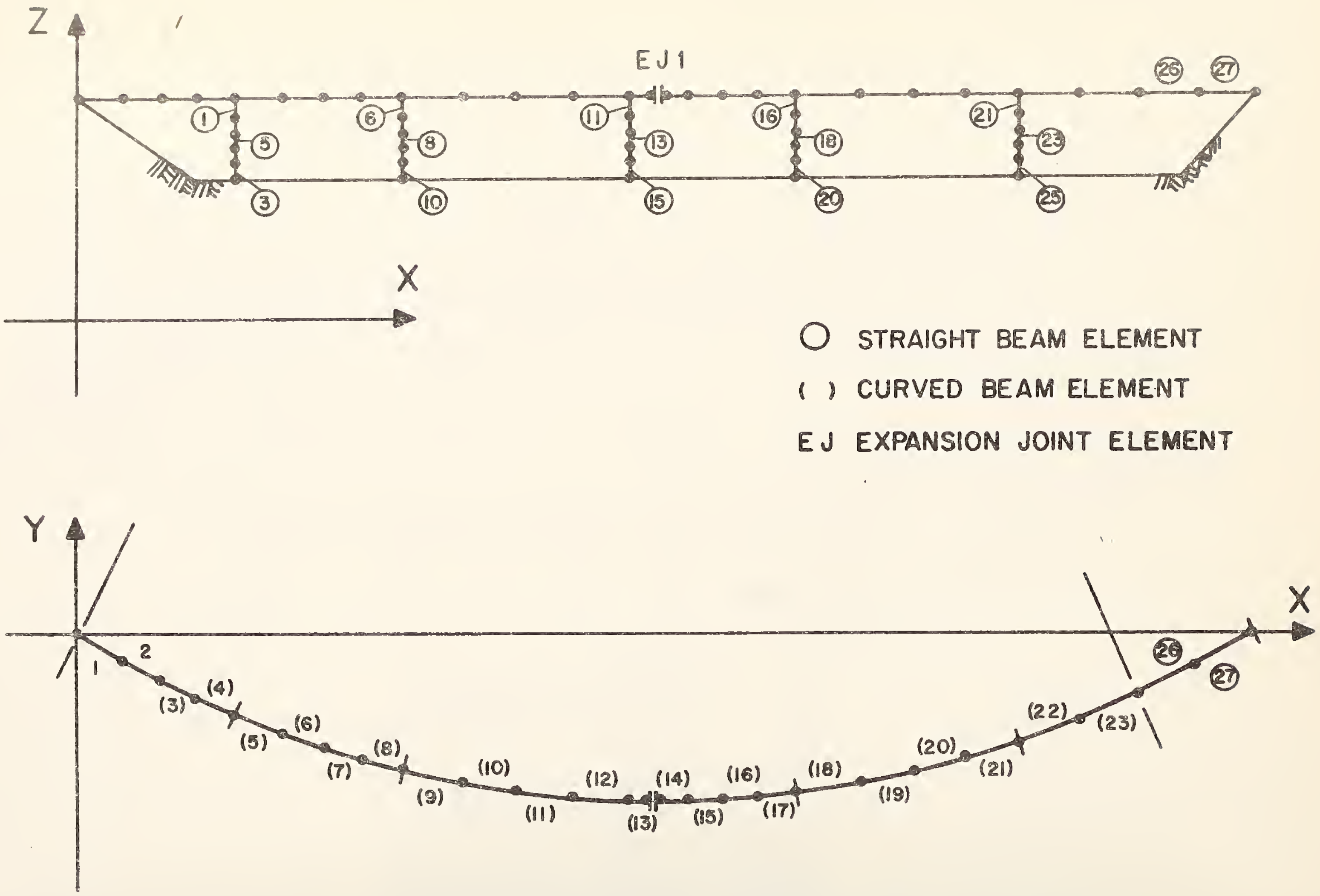


Fig. 67 Finite Element Model of the Curved Figueroa Street U. C. Connector

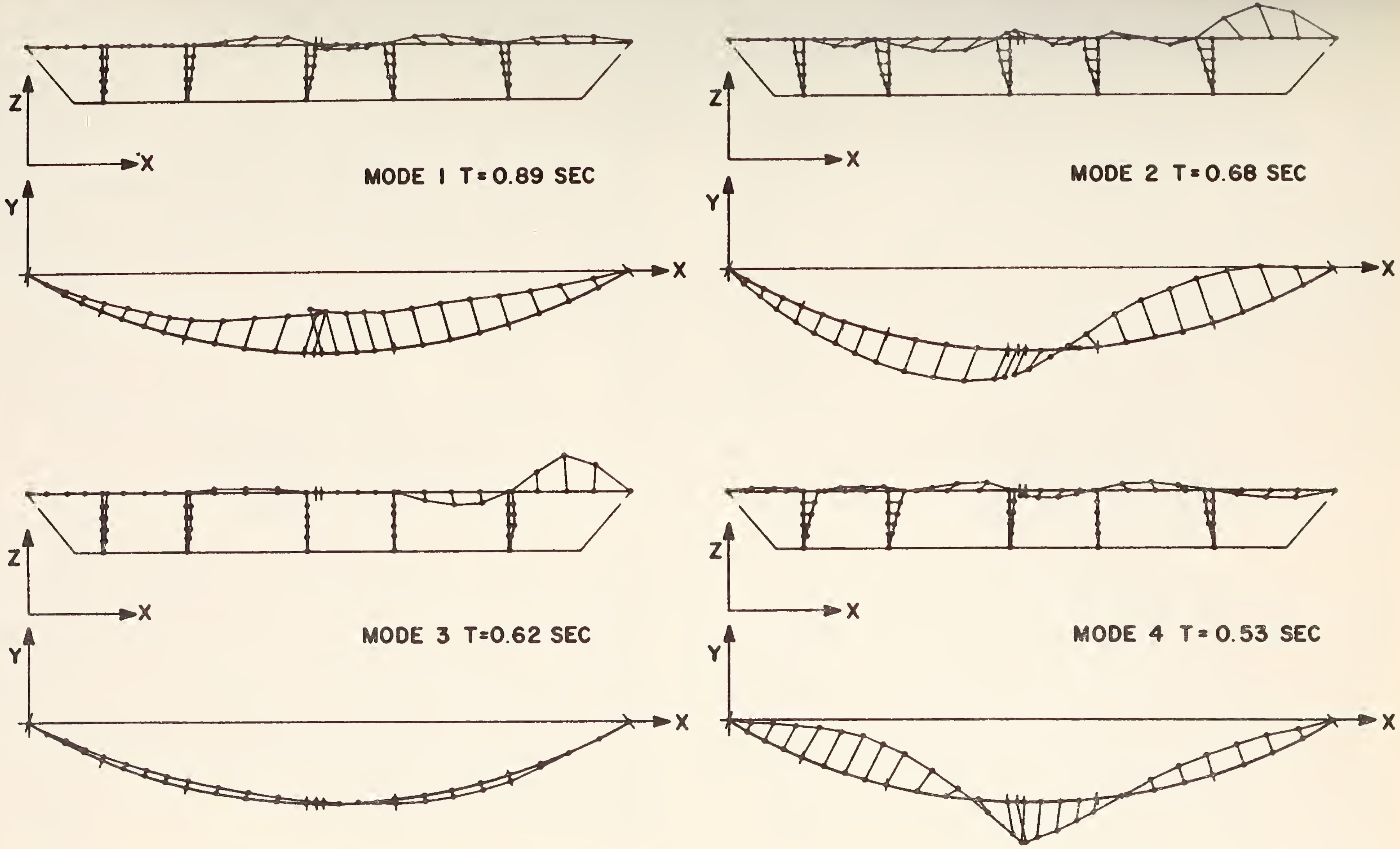
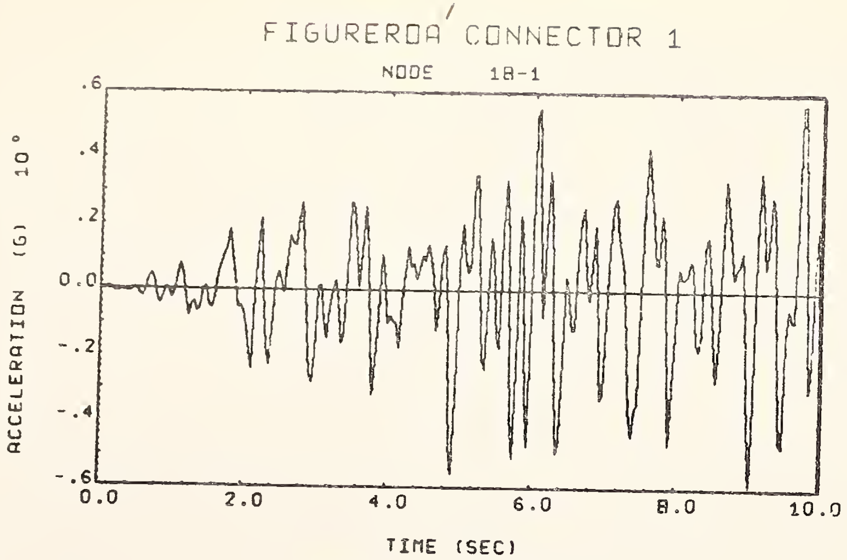
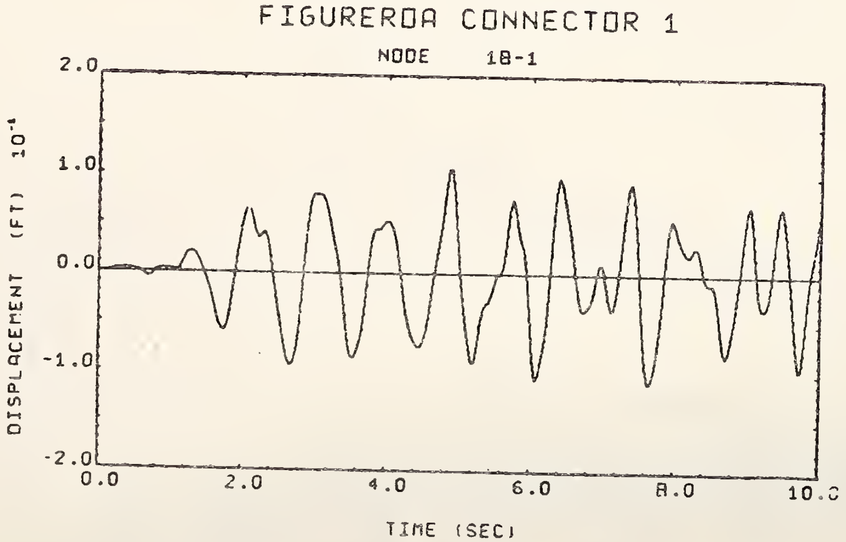
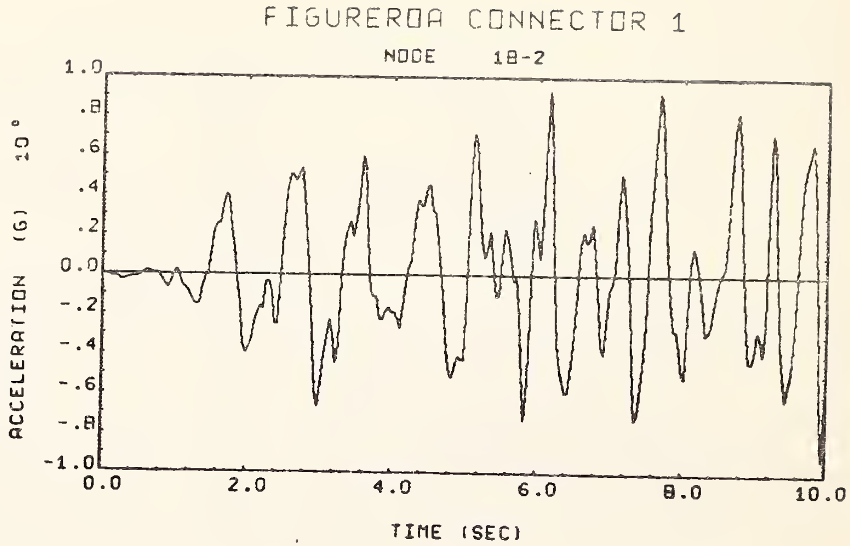


Fig. 68 Mode Shapes of the Curved Figueroa Street Undercrossing Connector: Zero Friction at the Expansion Joint



Global X - Component





Global Y - Component

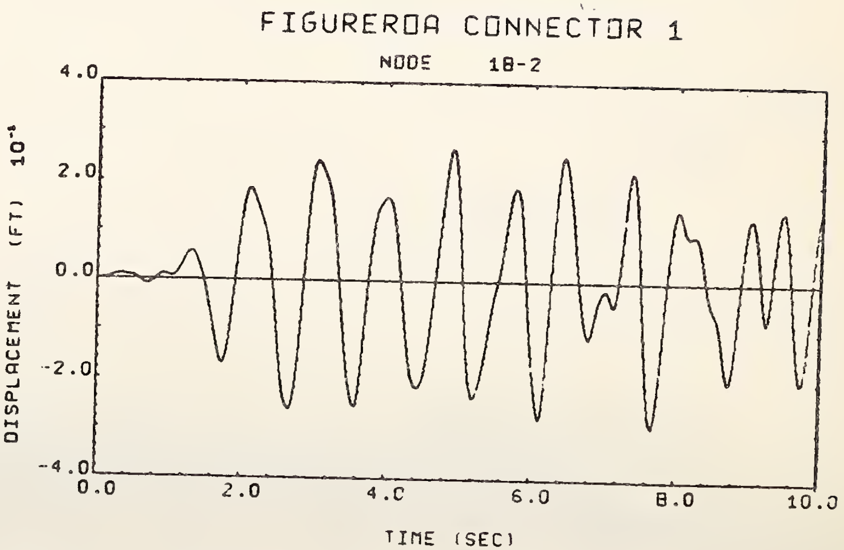
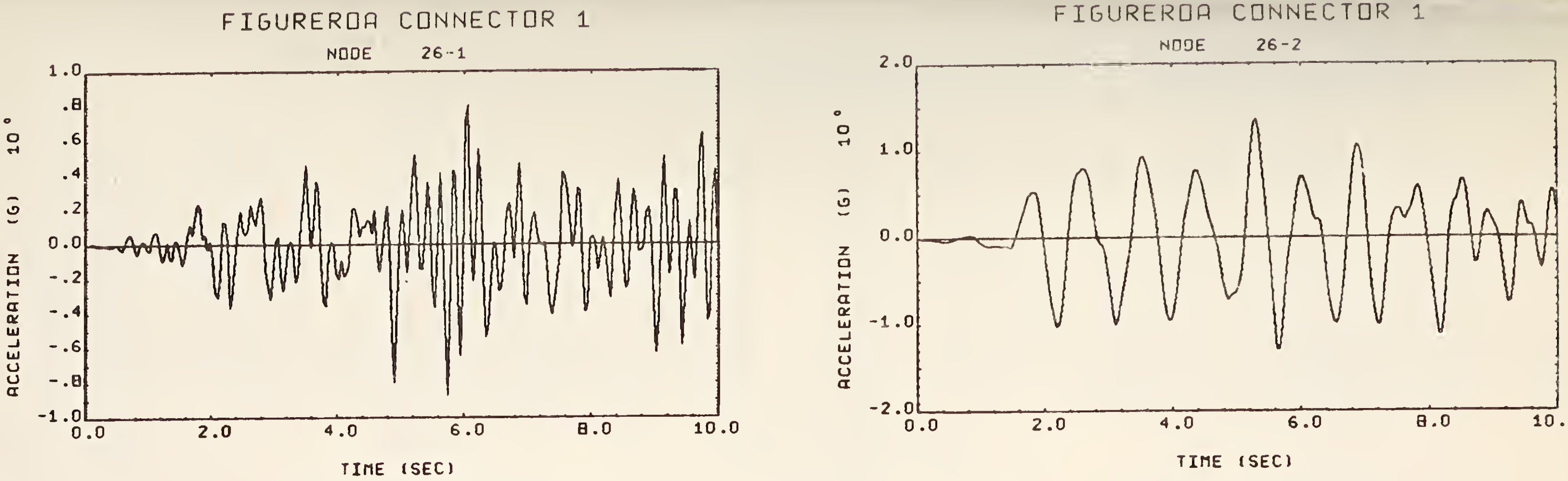


Fig. 69 Horizontal Accelerations and Displacements at the Top of Column # 3



Global X - Component

Global Y - Component

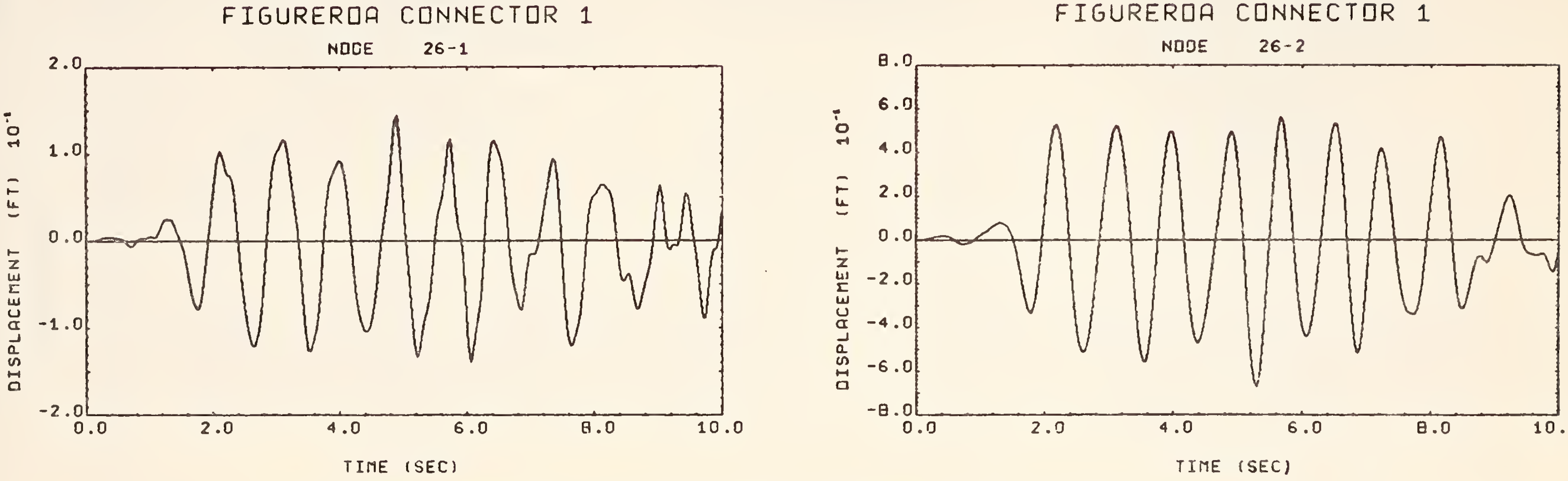
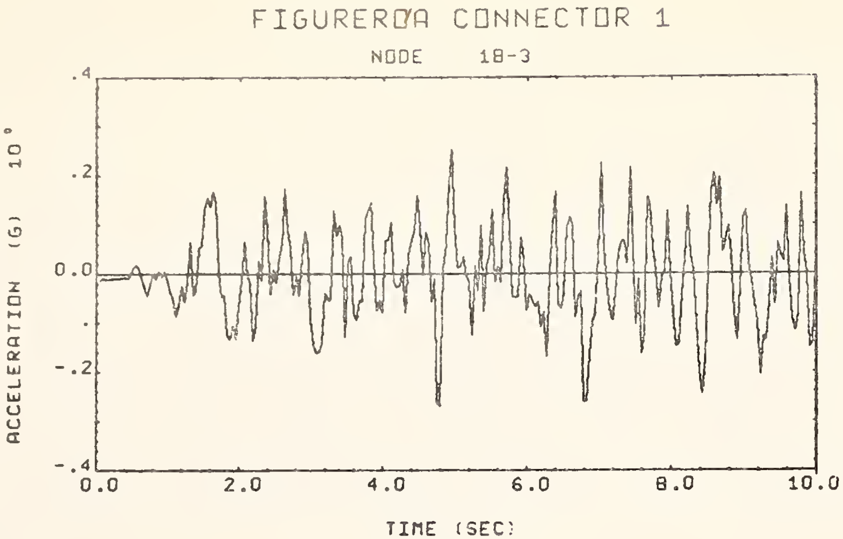
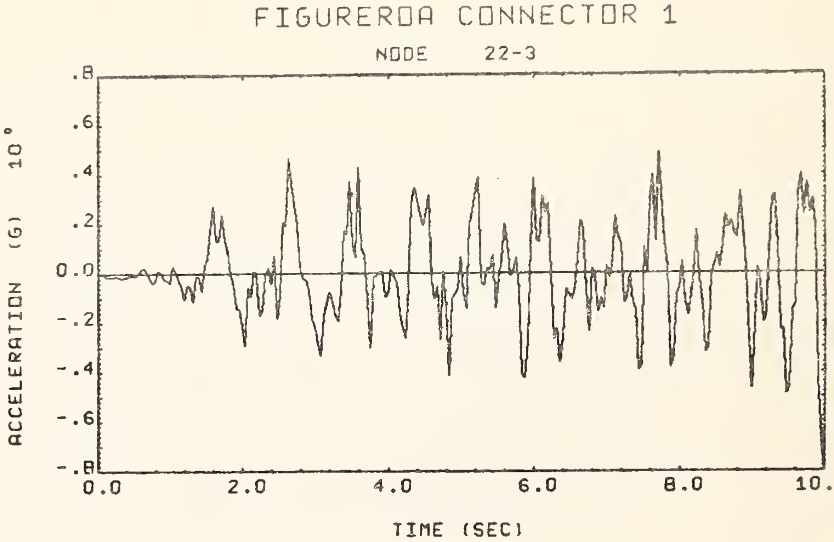


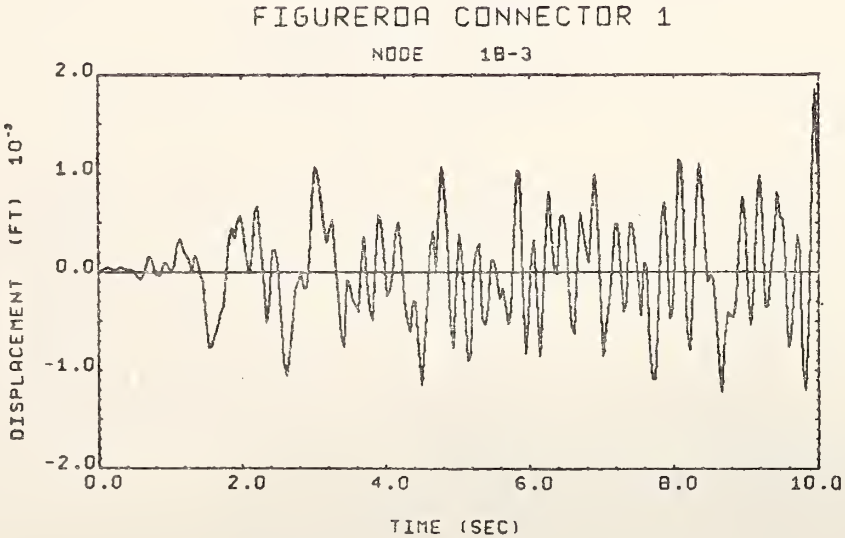
Fig. 70 Horizontal Accelerations and Displacements at the Top of Column # 4



Top of Column # 3



Center of Span # 3



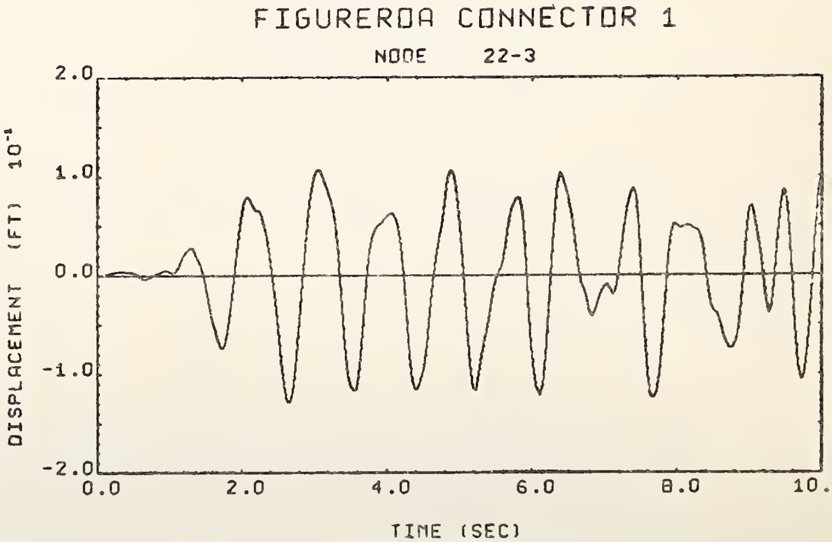
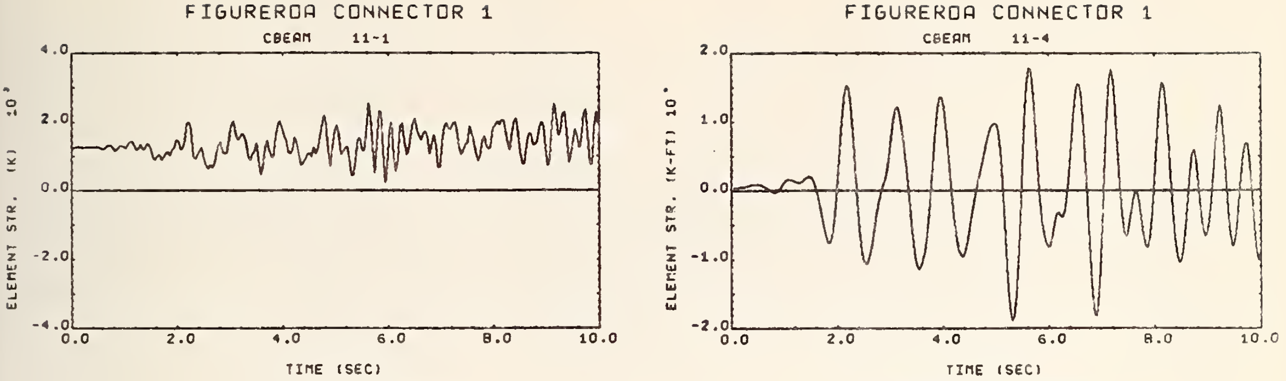
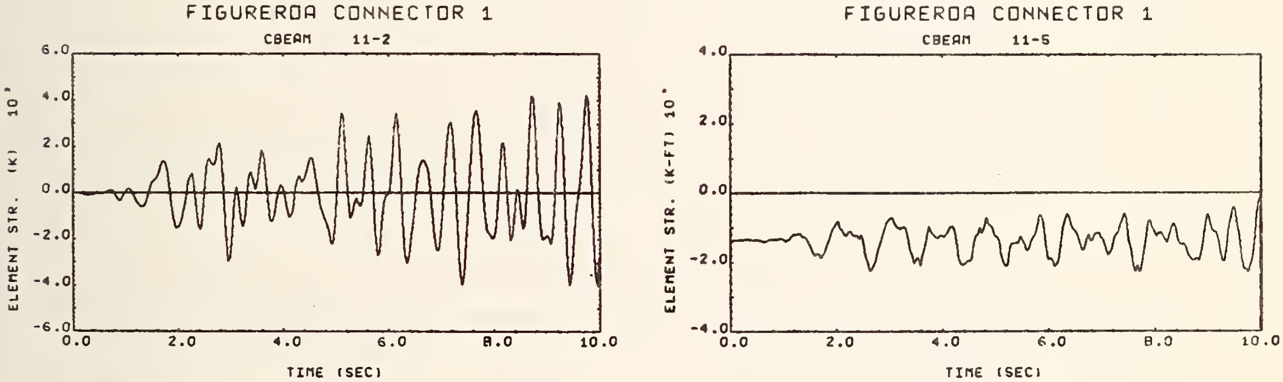


Fig. 71 Vertical Accelerations and Displacements at the Top of Column # 3 and the Center of Span # 3



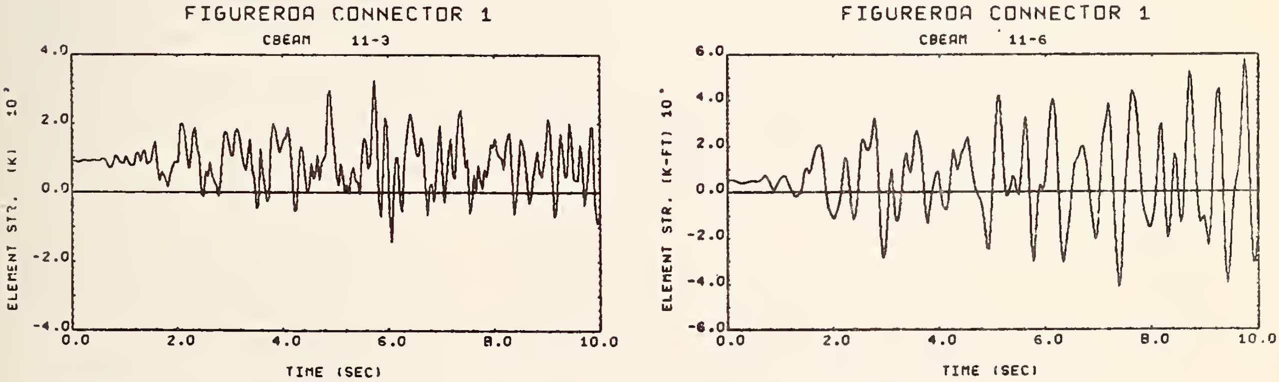
Axial Force

Torsional Moment



Shear in the Local y - Direction

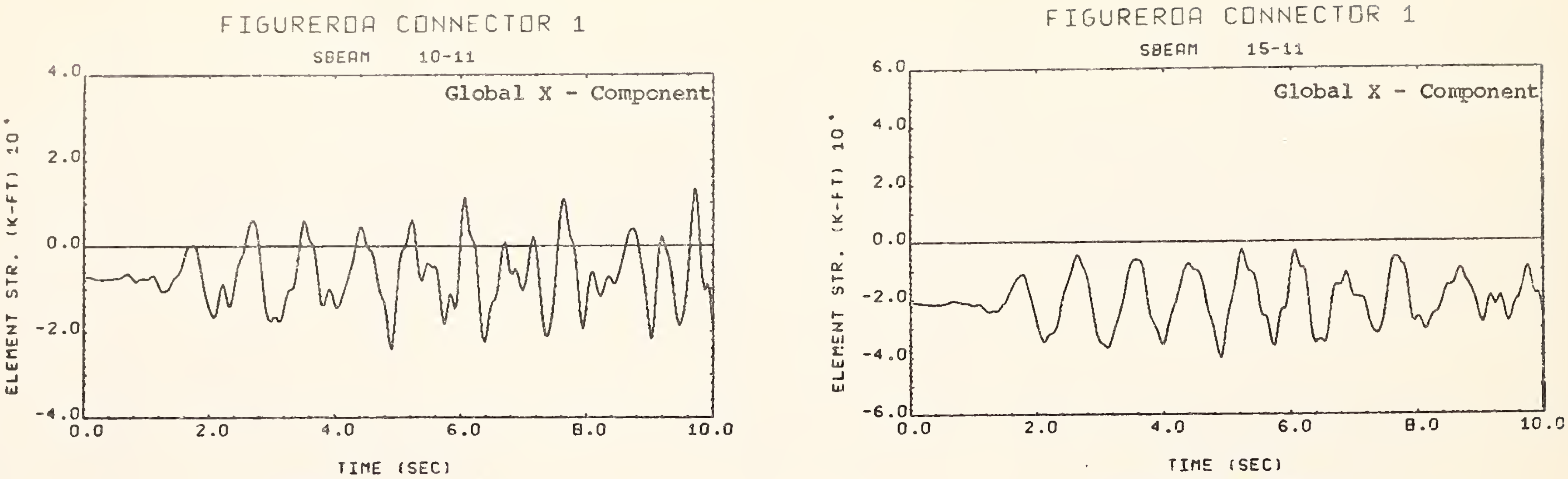
Moment about Local y - Axis



Shear in the Local z - Direction

Moment about Local z - Axis

Fig. 72 Generalized Forces in the Girder at the Center of Span # 3



Base of Column # 3

Base of Column # 4

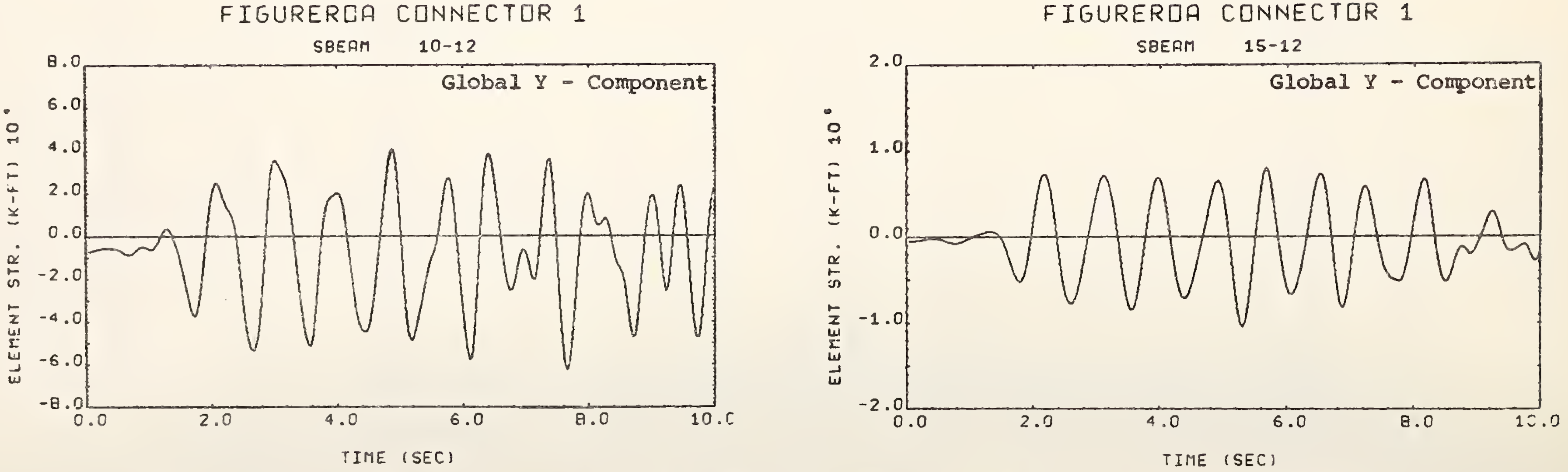
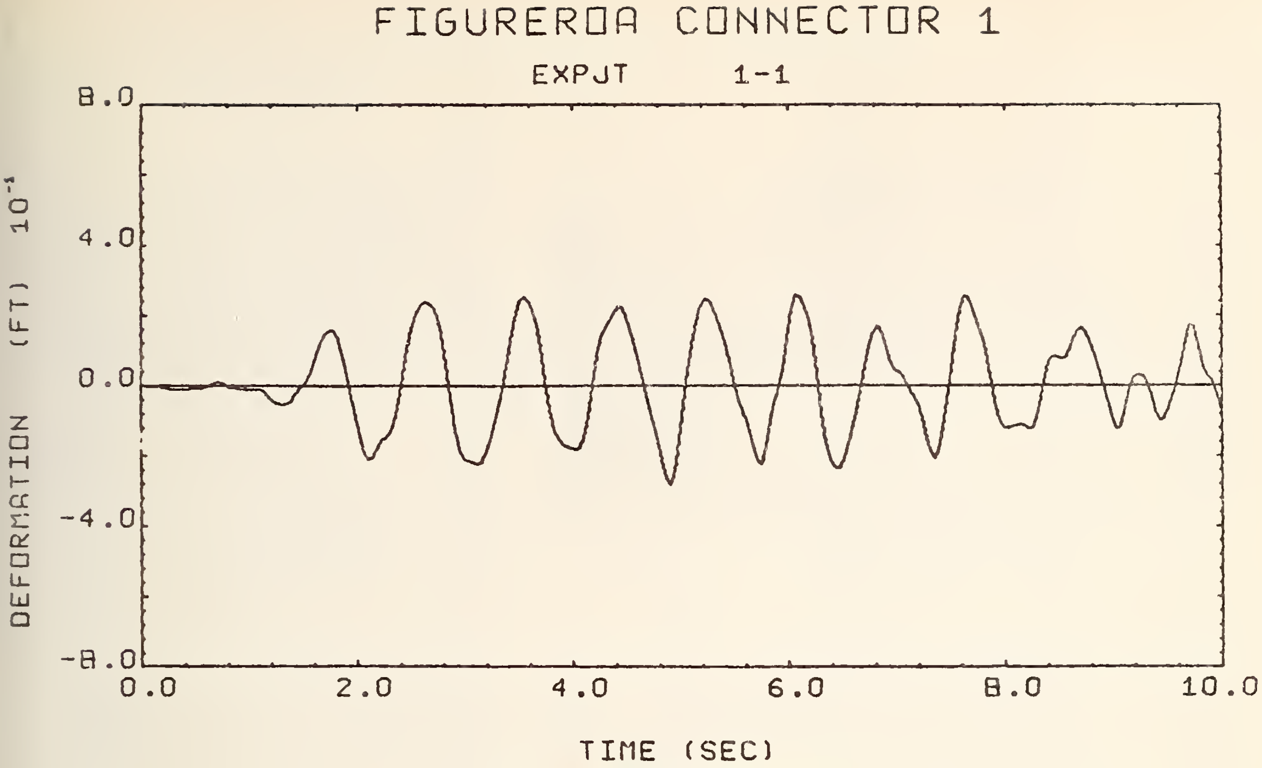
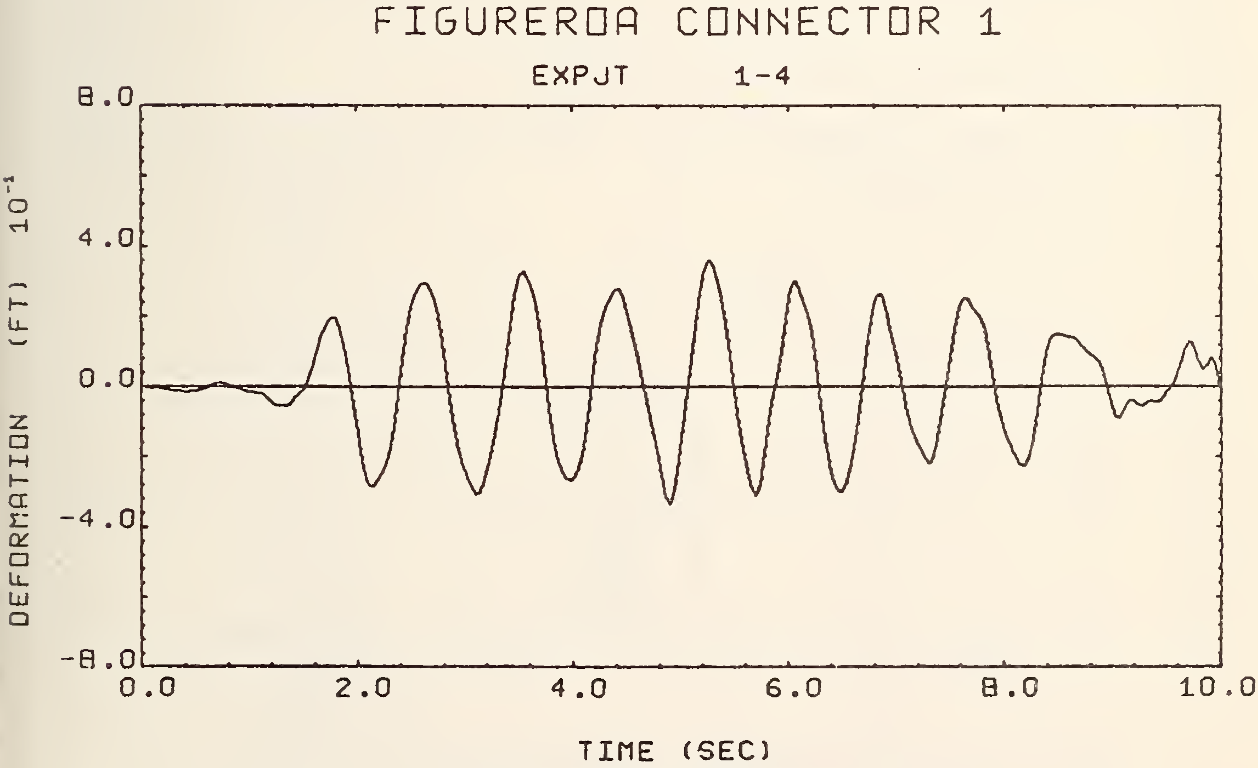


Fig. 73 Bending Moments at the Base of Columns # 3 and # 4

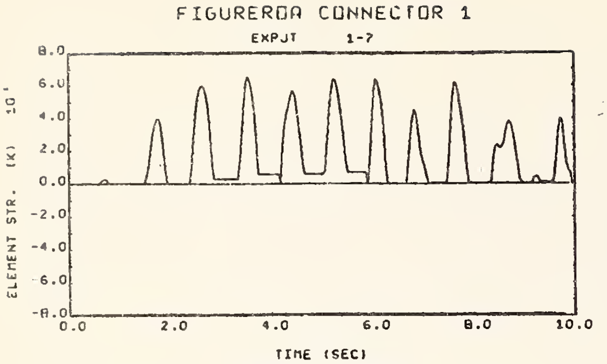


At Inner Edge of the Deck

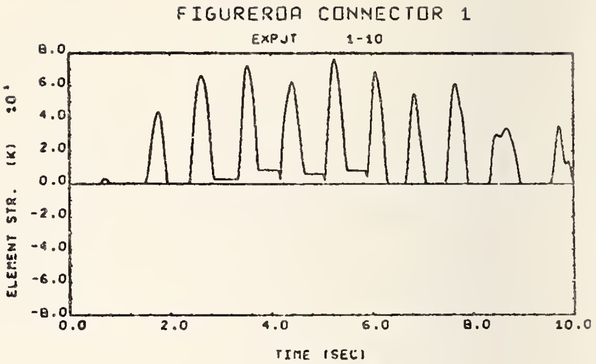


At Outer Edge of the Deck

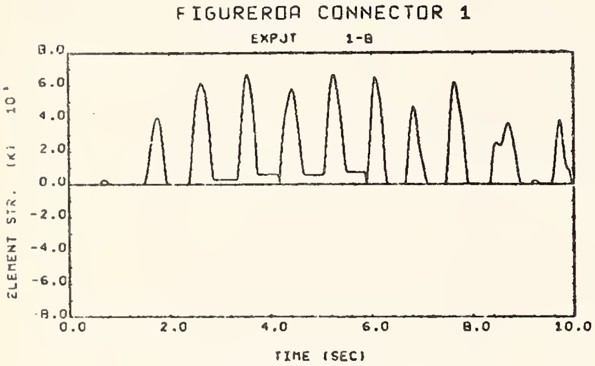
Fig. 74 Longitudinal Joint Separations at Expansion Joint # 1



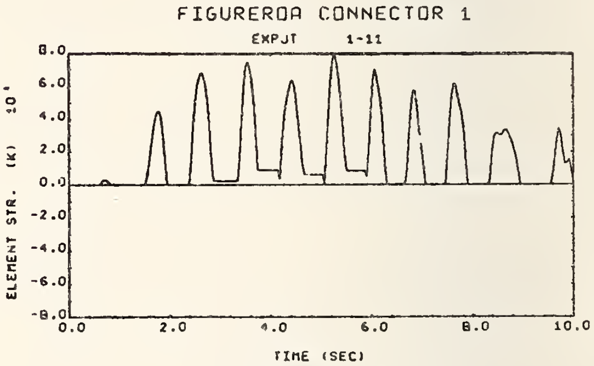
Tie Cable # 1



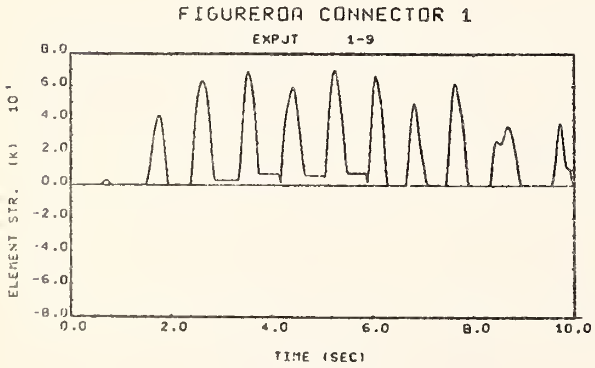
Tie Cable # 4



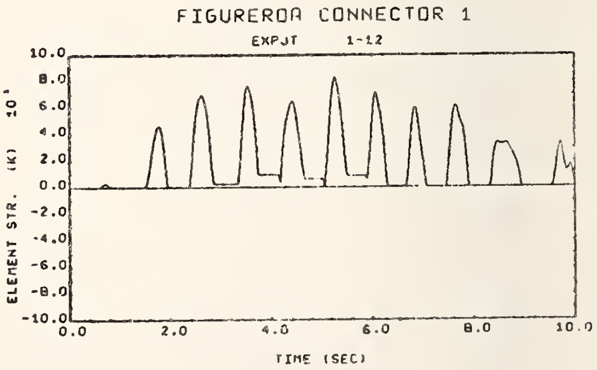
Tie Cable # 2



Tie Cable # 5

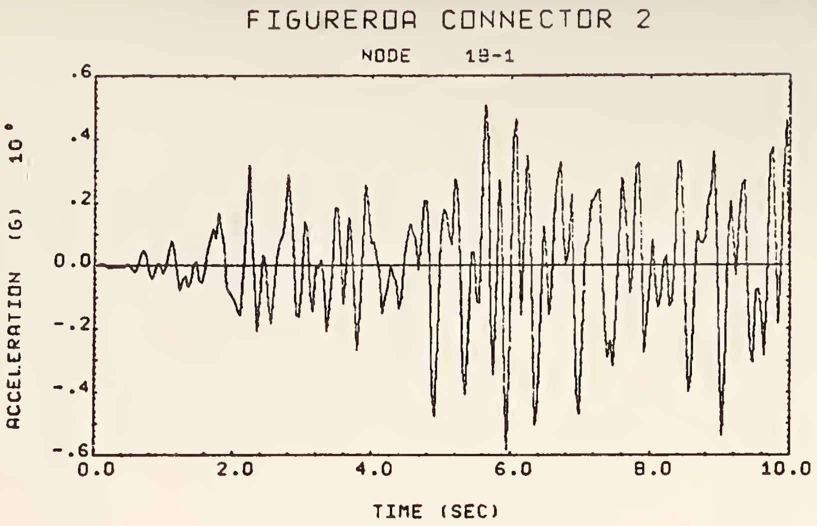


Tie Cable # 3

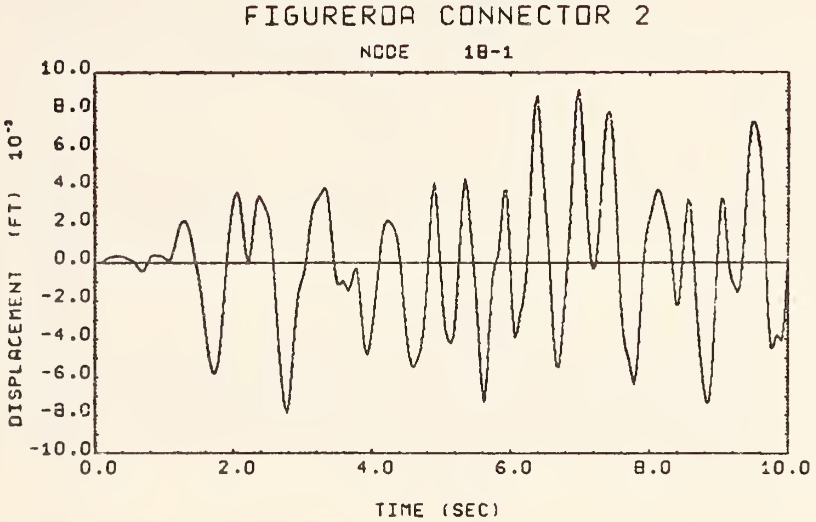


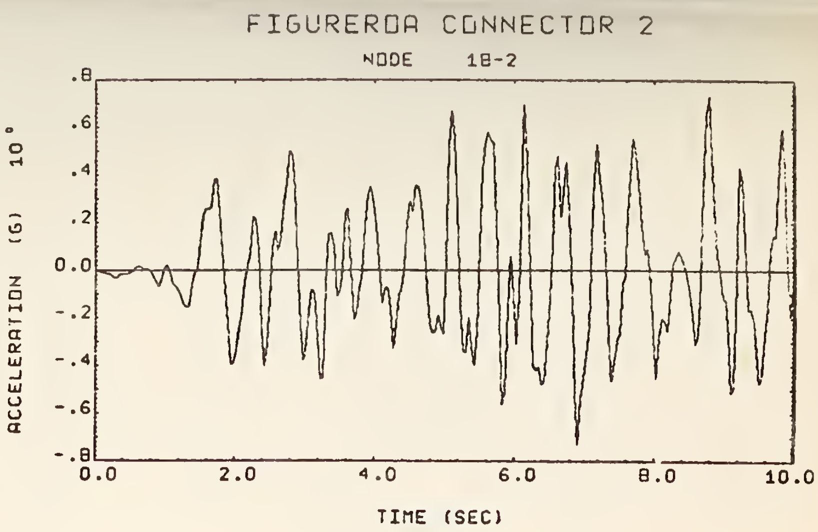
Tie Cable # 6

Fig. 75 Longitudinal Tie Cable Forces at Expansion Joint # 1



Global X - Component





Global Y - Component

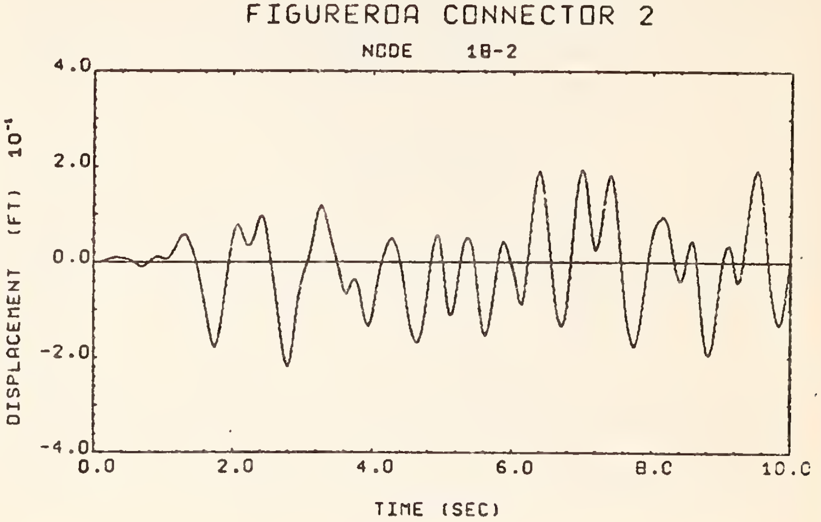
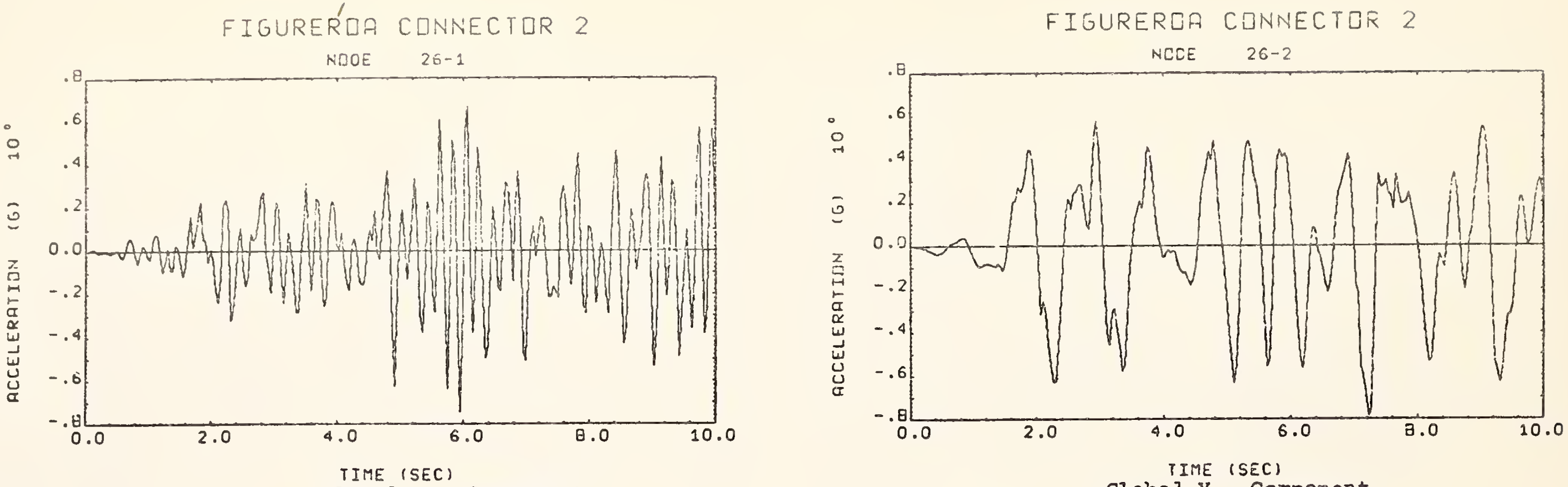


Fig. 76 Horizontal Accelerations and Displacements at the Top of Column # 3



Global X - Component

Global Y - Component

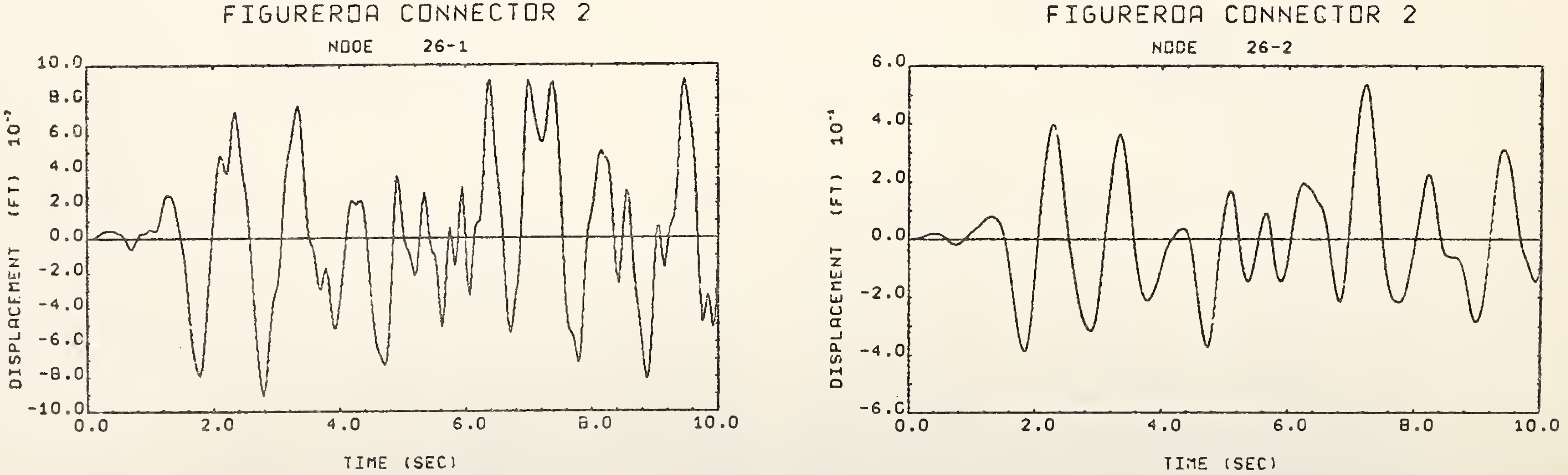
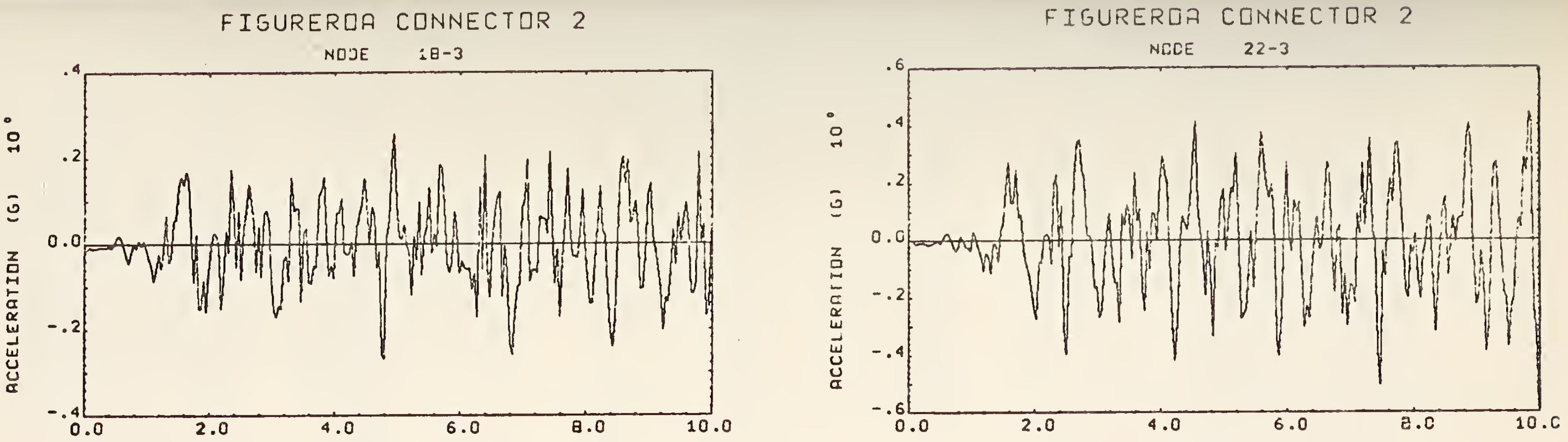


Fig. 77 Horizontal Accelerations and Displacements at the Top of Column # 4



Top of Column # 3

Center of Span # 3

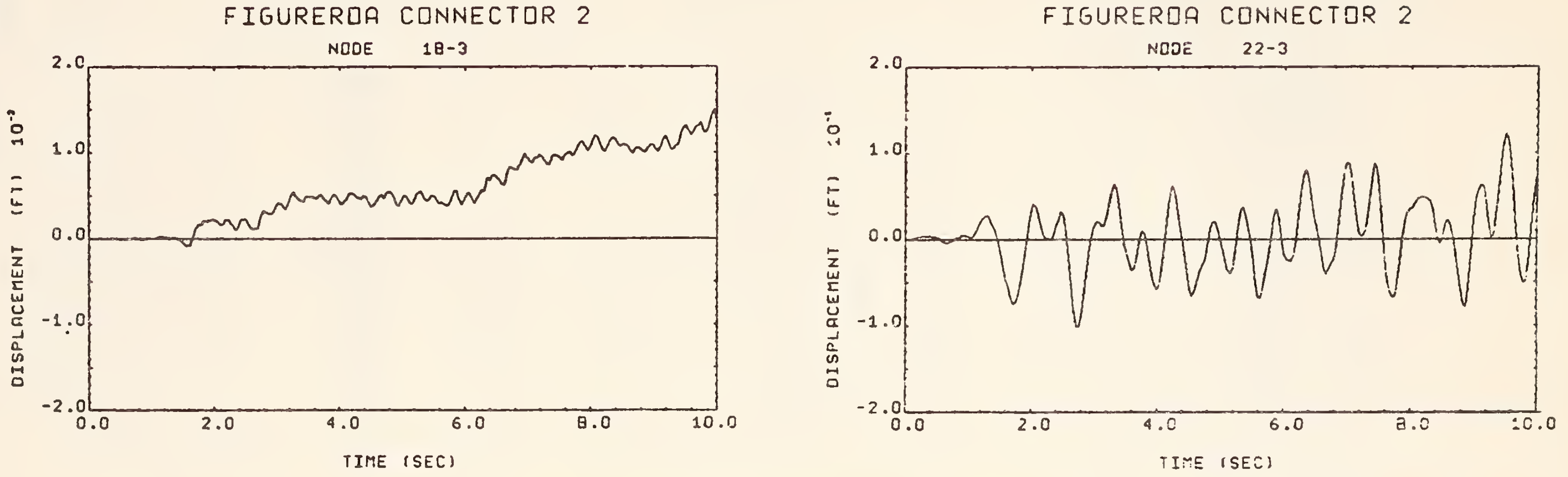
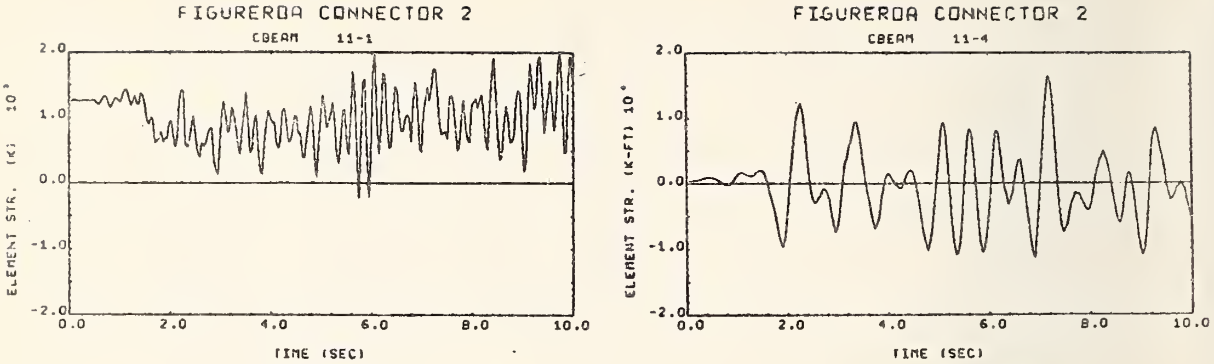
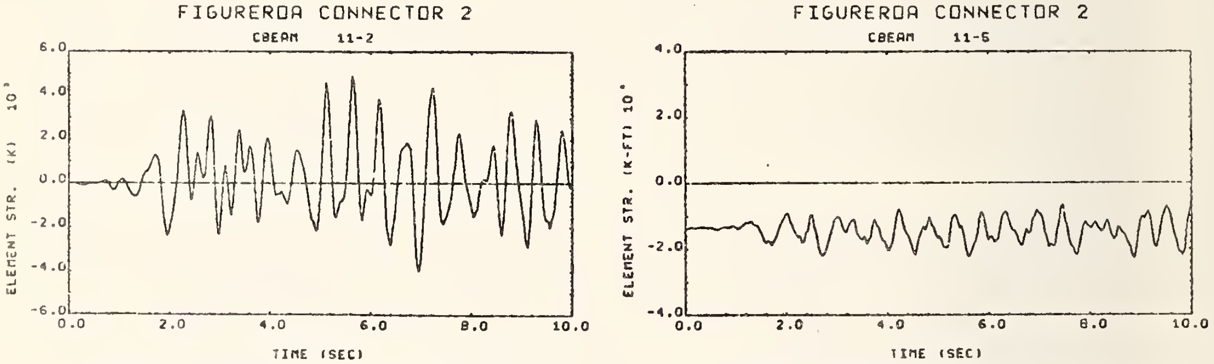


Fig. 78 Vertical Accelerations and Displacements at the Top of Column # 3 and the Center of Span # 3



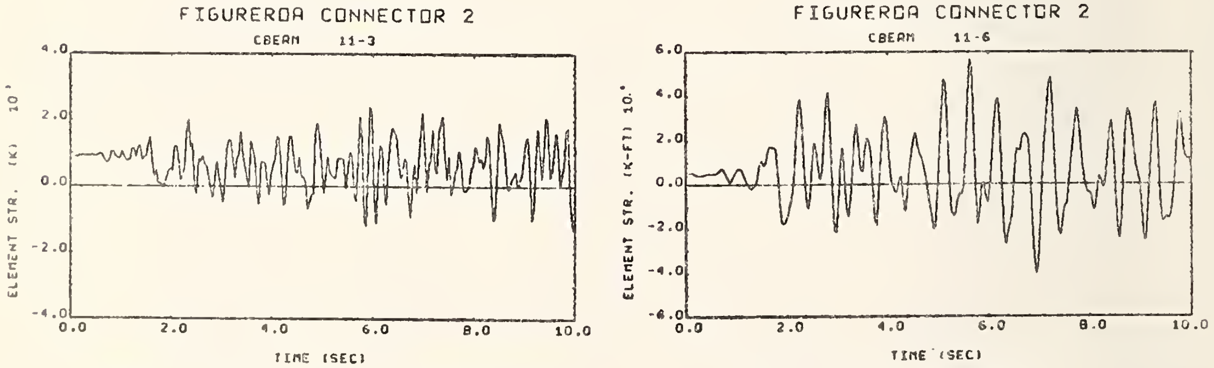
Axial Force

Torsional Moment



Shear in the Local y - Direction

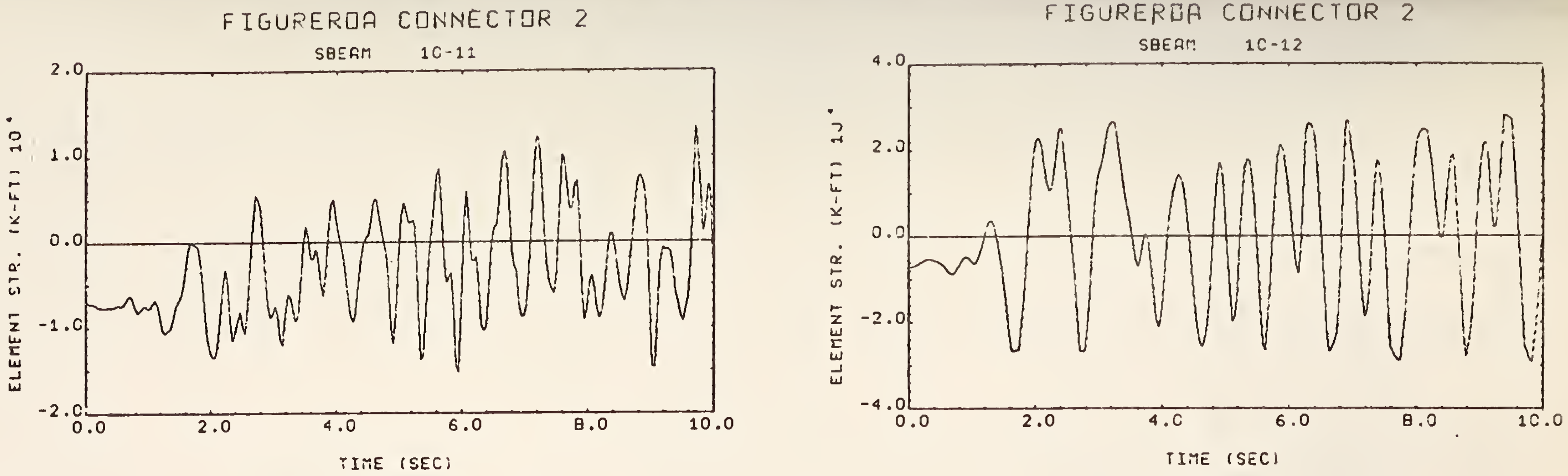
Moment about the Local y - Axis



Shear in the Local z - Direction

Moment about the Local z - Axis

Fig. 79 Generalized Forces in the Girder at the Center of Span # 3



About the Local y - Axis

About the Local z - Axis

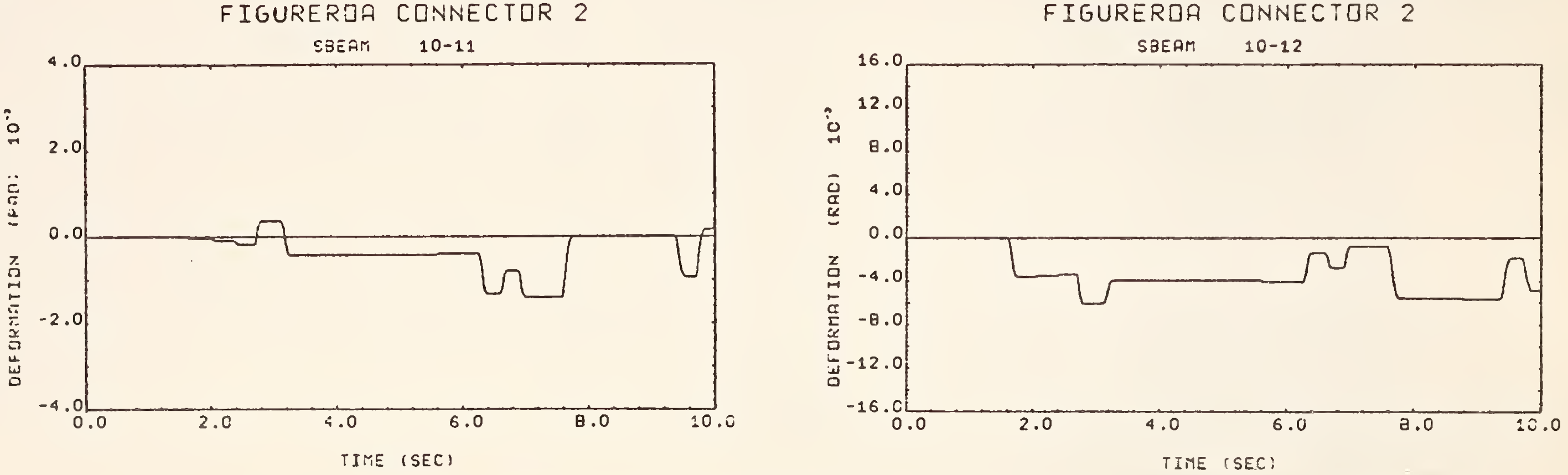
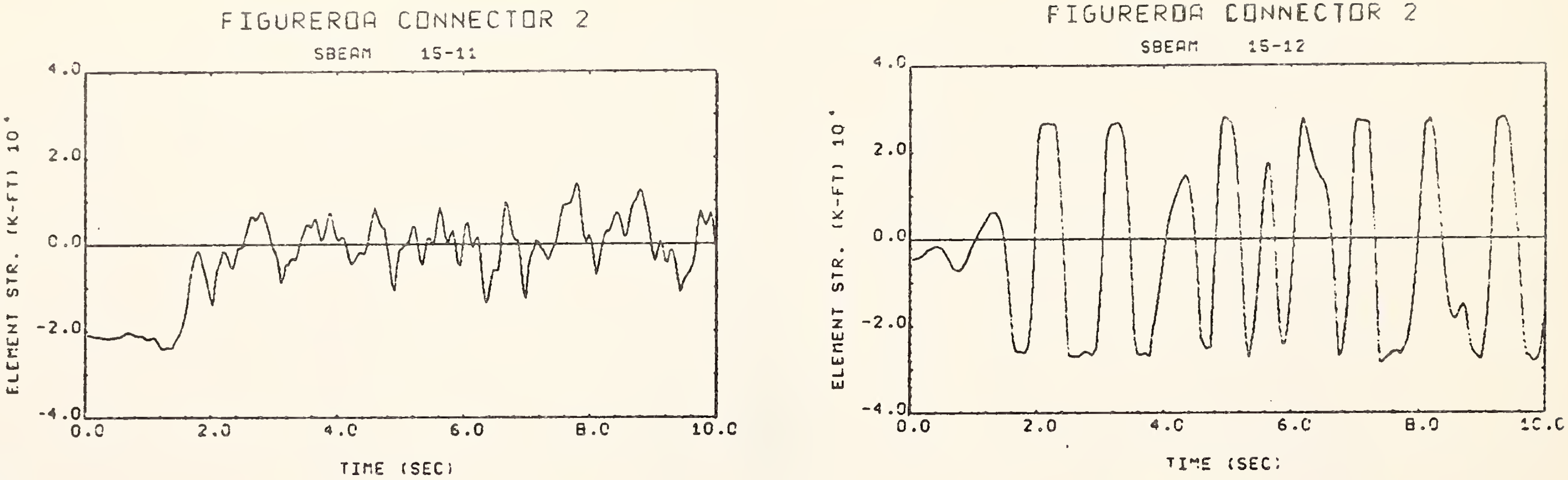


Fig. 80 Bending Moments and the Corresponding Plastic Rotations at the Base of Column # 3



About the Local y - Axis

About the Local z - Axis

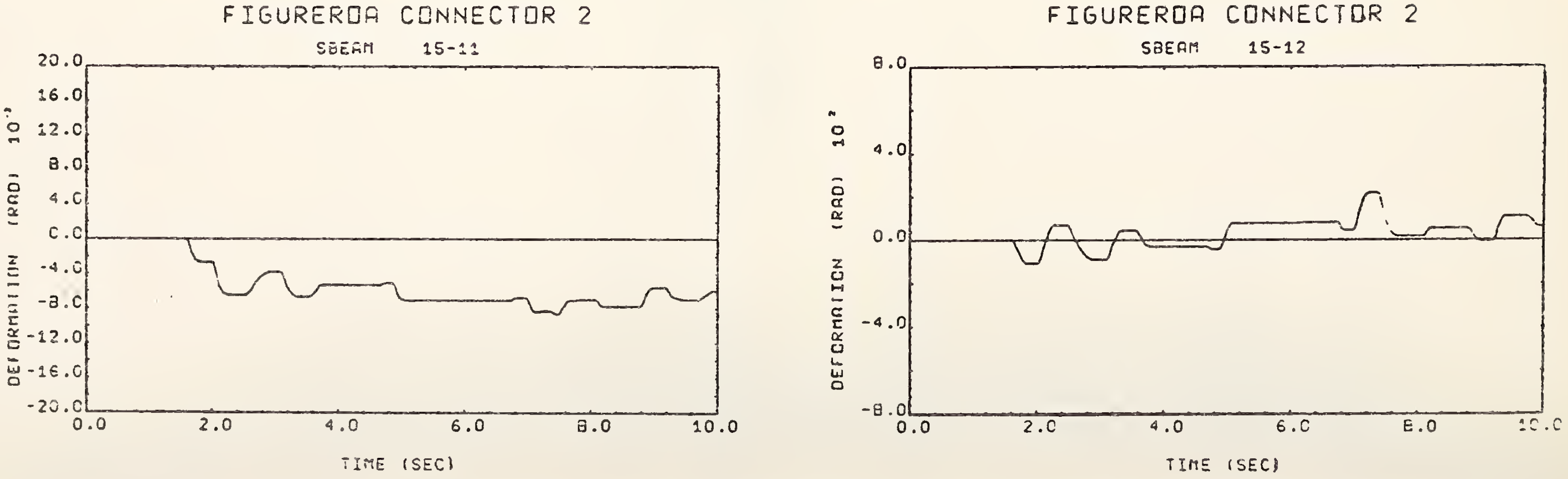
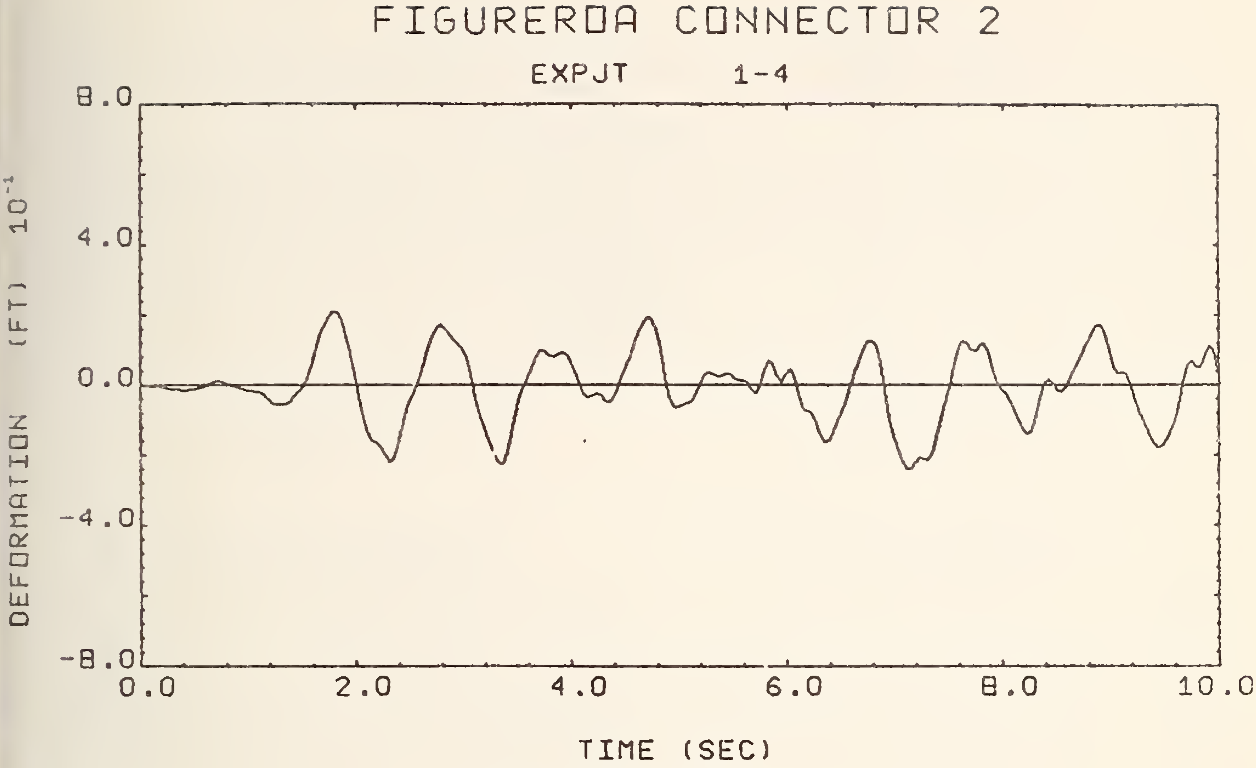
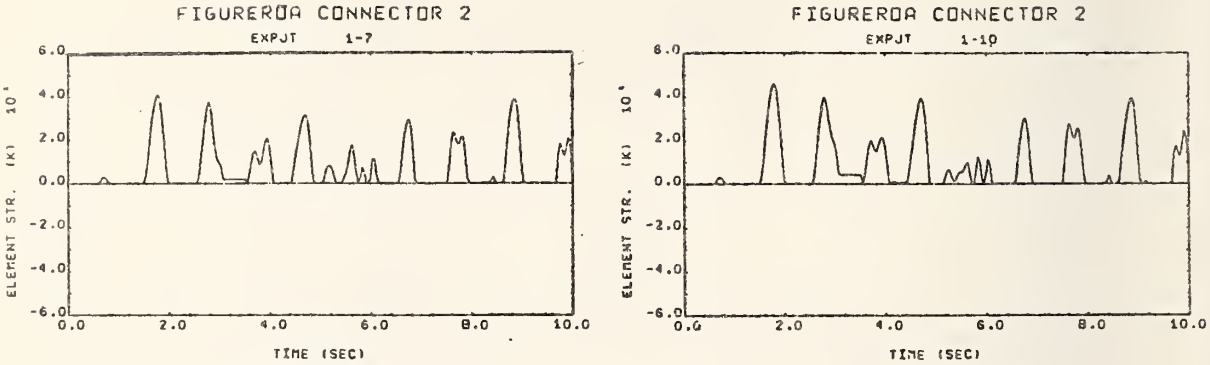


Fig. 81 Bending Moments and the Corresponding Plastic Rotations at the Base of Column # 4



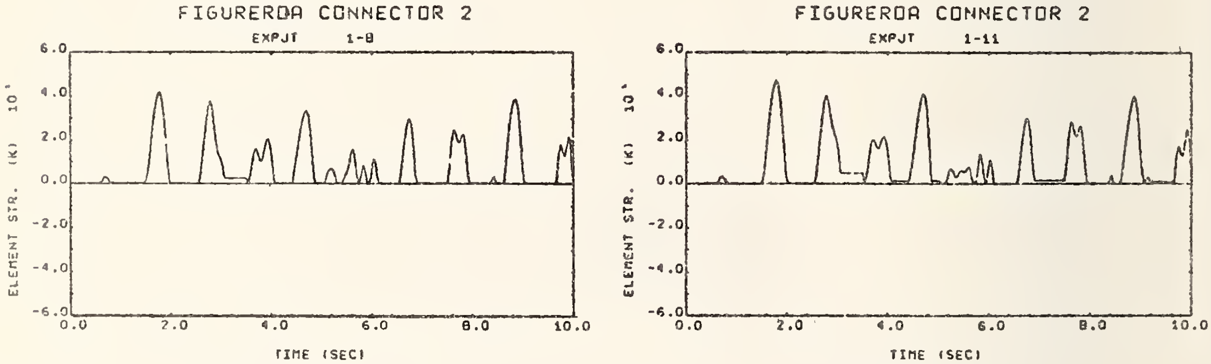
At the Outer Edge of the Deck

Fig. 82 Longitudinal Joint Separation at Expansion Joint # 1



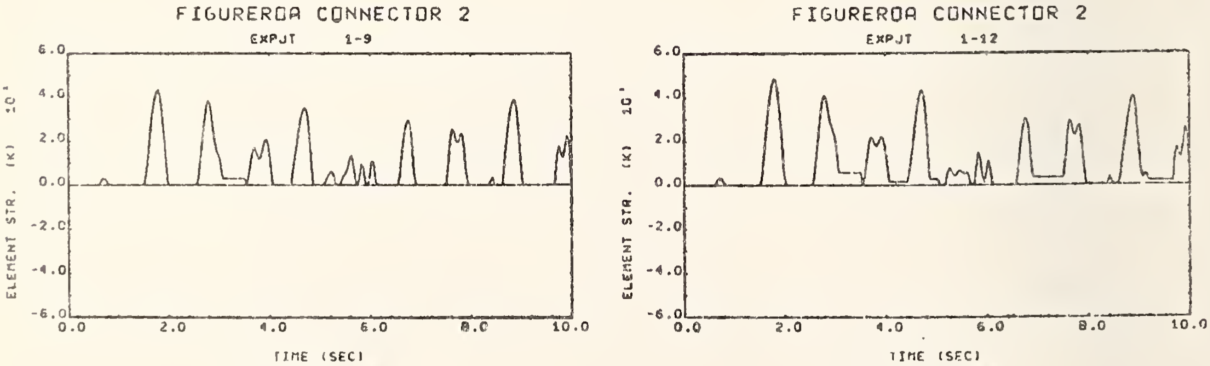
Tie Cable # 1

Tie Cable # 4



Tie Cable # 2

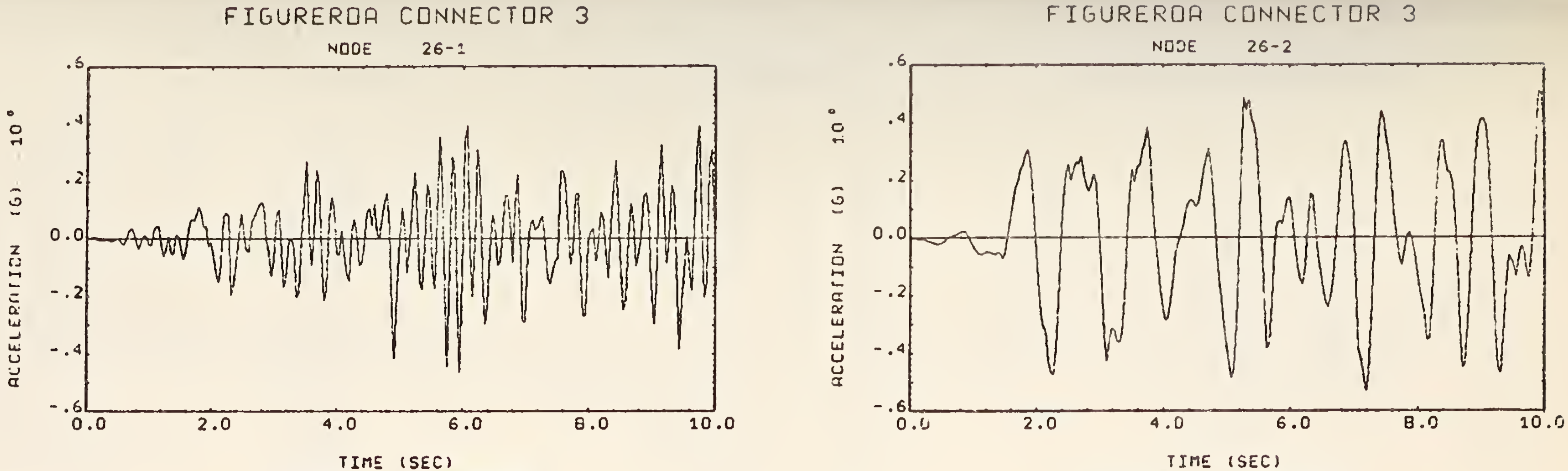
Tie Cable # 5



Tie Cable # 3

Tie Cable # 6

Fig. 83 Longitudinal Tie Cable Forces at Expansion Joint # 1



Global X - Component

Global Y - Component

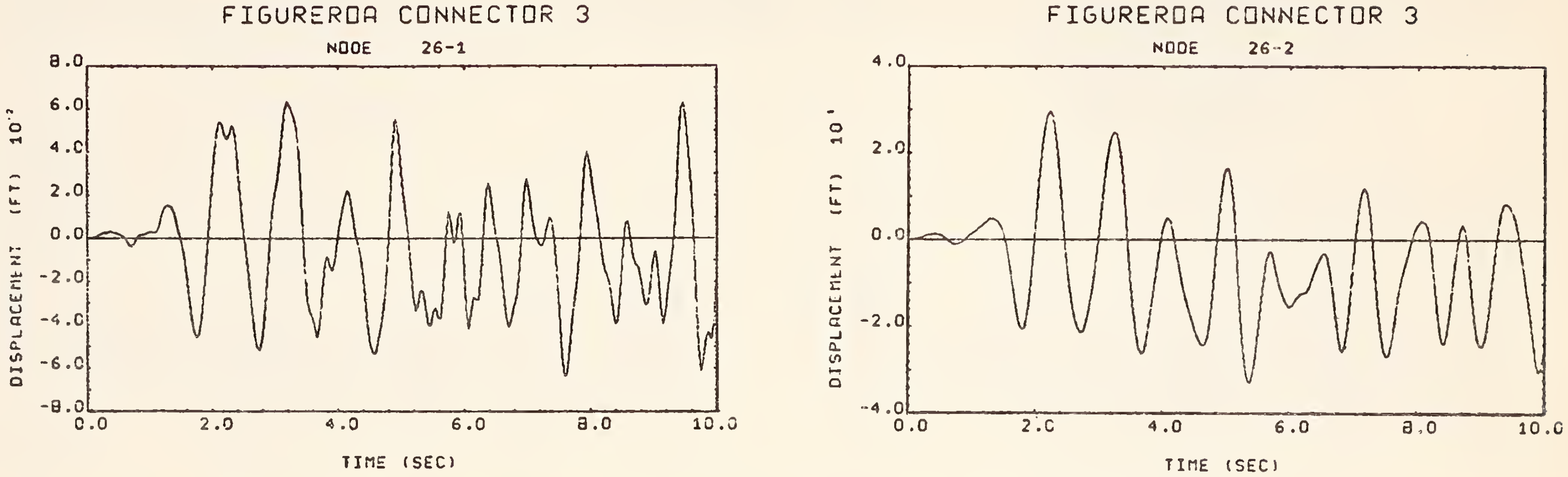
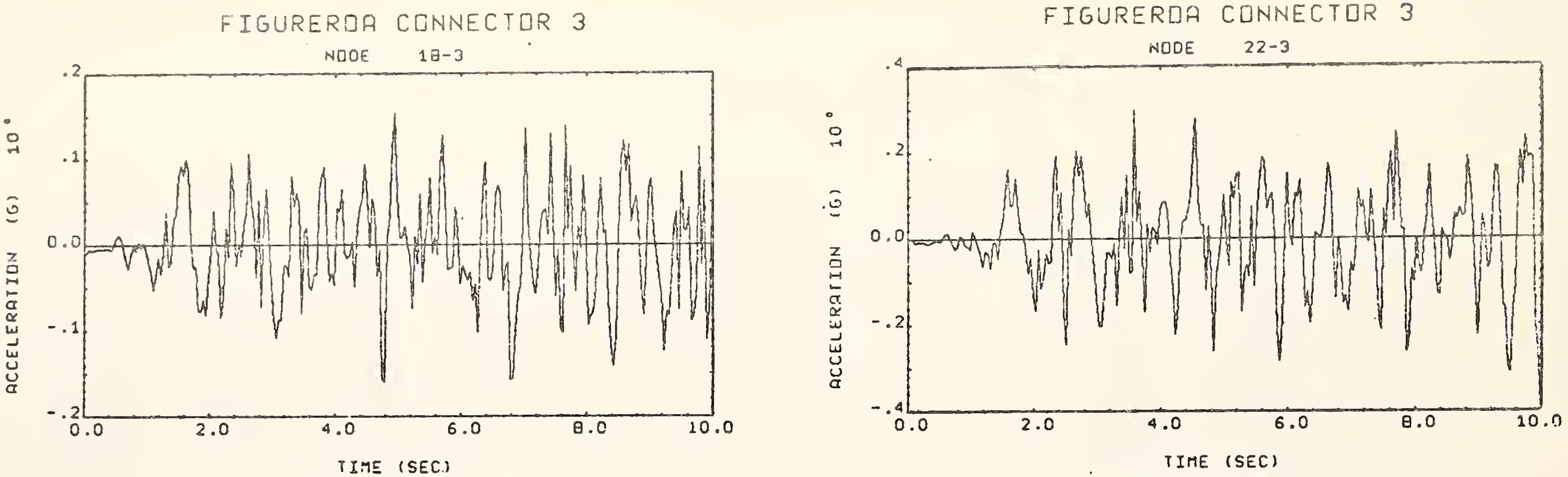


Fig. 84 Horizontal Accelerations and Displacements at the Top of Column # 3



Top of Column # 3

Center of Span # 3

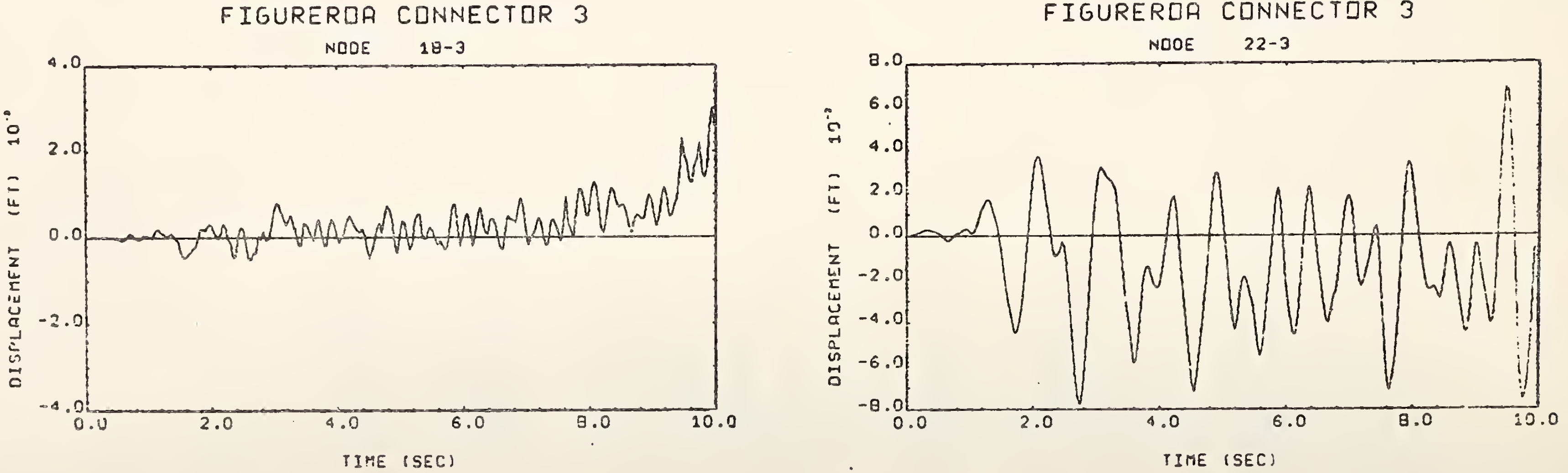
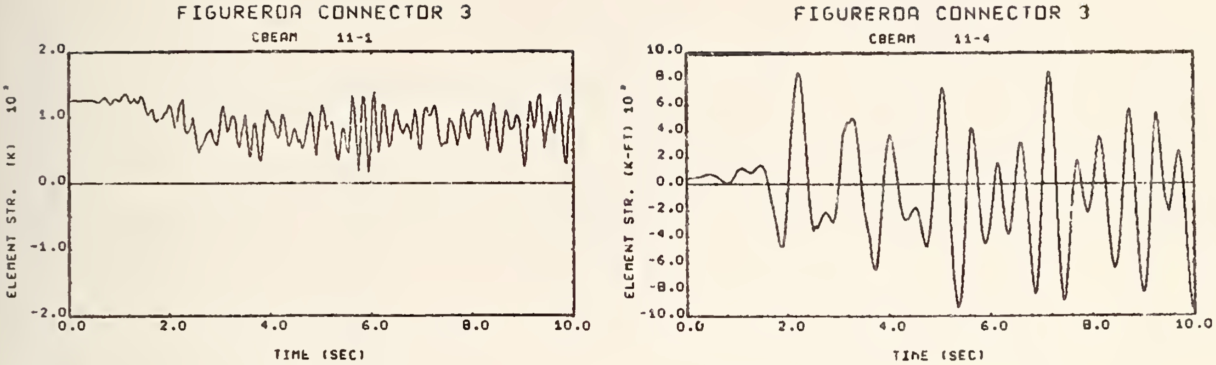
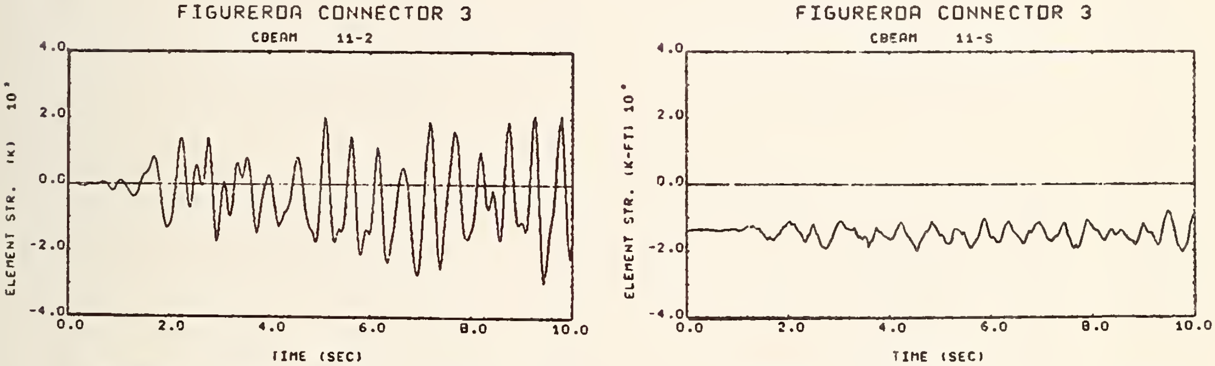


Fig. 85 Vertical Accelerations and Displacements at the Top of Column # 3 and the Center of Span # 3



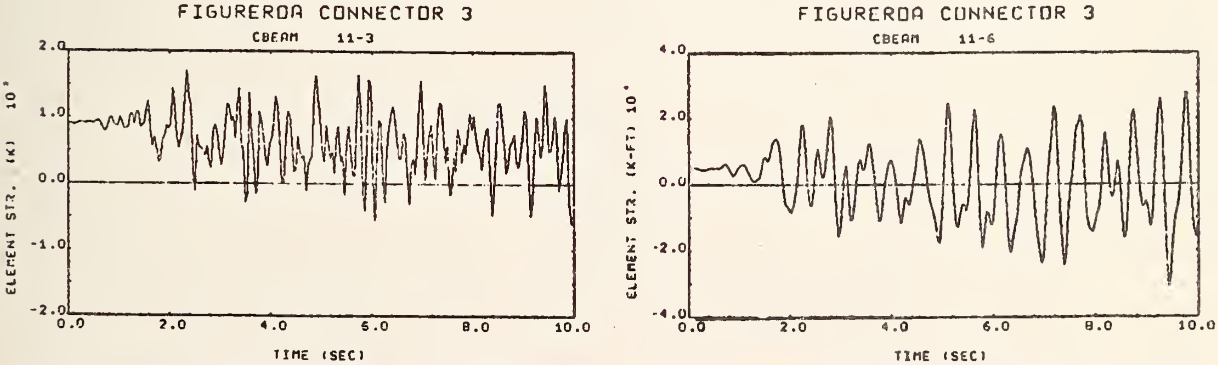
Axial Force

Torsional Moment



Shear in the Local y - Direction

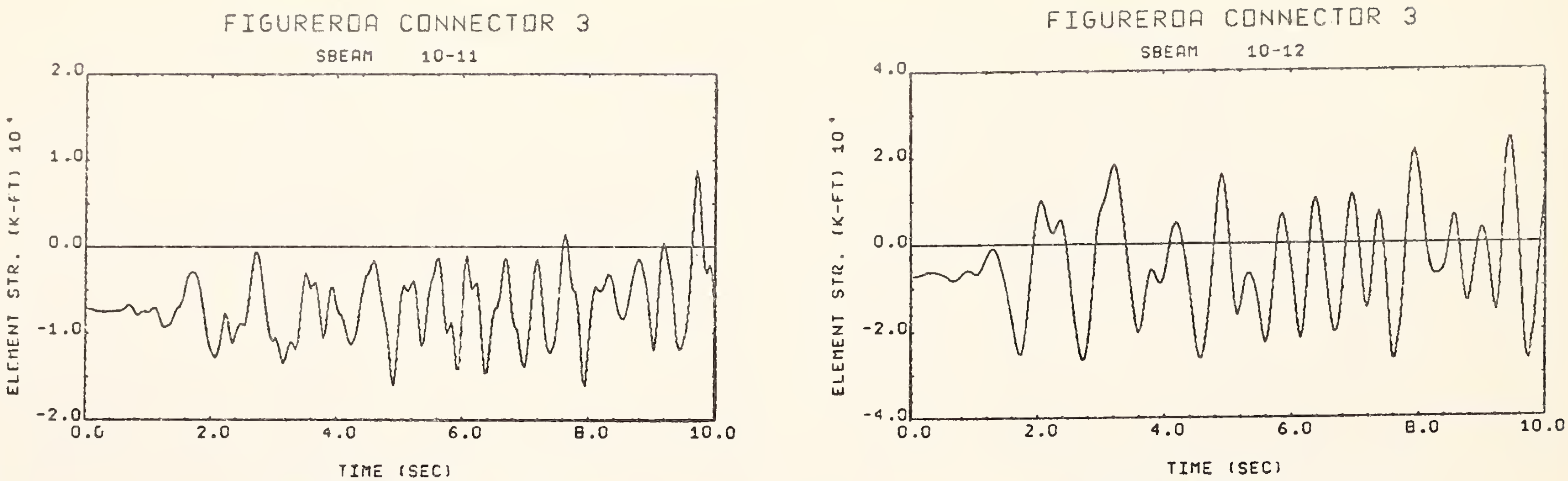
Moment about the Local y - Axis



Shear in the Local z - Direction

Moment about the Local z - Axis

Fig. 86 Generalized Forces in the Girder at the Center of Span # 3



About the Local y - Axis

About the Local z - Axis

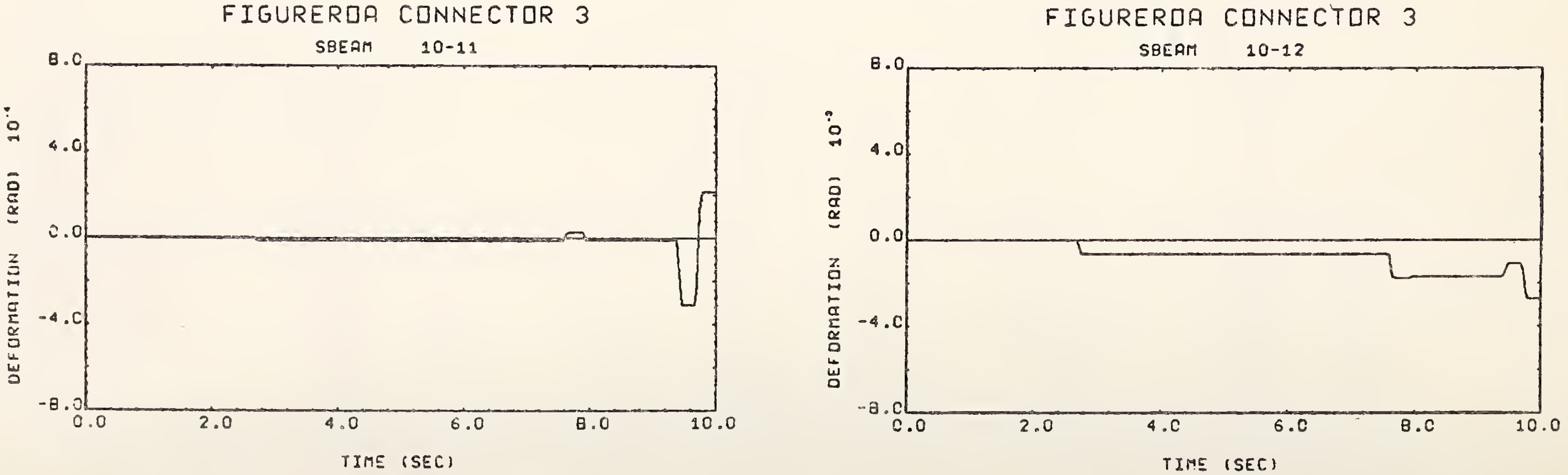
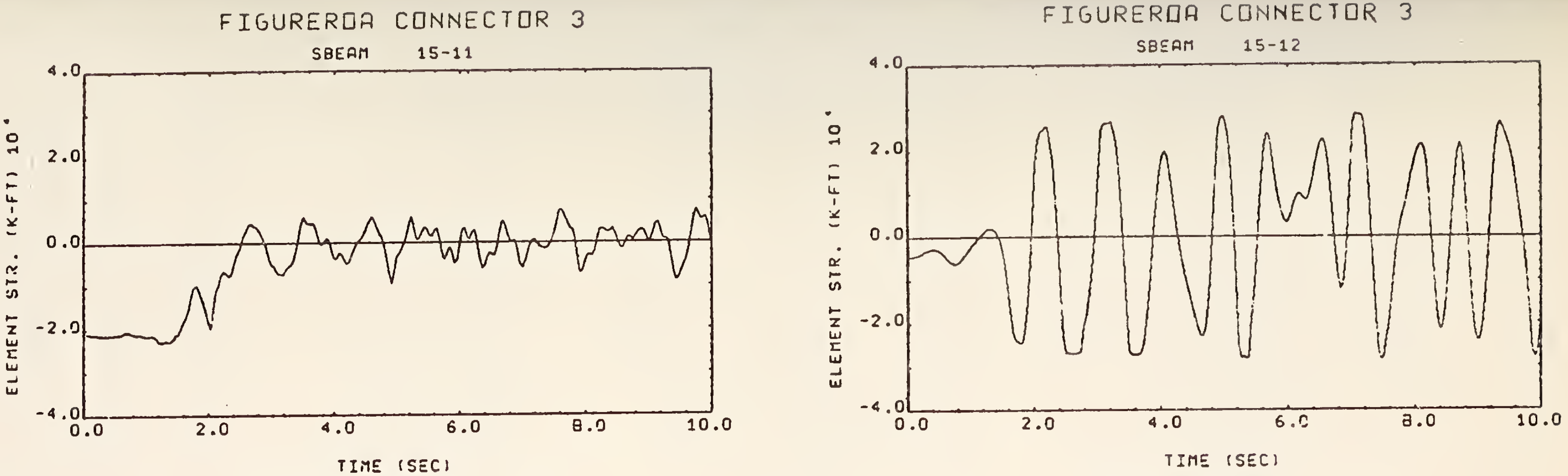


Fig. 87 Bending Moments and the Corresponding Plastic Rotations at the Base of Column # 3



About the Local y - Axis

About the Local z - Axis

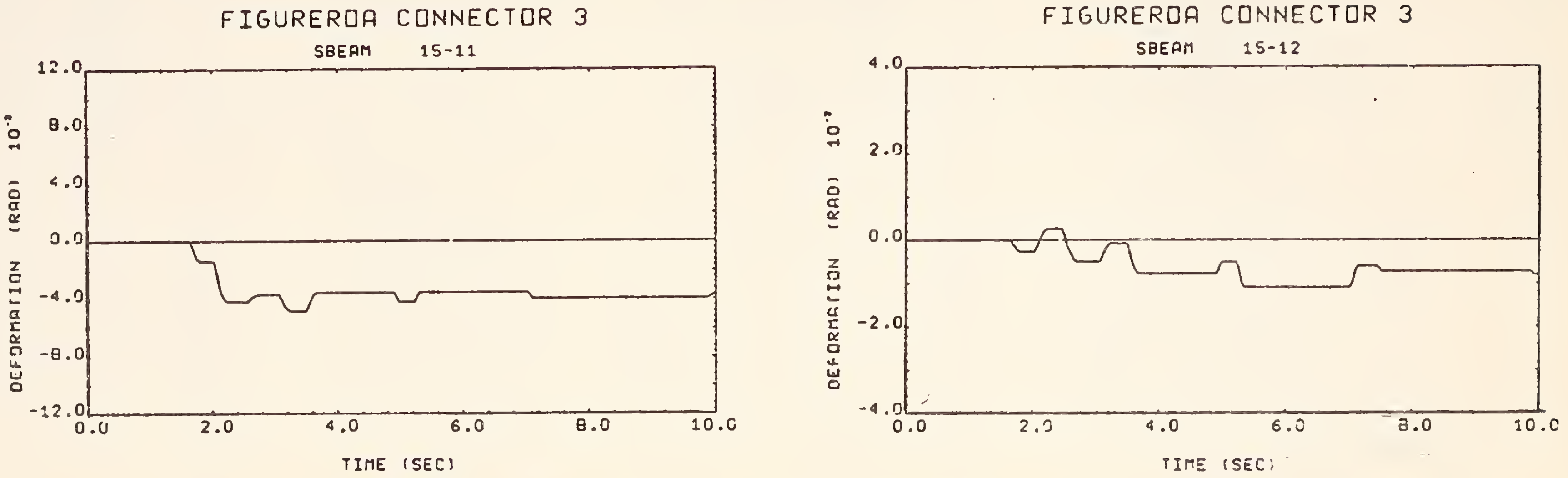
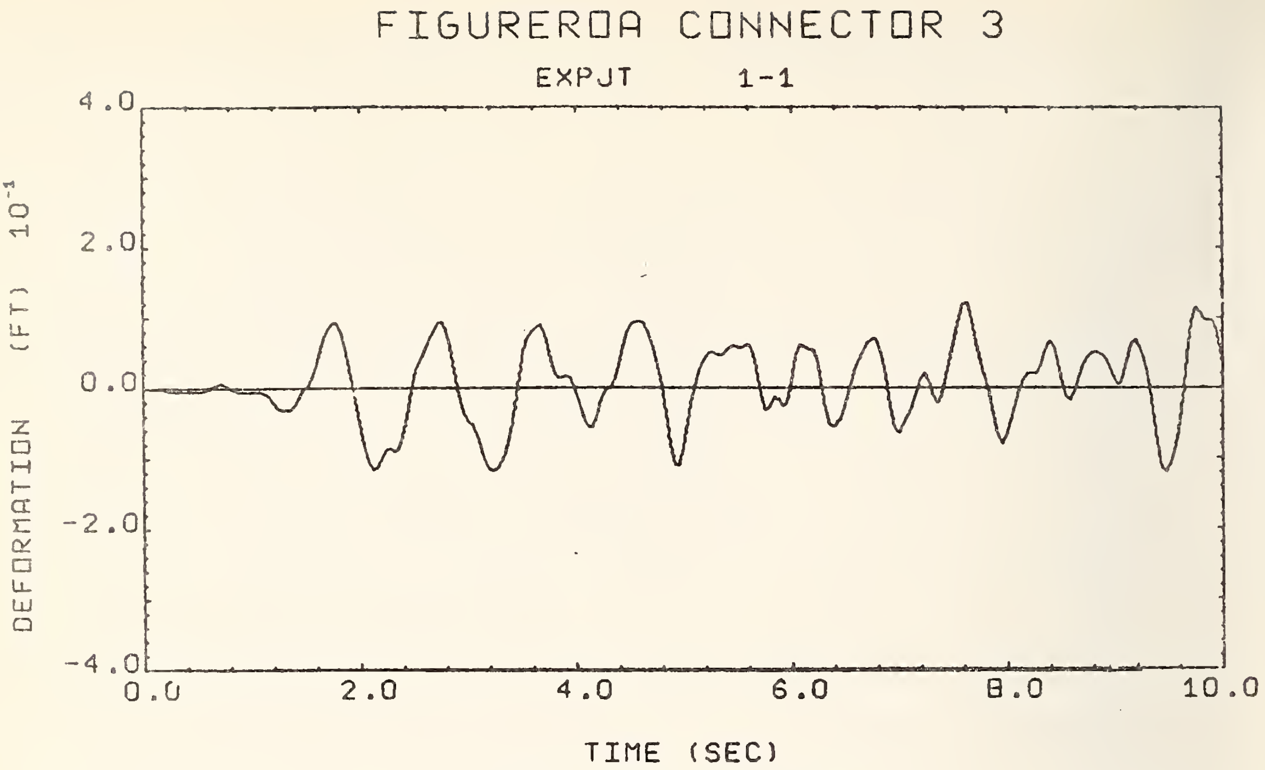
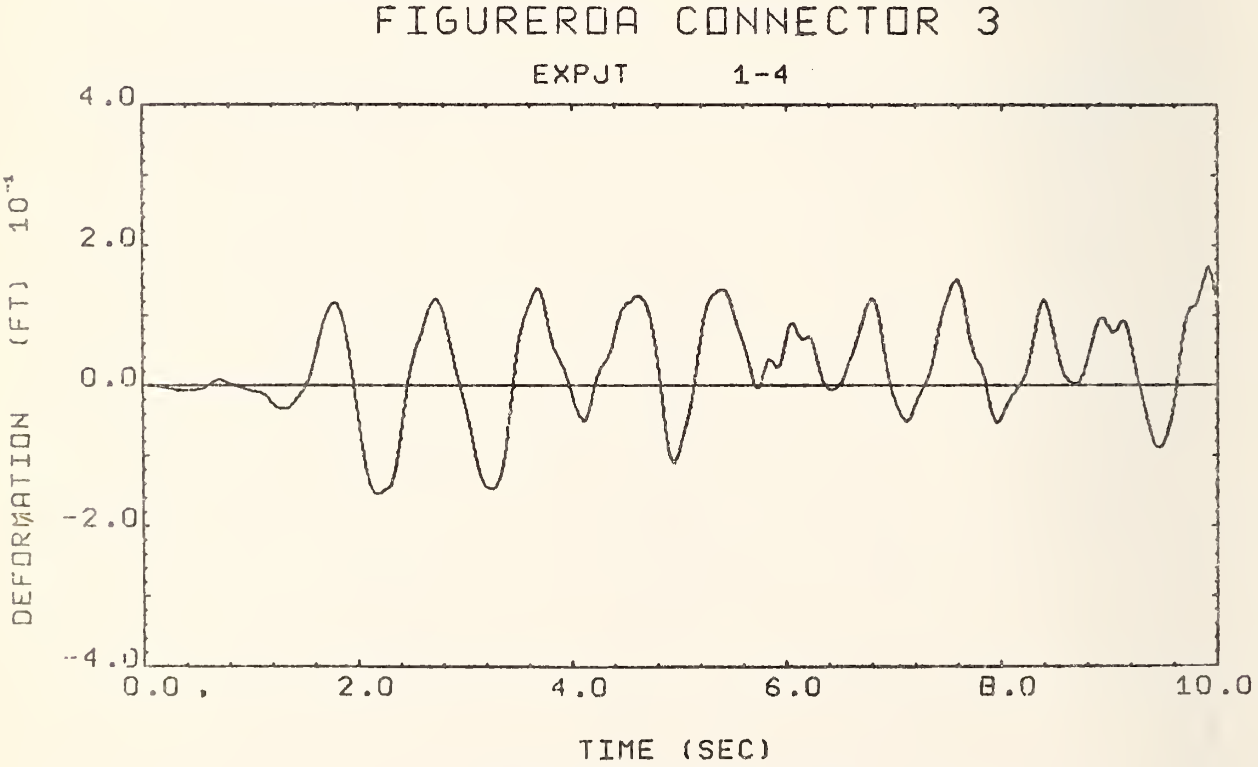


Fig. 88 Bending Moments and the Corresponding Plastic Rotations at the Base of Column # 4

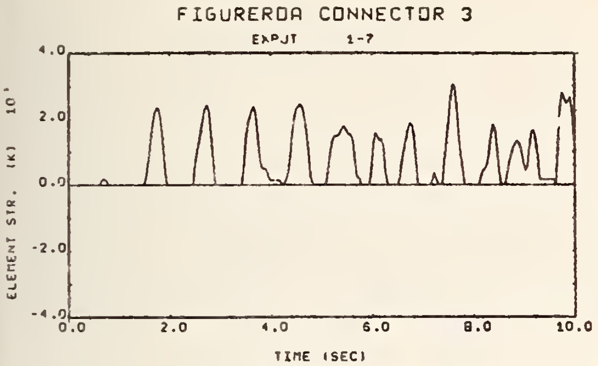


At Inner Edge of the Deck

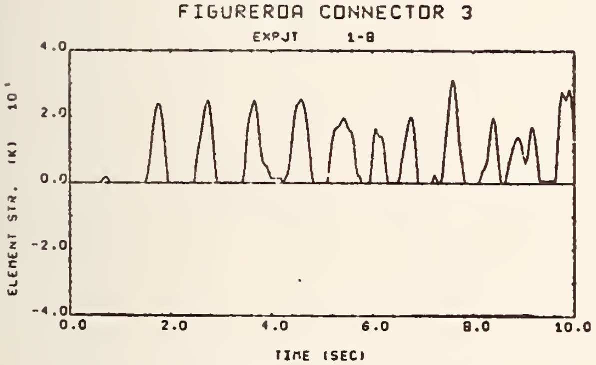


At Outer Edge of the Deck

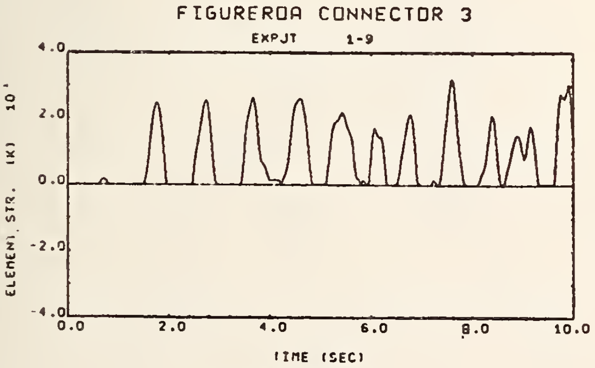
Fig. 89 Longitudinal Joint Separations at Expansion Joint # 1



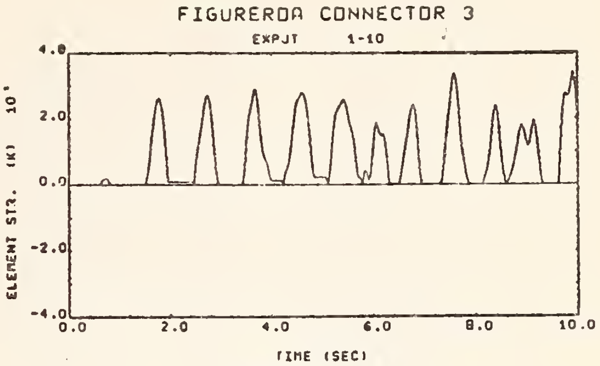
Tie Cable # 1



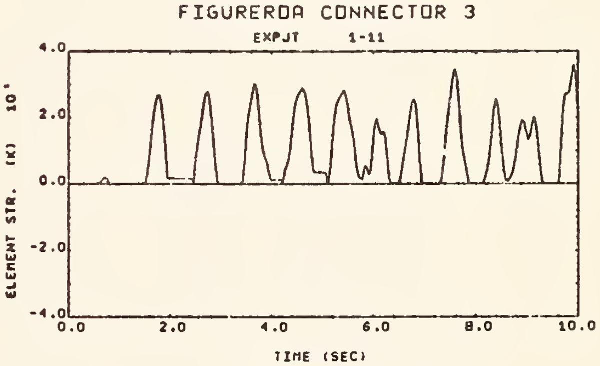
Tie Cable # 2



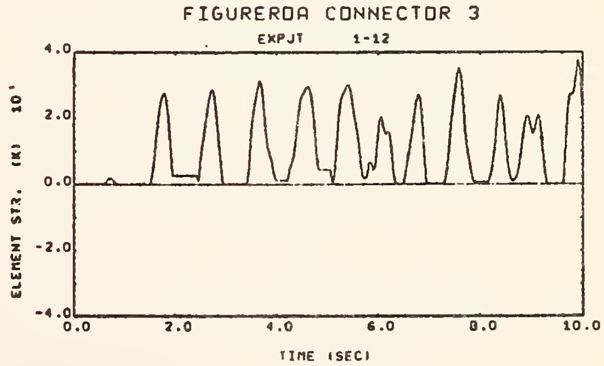
Tie Cable # 3



Tie Cable # 4



Tie Cable # 5



Tie Cable # 6

Fig. 90 Longitudinal Tie Cable Forces at Expansion Joint # 1

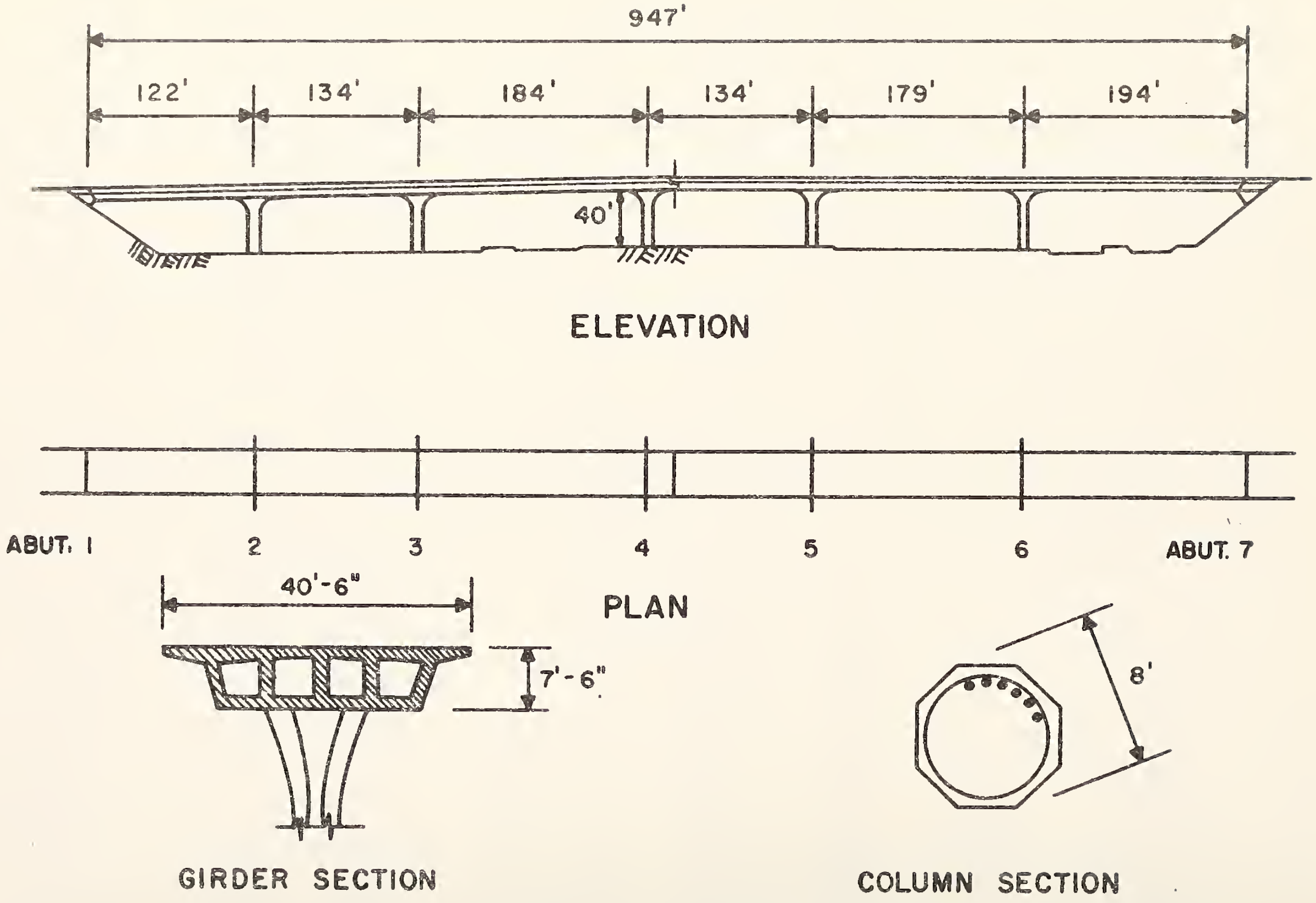


Fig. 91 The Structural System of the Straight Figueroa Street U. C. Connector

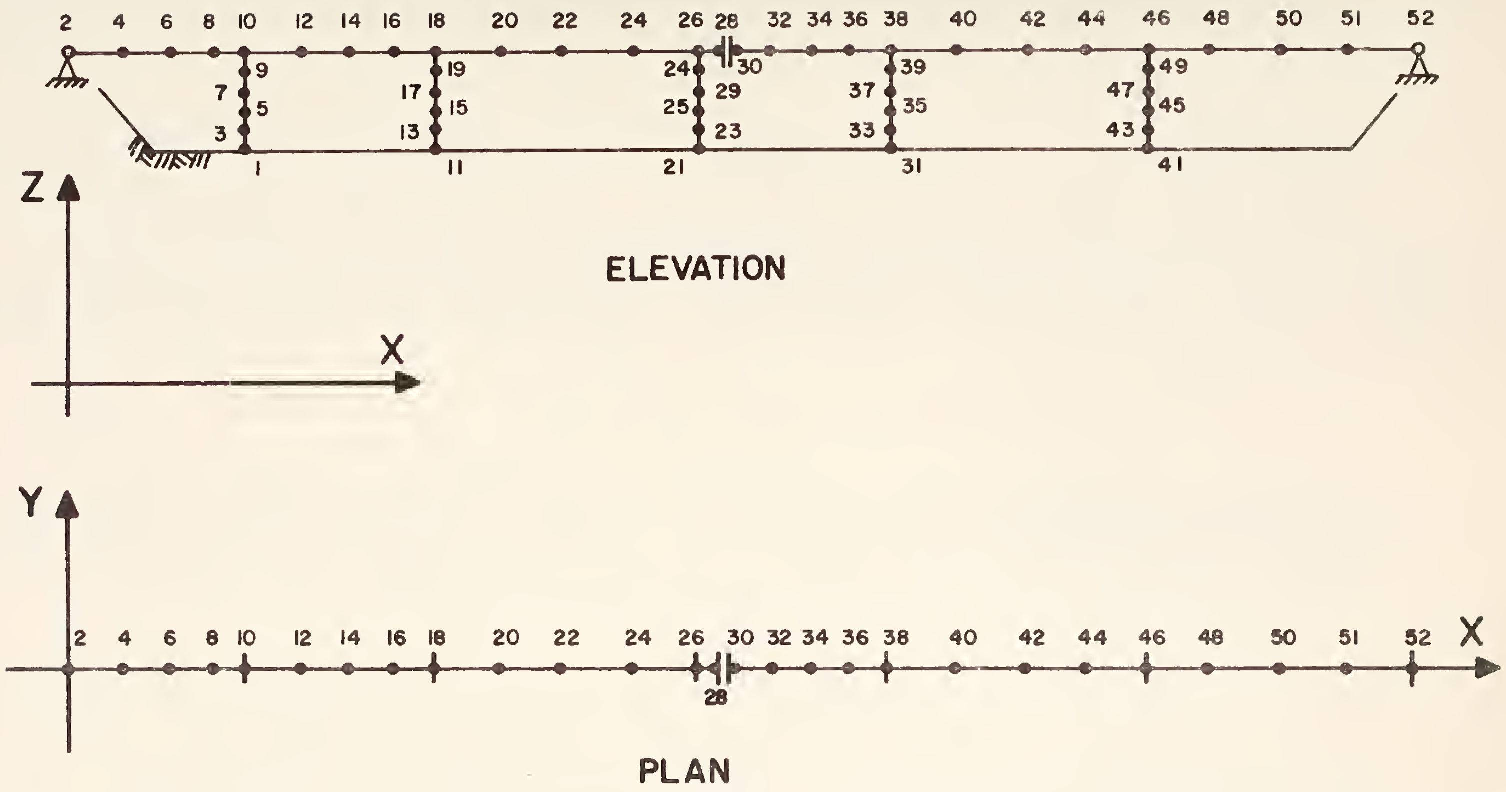


Fig. 92 Lumped Parameter System of the Straight Figueroa Street U. C. Connector

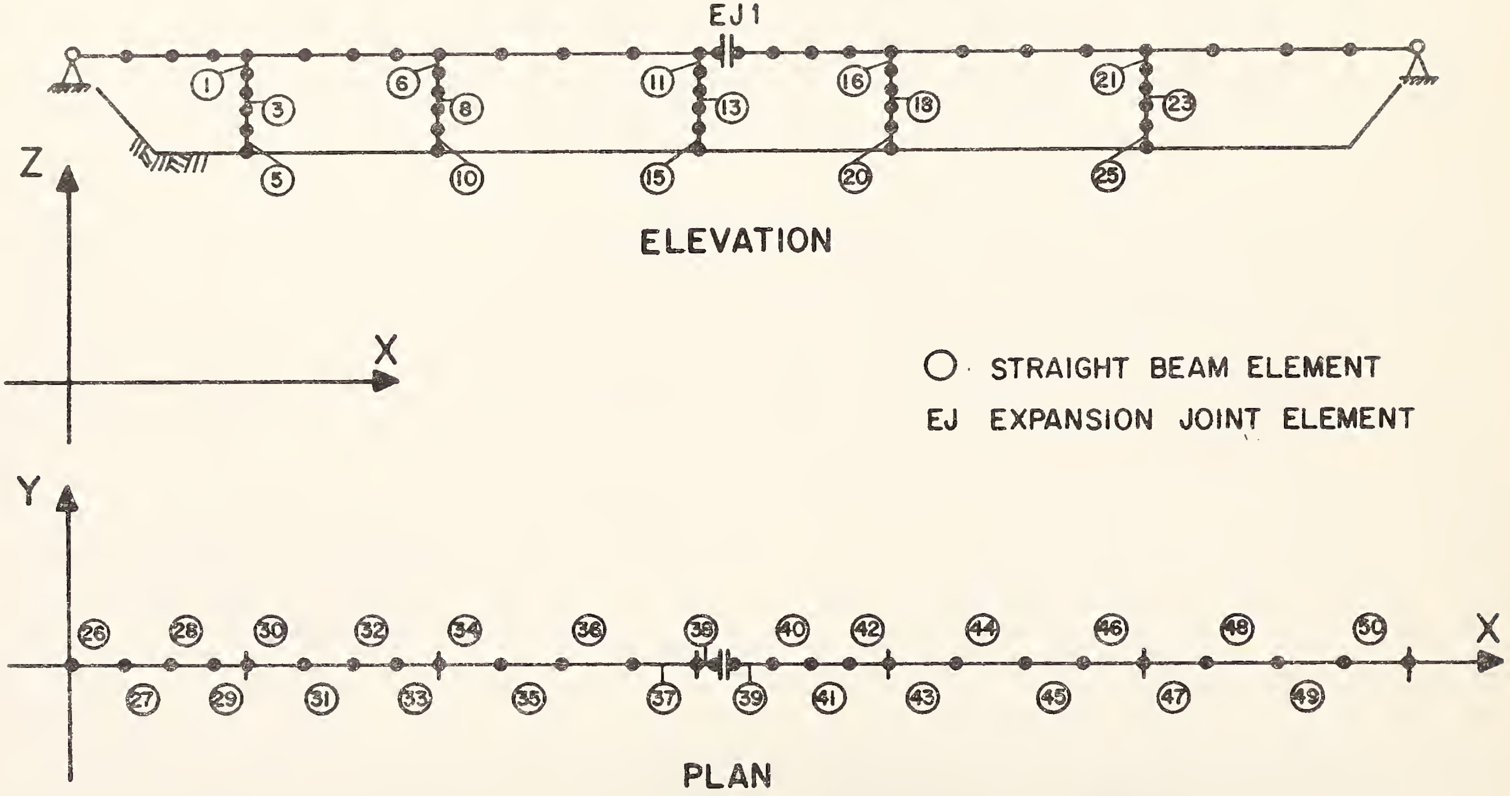


Fig. 93 Finite Element Model of Straight Figueroa Street U. C. Connector

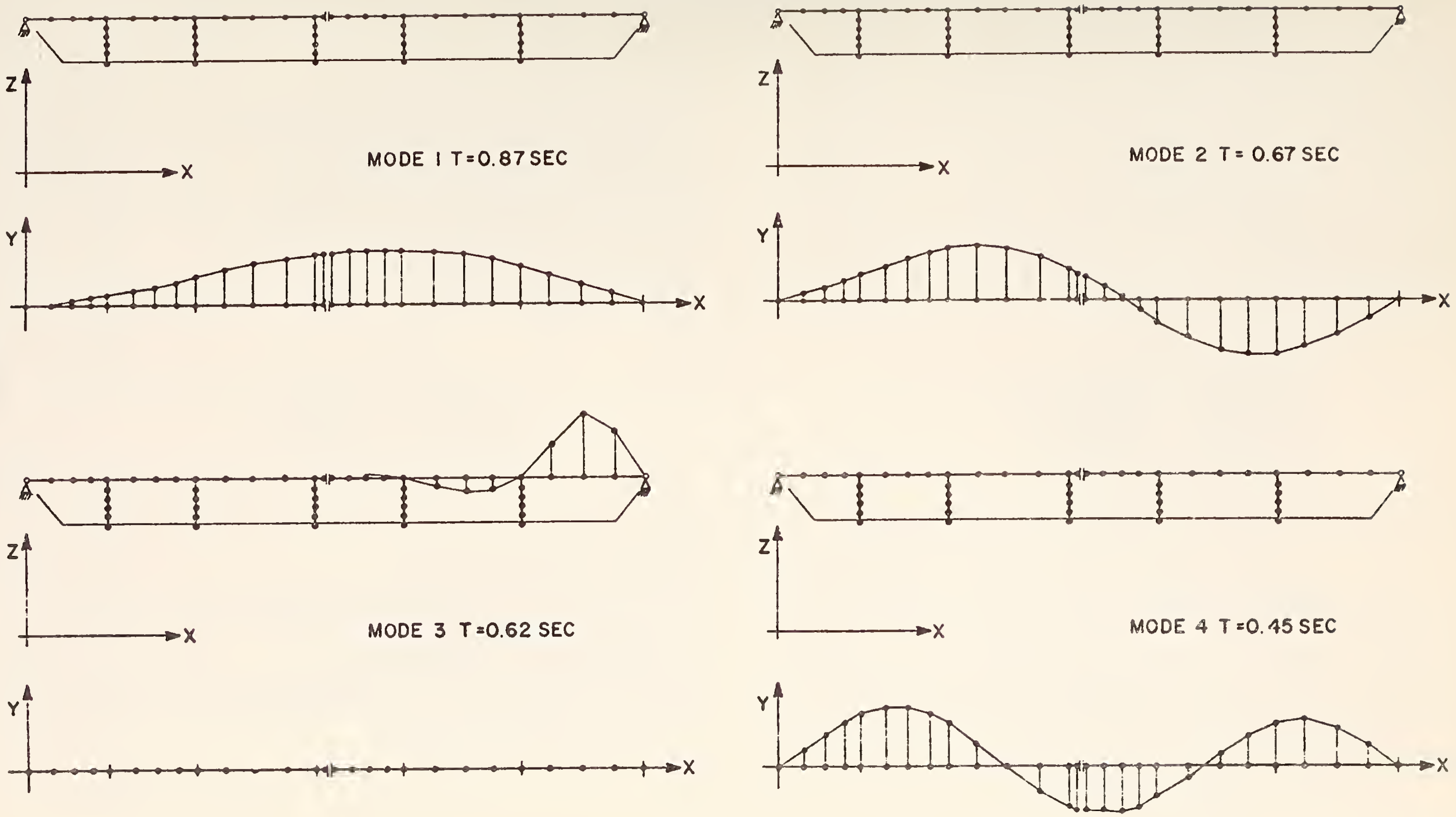
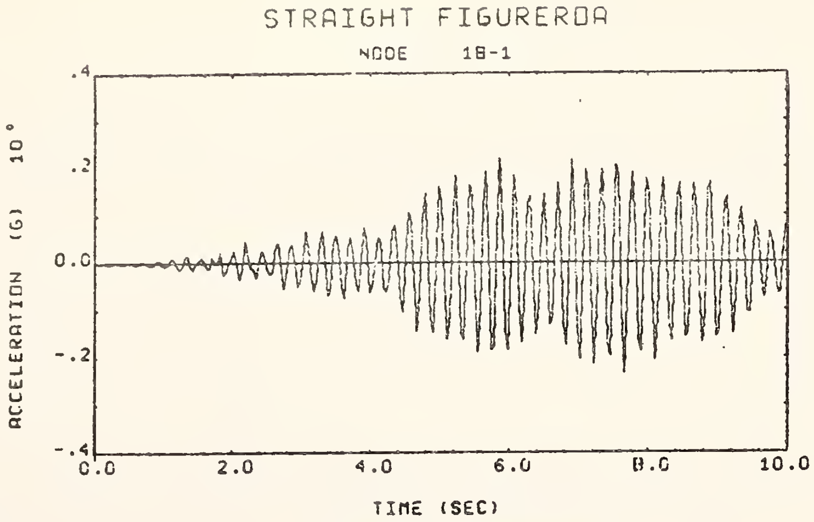
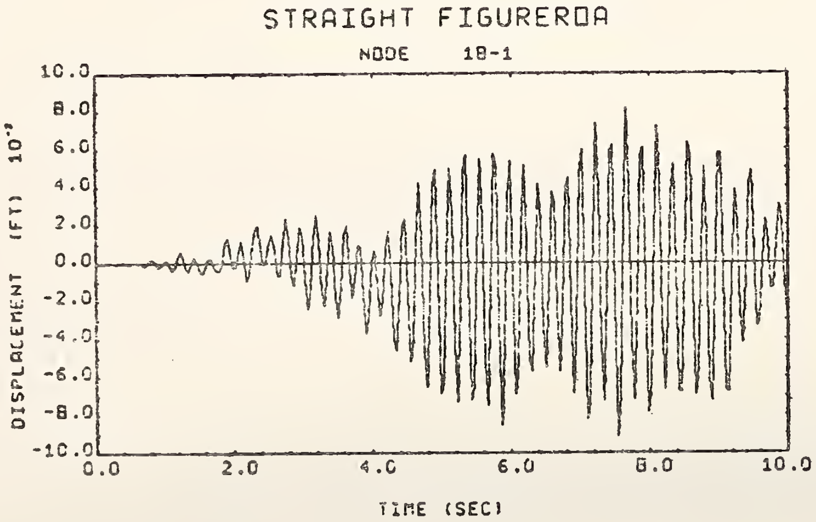
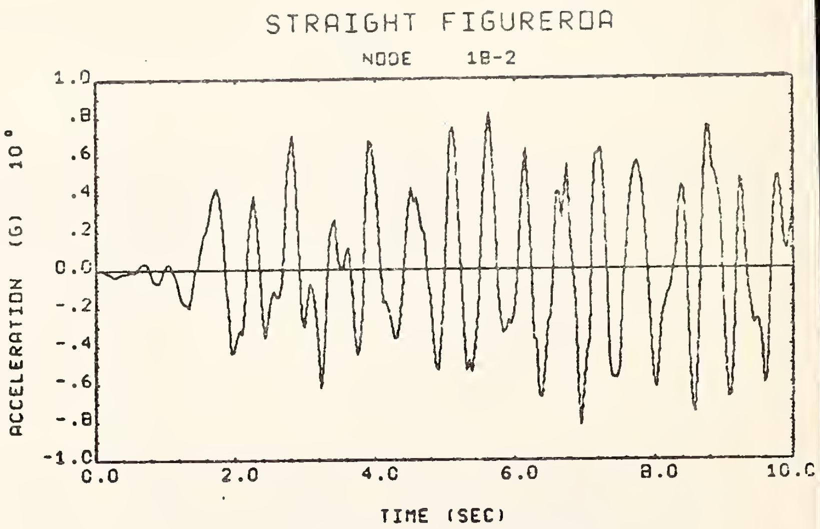


Fig. 94 Mode Shapes of the Straight Figueroa Street Undercrossing Connector: Zero Friction at the Expansion Joint



Global X - Component





Global Y - Component

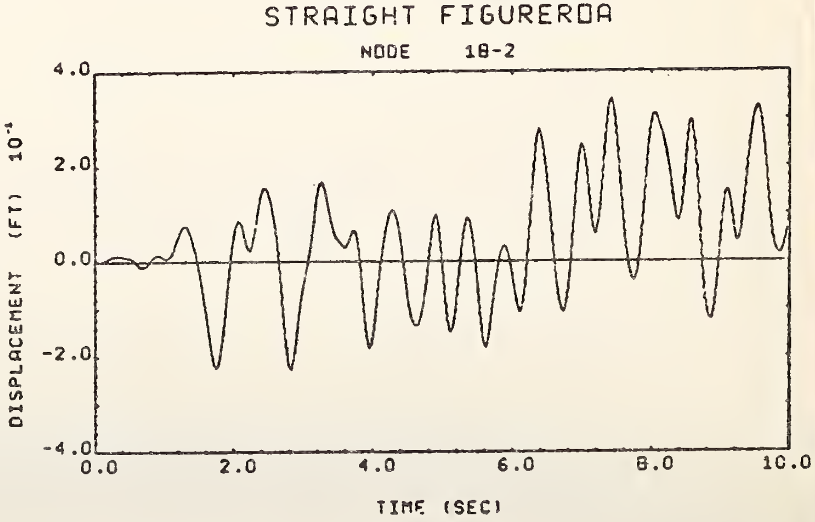
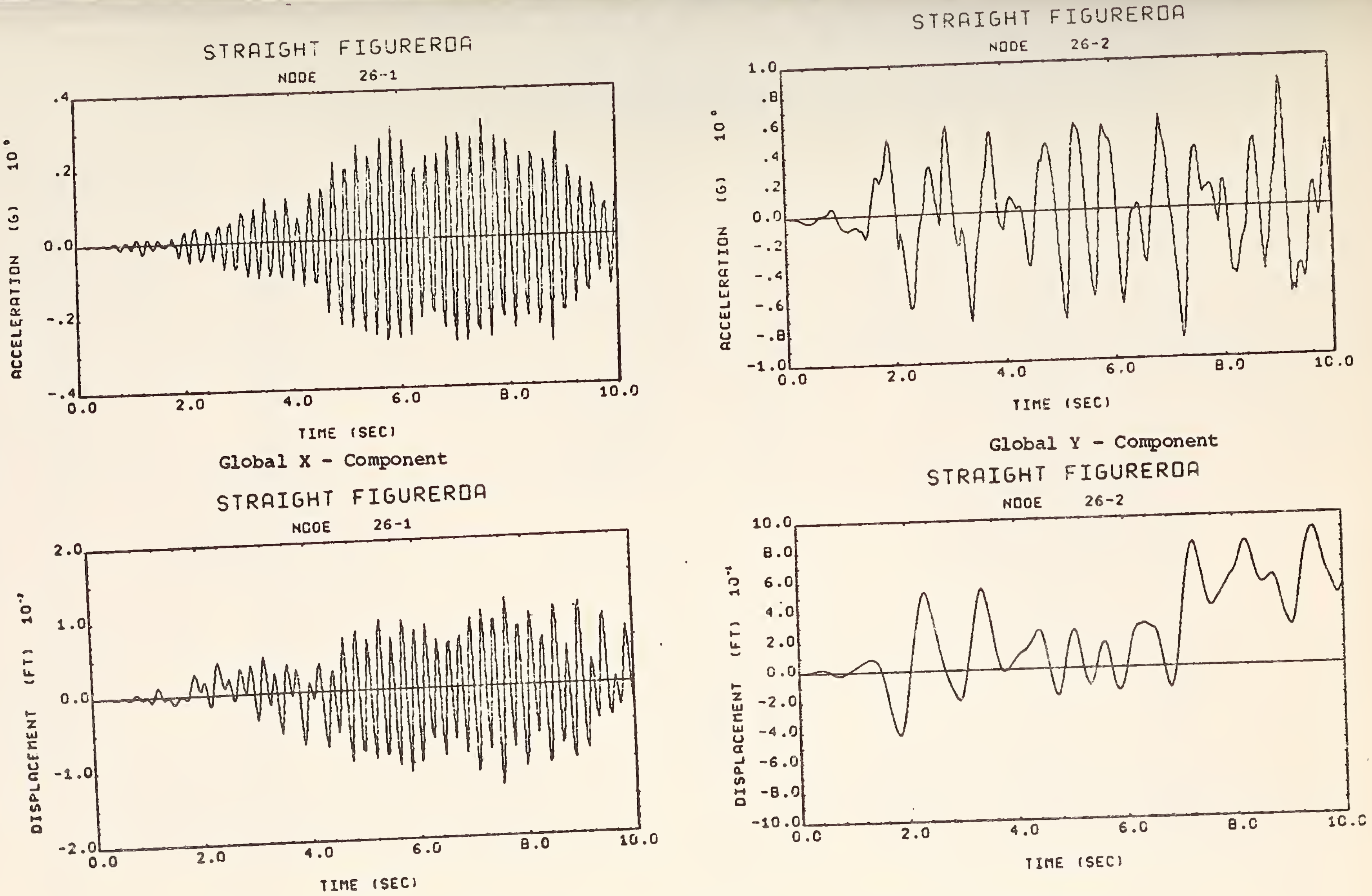


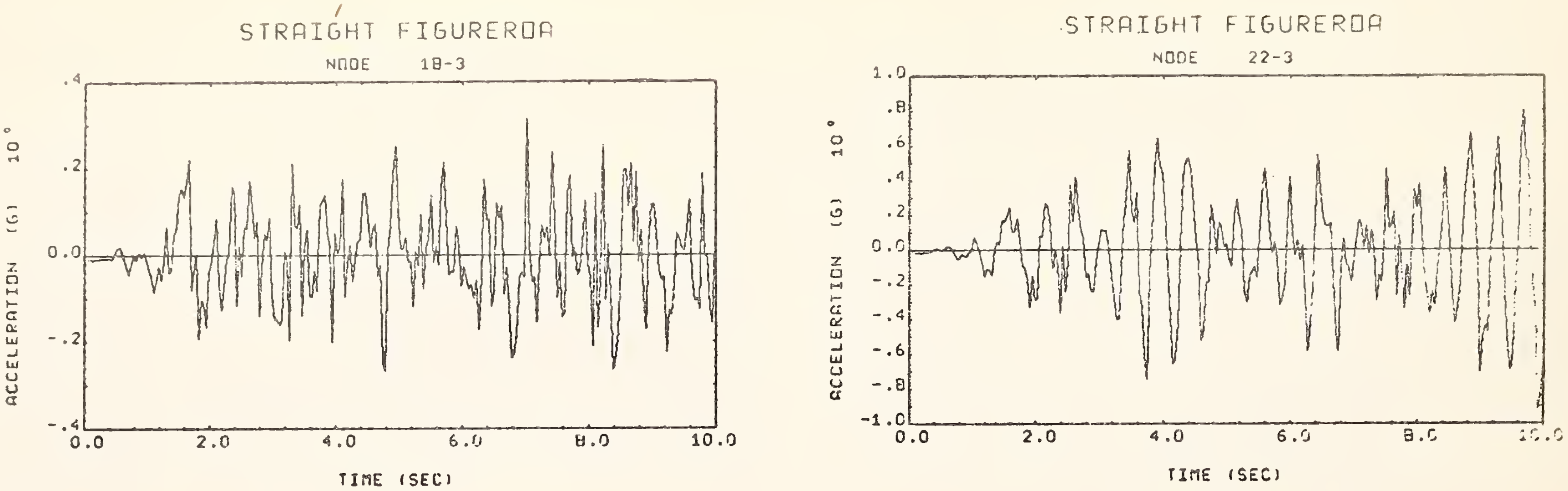
Fig. 95 Horizontal Accelerations and Displacements at the Top of Column # 3



Global X - Component

Global Y - Component

Fig. 96 Horizontal Accelerations and Displacements at the Top of Column # 4



Tie Column # 3

Center of Span # 3

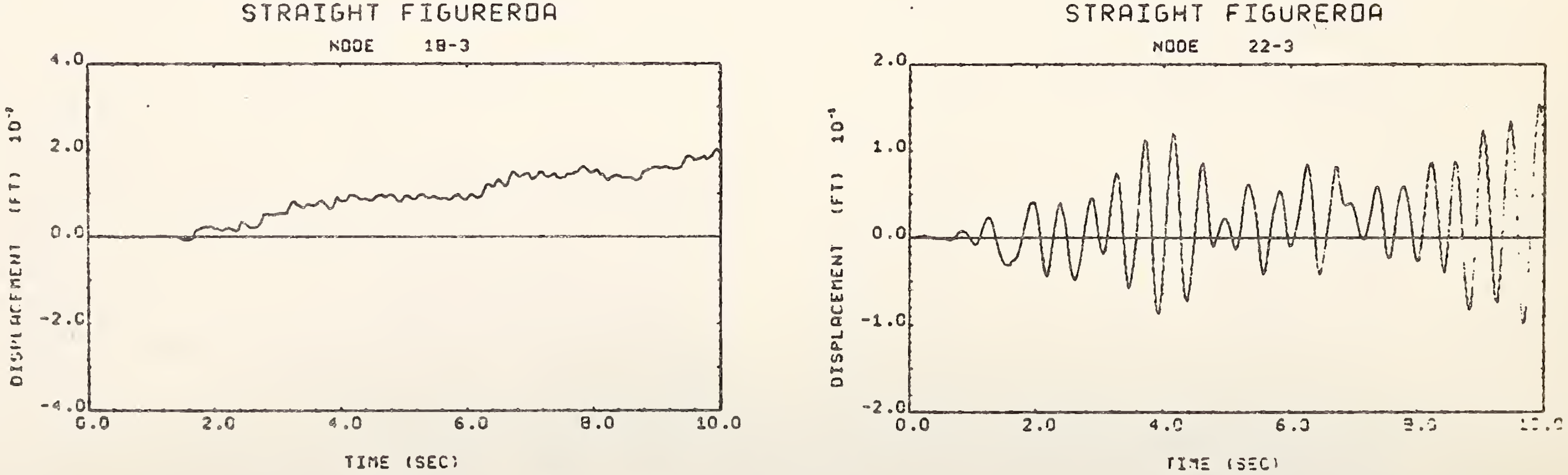
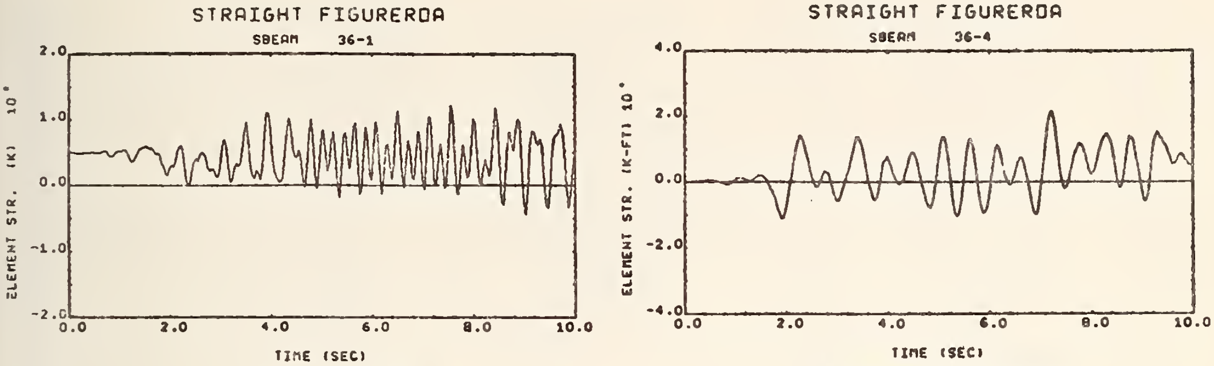
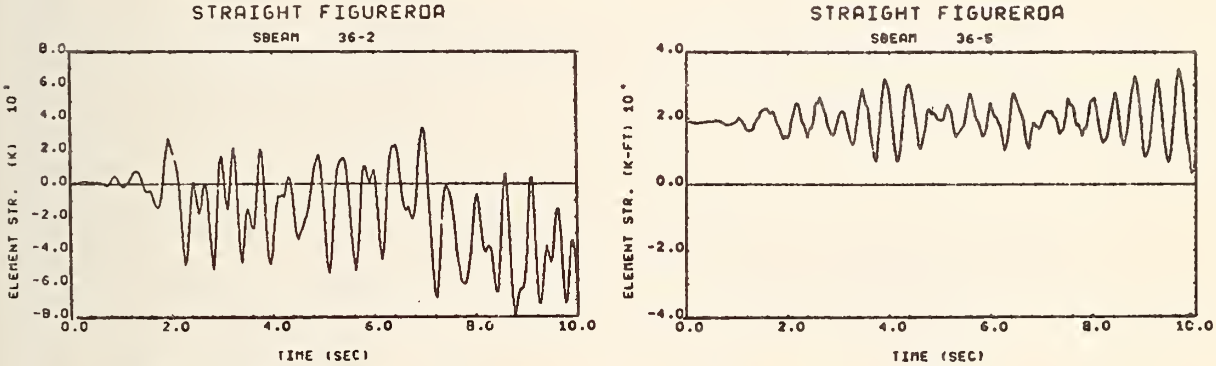


Fig. 97 Vertical Accelerations and Displacements at the Top of Column # 3 and the Center of Span # 3



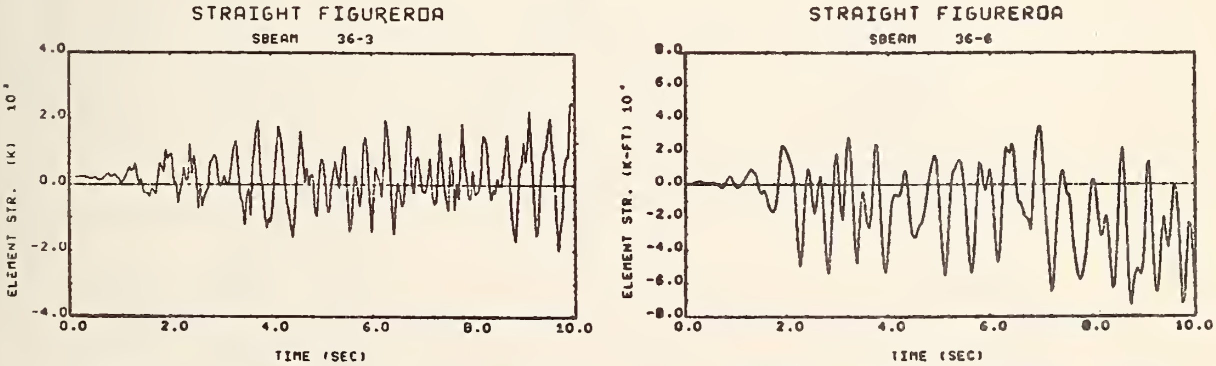
Axial Force

Torsional Moment



Shear in the Local y - Direction

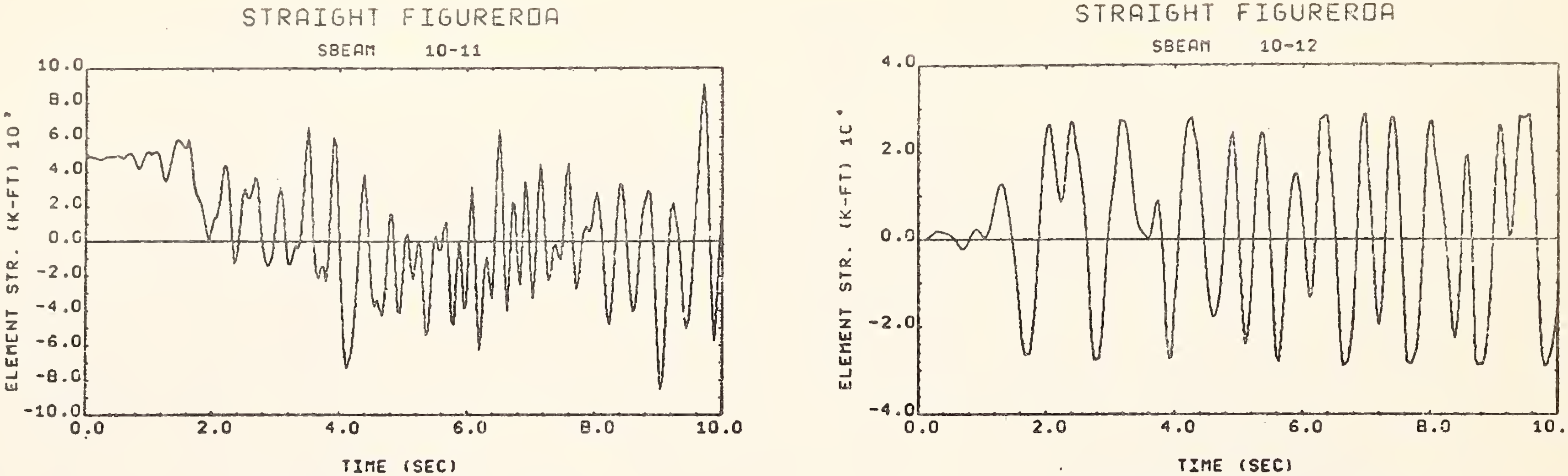
Moment about the Local y - Axis



Shear in the Local z - Direction

Moment about the Local z - Axis

Fig. 98 Generalized Forces in the Girder at the Center of Span # 3



About the Local y - Axis

About the Local z - Axis

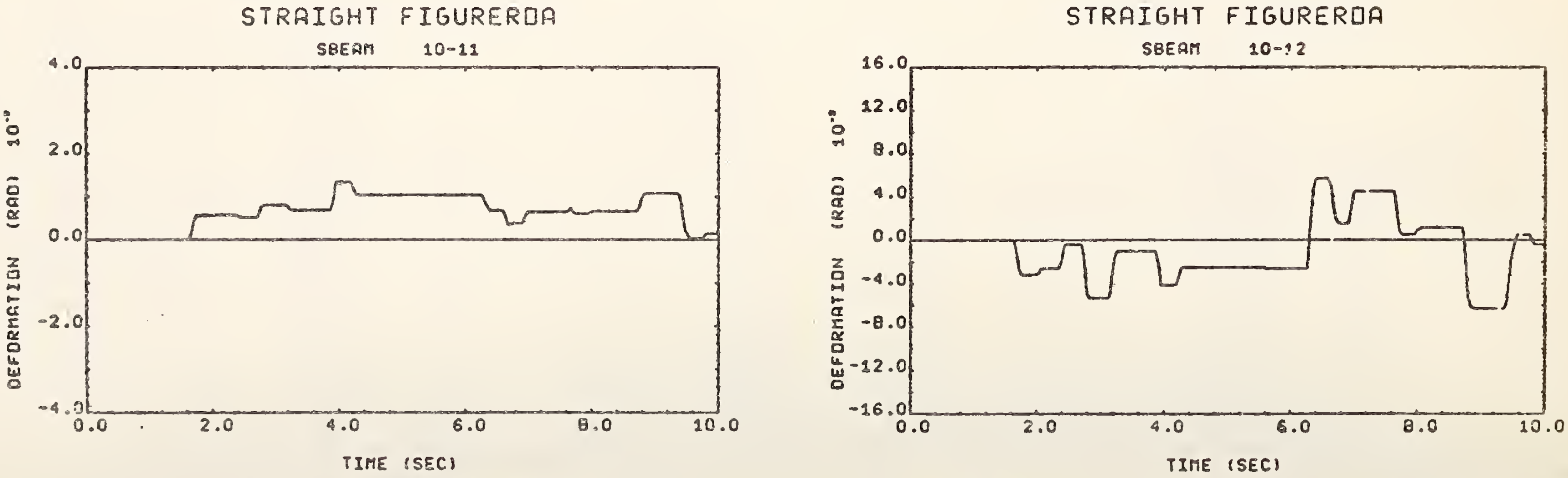
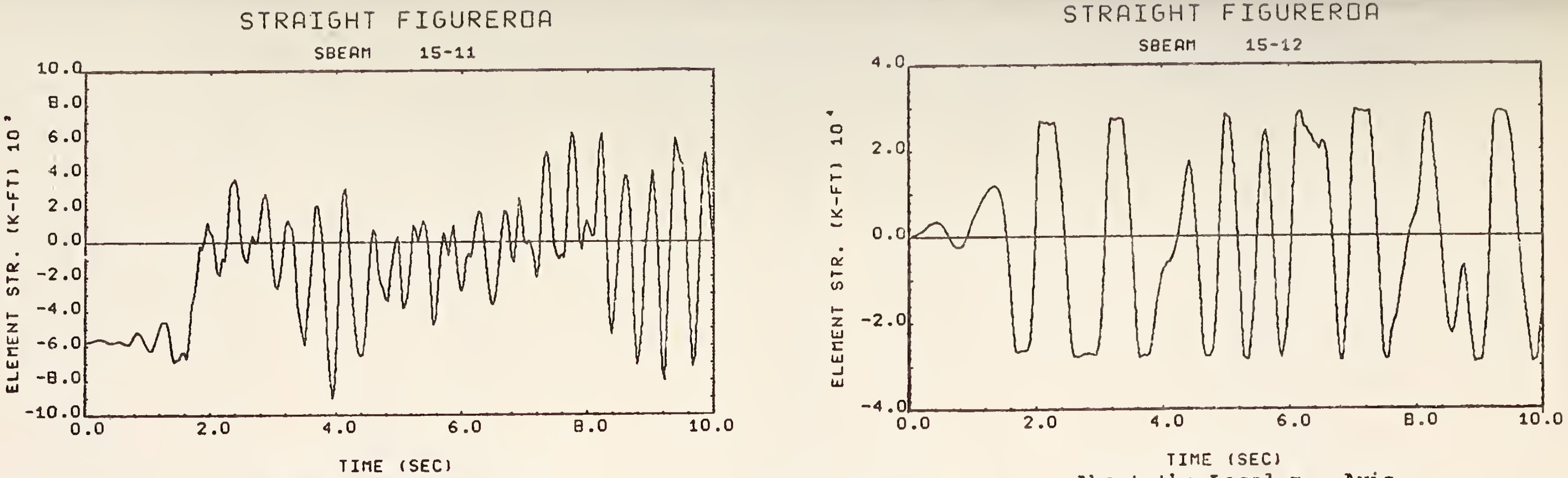


Fig. 99 Bending Moments and the Corresponding Plastic Rotations at the Base of Colum # 3



About the Local y - Axis

About the Local z - Axis

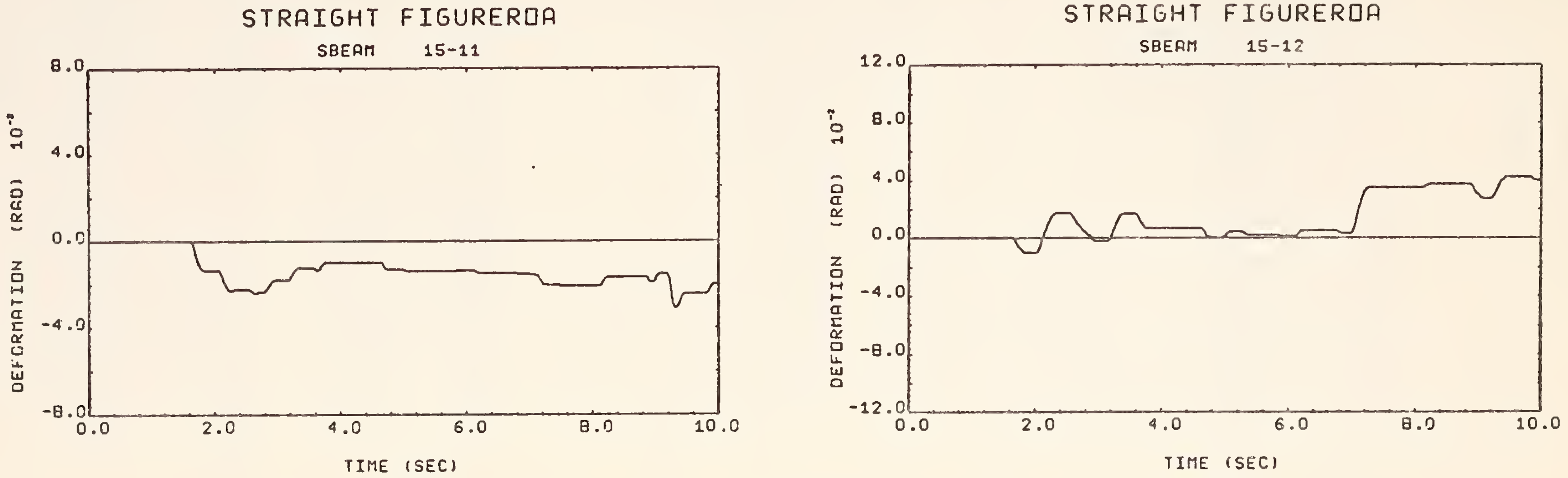
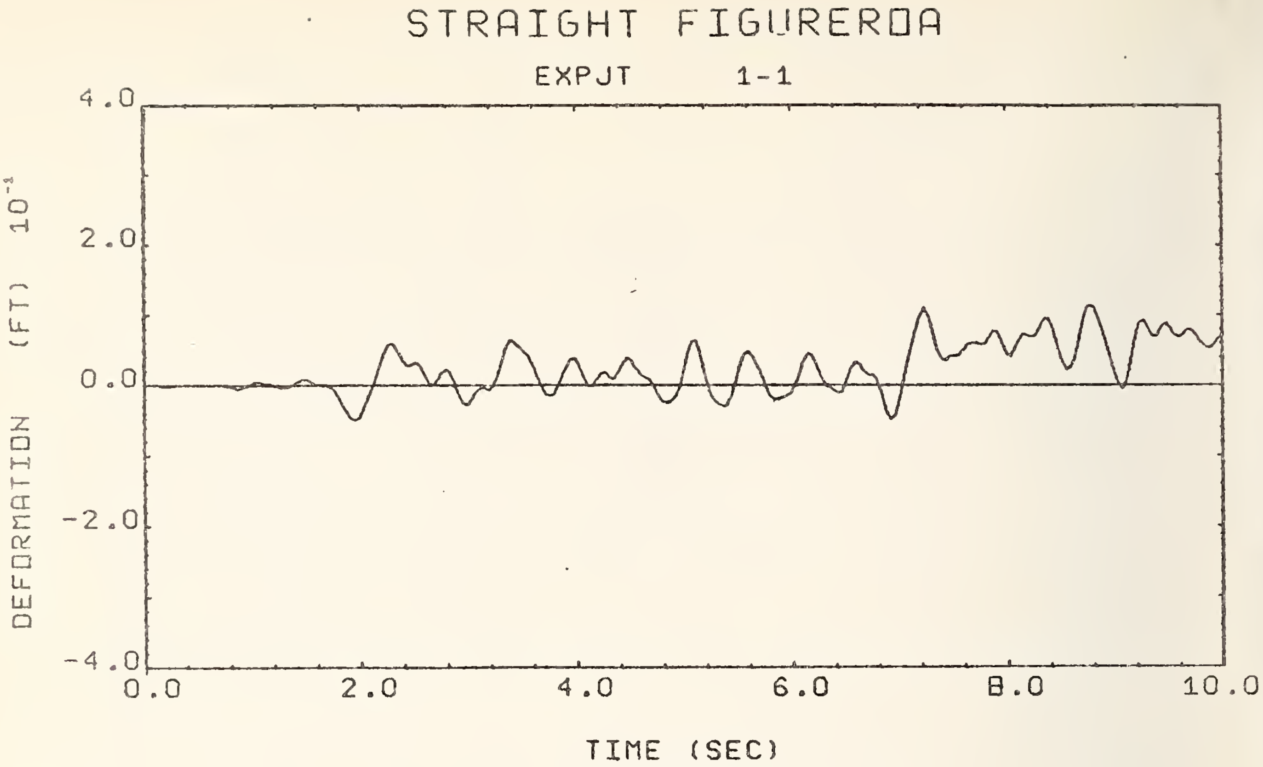
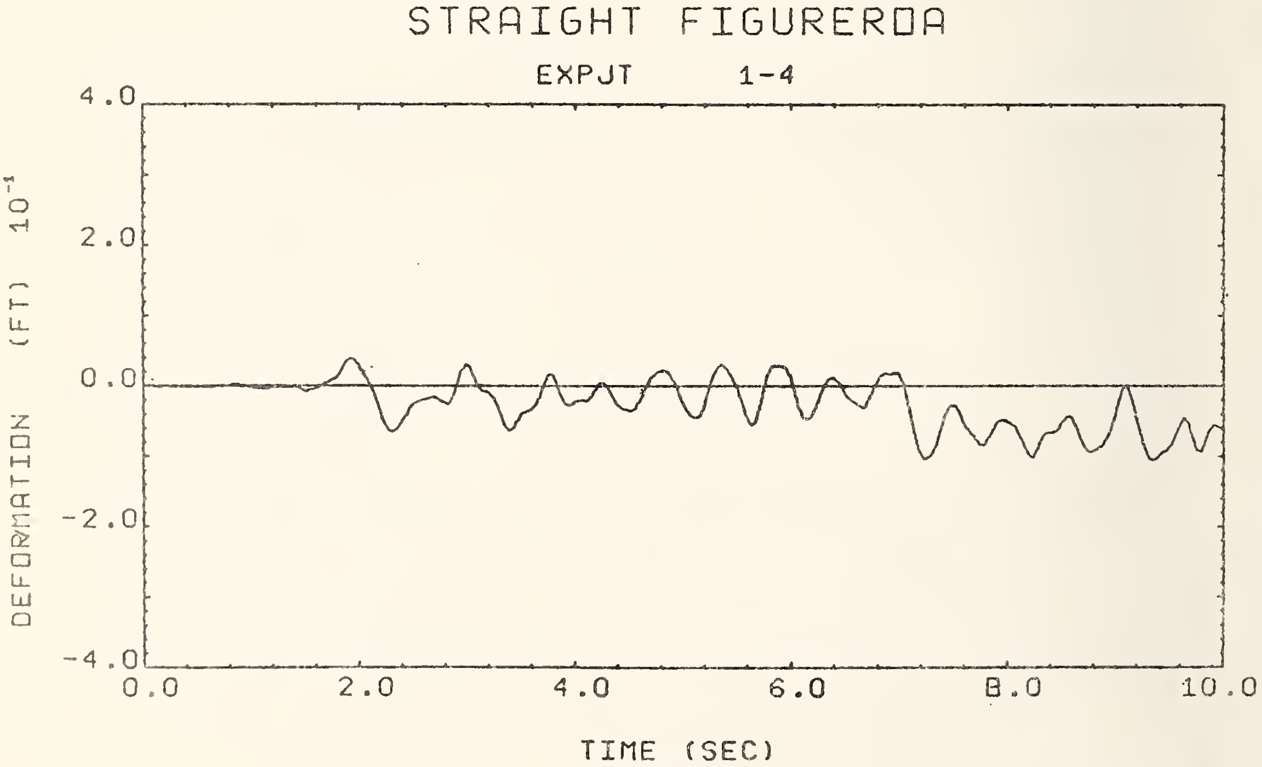


Fig. 100 Bending Moments and the Corresponding Plastic Rotations at the Base of Column # 4

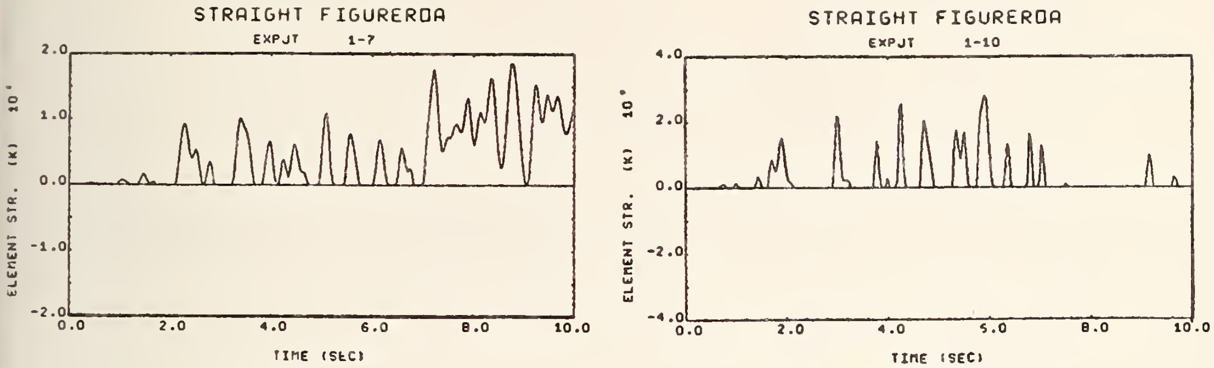


At Contact Point A



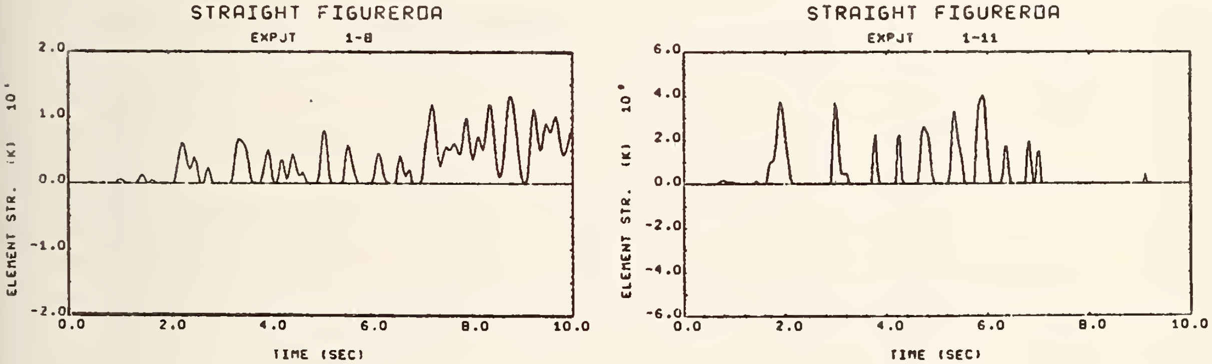
At Contact Point B

Fig. 101 Longitudinal Joint Separations at Expansion Joint # 1



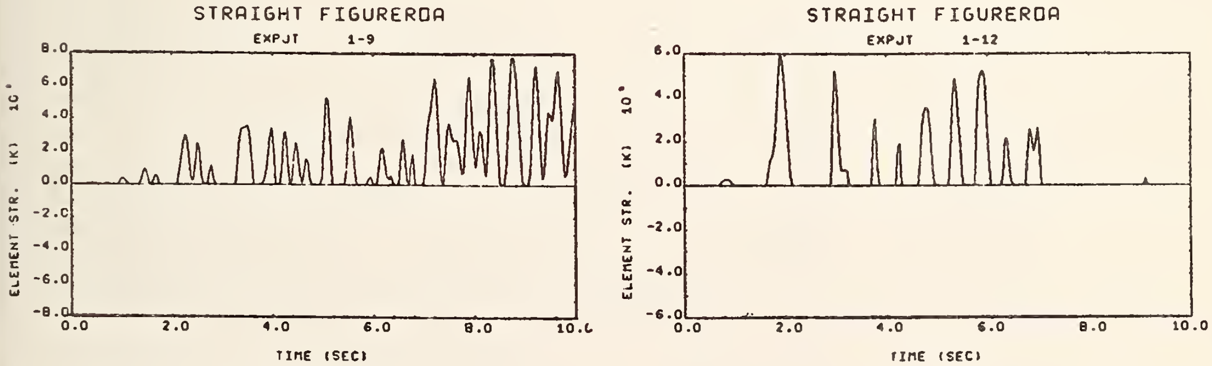
Tie Cable # 1

Tie Cable # 4



Tie Cable # 2

Tie Cable # 5



Tie Cable # 3

Tie Cable # 6

Fig. 102 Longitudinal Tie Cable Forces at Expansion Joint # 1

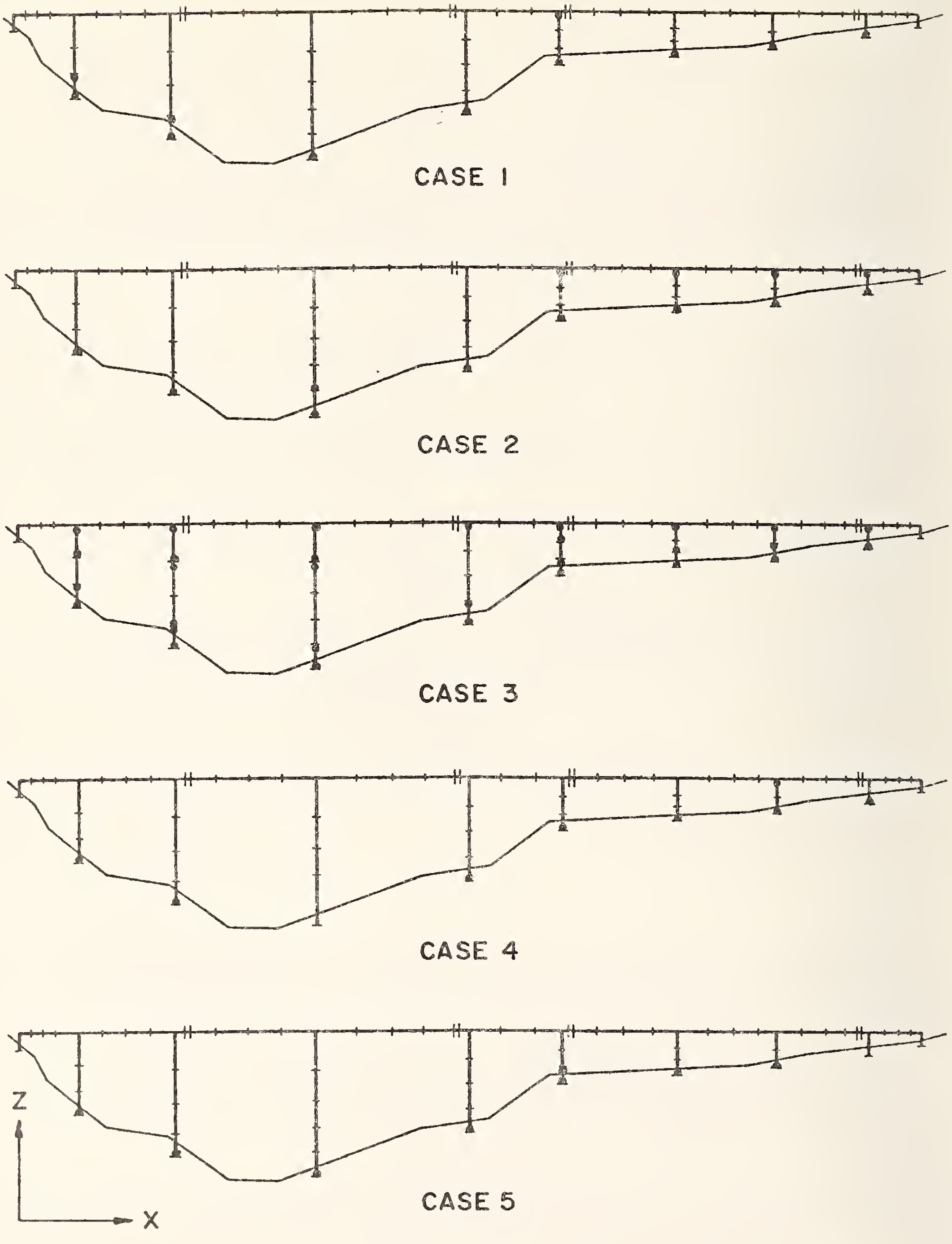


Fig. 103 Location of Maximum Flexural Yielding in Columns of 5/14 South Connector Overcrossing

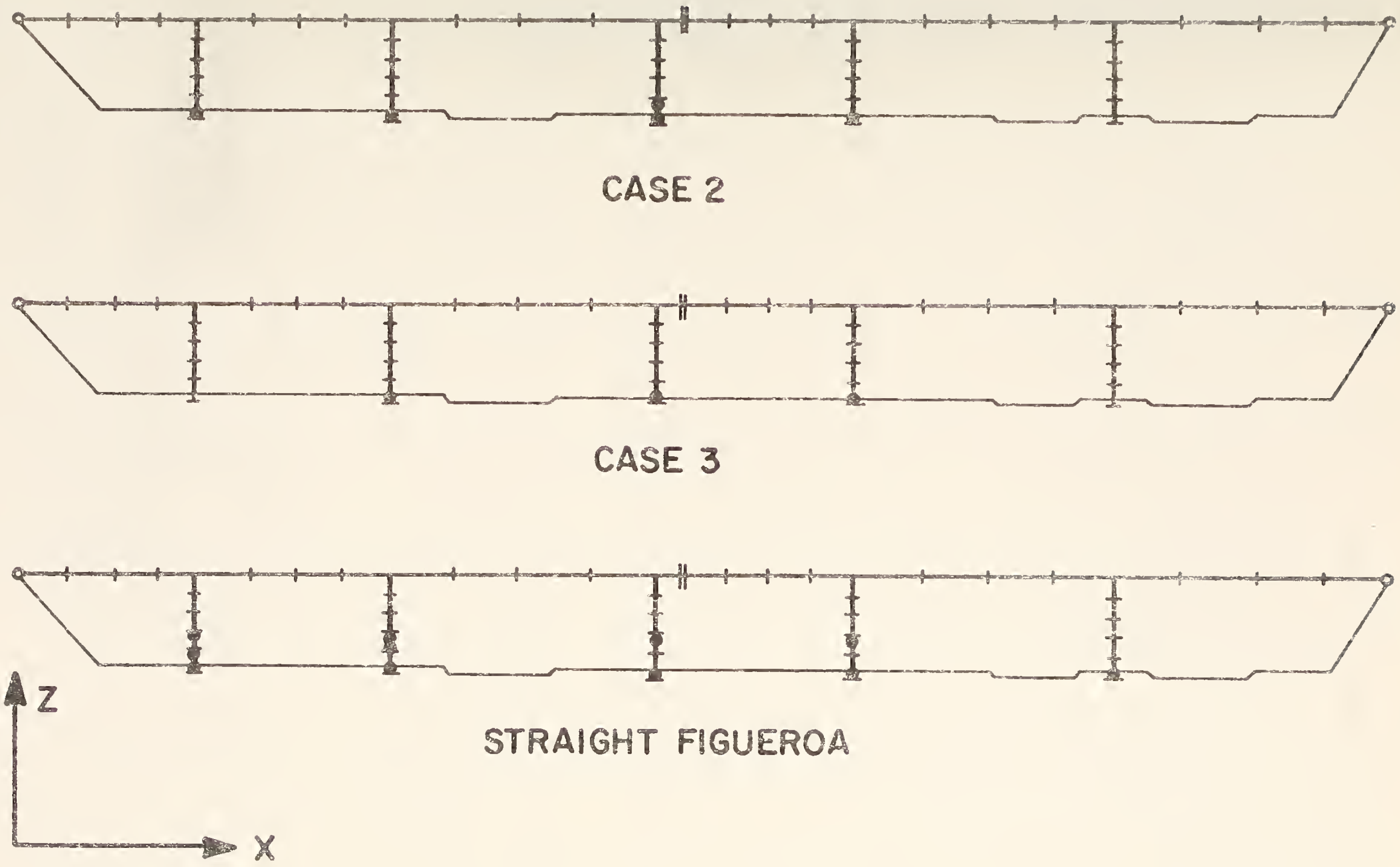


Fig. 104 Location of Maximum Flexural Yielding in Columns of the Curved and Straight Figueroa Street Undercrossing Connector

VIII DISCUSSION OF NUMERICAL RESULTS

A. SEISMIC RESPONE CHARACTERISTICS OF BRIDGE SYSTEMS

The analytical results previously presented for the 5/14 South Connector Overcrossing showed that this bridge experienced large amplitude response when subjected to severe ground shaking (Cases 1, 2, 3, and 5). For Case 2 which uses weak short tie bars, the longitudinal deck separations exceeded the supporting ledge dimension (1.25 ft.) at expansion joints Nos. 1 and 2; see Table 27. The magnitudes of these separations correspond to tie bar ductility factors far in excess of those actually provided; thus, early failures of these bars would result allowing high amplitude response similar to Case 3 (no tie bars) to develop. This high amplitude response would result in collapse of the structure.

For Cases 1 and 5 which use strong short tie bars, the joint separations never exceeded the supporting ledge dimension even under conditions of severe shaking; thus, collapse of the structure in these 2 cases would not occur if the required tie ductilities are provided.

The analytical results presented for the curved Figueroa Street Undercrossing Connector (Case 2) showed relatively high amplitude acceleration response for the condition of severe excitations. Although this response causes some flexural damages at the bases of columns supporting the central spans, the oscillatory motions were quite stable. Due to the fact that the columns are relatively short and only one expansion joint is present in the deck, the deck displacements and the expansion joint separations are relatively small. The longitudinal restrainer cables at the joint never reach their yield condition (0.64 ft. elongation) even under severe shaking. The maximum joint separation in this case was 0.3 ft.

The horizontal time histories at the tops of all columns for all bridge structures examined show that all segments of deck between expansion joints move essentially as rigid bodies in the horizontal directions, i.e, transverse bending effects in the deck are small. Therefore, the deck segments vibrate horizontally almost as rigid bodies linked together at the expansion joints. The principal mode of vibration resulting from transverse excitation is similar to the transverse fundamental mode for the linear zero friction case, i.e., similar to mode No. 2 shown in Fig. 17 for the 5/14 South Connector Overcrossing and mode No. 1 shown in Figs. 68 and 94 for the curved and straight Figueroa Street Undercrossing Connector, respectively.

It should be noted that for curved bridges, the transverse fundamental mode of vibration caused large forces to develop in the restrainer tie bars. Therefore, the properties of the restrainer tie bars greatly affect the transverse seismic responses of such bridges. These effects can be observed by comparing the results for Case 1 and 2 of the 5/14 South Connector Overcrossing.

The overall horizontal displacement responses for all cases of the 5/14 South Connector Overcrossing indicate drifts of the displacements in directions away from the centers of curvature of the decks, i.e., outward motions. This characteristic behavior is due to (1) the relatively high stiffnesses which develop as expansion joints close with inward displacements causing the deck to act as a continuous stiff arch from abutment to abutment, and (2) the relatively low stiffnesses which remain with outward displacements. For outward displacements, the stiffnesses are contributed only by the actions of columns and restrainer tie bars.

The drifts of horizontal displacements in the outward directions are particularly noticeable for those cases of loading which cause plastic deformations to occur in the columns and in the restrainer tie bars. Because of this biased behavior, the plastic deformations tend to accumulate in the outward direction. As a result, the entire bridge structure acts as a "shake-down" system.

In all cases of the curved Figueroa Street Undercrossing Connector, the overall horizontal displacement response was oscillatory and approximately equal in both inward and outward directions. The plastic flexural rotation at the column bases was also oscillatory and did not accumulate in the outward direction as in the case of 5/14 South Connector Overcrossing. The stable response of the curved Figueroa Street Undercrossing is due to the fact that arch action of the deck does not occur because the amplitude of inward motions is insufficient to cause closure of the large (6 in.) joint gap in this bridge as indicated in Figs. 74, 82 and 89.

From the above comparison, it appears that for curved bridges, a bigger joint gap is desirable in order to avoid biased seismic responses and to prevent the shake-down situation under severe shaking conditions.

B. FLEXURAL YIELDING OF BRIDGE COLUMNS

For both 5/14 South Connector Overcrossing and Figueroa Street Undercrossing Connector, the location of the maximum flexural yielding in the columns (Figs 103 and 104) showed that under moderate shaking conditions flexural yielding was more likely to occur at the base of the columns (Case 4 of Fig. 103 and Case 3 of Fig. 104). Under severe shaking conditions, flexural yield hinges also developed at the top of the prismatic columns (Cases 1, 2, 3, and 5 of Fig. 103 and Cases 1 and 2 of Fig. 104). Columns located near expansion joints were shown to be more likely to develop yield hinges because of the relatively higher lateral displacement resulting from the fundamental transverse mode of vibration.

The damages due to flexural yielding at the top of a column will penetrate the deck and hence should be prevented. An effective method of achieving this is by flaring the upper portion of the column as in the case of Figueroa Street Undercrossing Connector.

The maximum local bending ductility factors at the base of some columns under severe shaking substantially exceeded the ductility that could actually be provided, however, under moderate shaking they remained within tolerable level (<5) which could be accommodated by well-designed columns. In all cases, the ductility factors required by shorter columns are higher than for the longer columns. This results from the fact that deck segments vibrate laterally almost as rigid bodies causing similar lateral displacements of all columns and hence higher ductility requirements of the shorter columns.

The results of Case 1 (elastic columns) and Case 2 (elasto-plastic columns) of the curved Figueroa Street Undercrossing Connector indicate that elastic columns substantially increase the acceleration response in the deck without significantly affecting the displacement responses of the system.

C. EXPANSION JOINT RESTRAINERS

The results of Cases 1, 2, and 3 for 5/14 South Connector Overcrossing illustrate the effectiveness of longitudinal restrainers in reducing the seismic response of the bridge. It is clear from these results that longitudinal restrainers at expansion joints are essential in curved bridges with several expansion joints. The results of Case 2 illustrates the analytical inadequacy of the weak short tie bars, a fact exemplified by the bridge during the San Fernando earthquake. Cases 1 and 5 with strong short tie bars indicate sufficient strengths and stiffnesses but inadequate ductilities of these bars. The results of Table 27 showed that the required tie ductility factors of the strong short ties greatly exceeded the levels that these ties can accommodate even under moderate shaking (Case 4 in Table 27). Hence short tie bars are unsuitable for use as longitudinal restrainers at the expansion joints of long curved bridges because of insufficient reserved ductility.

The long cable longitudinal restrainers used at the expansion joint of Figueroa Street Undercrossing Connector had both sufficient strength and ductility to resist severe shaking conditions (Cases 1, 2 and straight). In all cases, the results showed that these cables did not experience yielding indicating that the length of the cables is more than required for resisting severe shaking. A reduction of 2 in length would have still enabled the cables to resist severe shaking elastically.

The required length L_T of a longitudinal restrainer tie can be estimated from the expansion joint ledge dimension Δ_S and the yielding strain ε_T^Y of the tie by the relation

$$L_T = \frac{\Delta_S - \Delta_T}{\mu_T \ \varepsilon_T^Y} \tag{116}$$

where Δ_T is the tie gap and μ_T is the tie ductility factor (Eq. 115) which is a function of its material and size. Δ_S can be estimated from the maximum horizontal displacement response of the bridge and the geometry of the transverse fundamental mode of vibration for the linear zero friction case.

The purpose of using vertical restrainers at expansion joints is to prevent spans lifting from their supports. An examination of the time history response of the forces in the vertical restrainers at expansion joints Nos.1 and 2 of Cases 1 and 2 of 5/14 South Connector Overcrossing as shown in Figs. 34 and 45, indicates that under severe shaking tensile forces occur, although infrequently. The possibility of tensile forces in the vertical restrainers indicates the desirability of the use of the restrainers to prevent vertical separation of spans at the expansion joint.

D. INFLUENCE OF DECK CURVATURE

The influence of deck curvature on the dynamic characteristics of bridges was illustrated by the mode shapes of vibration shown in Figs. 68 and 93 for the curved and straight Figueroa Street Undercrossing Connector. It is clear from the results presented that horizontal curvature of the deck causes coupling between the two horizontal components, and superelevation of the deck causes coupling between vertical and horizontal components.

The effect of deck curvature on the seismic response of the bridges was shown in the time history responses presented for the straight and Case 2 of the curved Figueroa Street Undercrossing Connector (Figs 84-88 and 76-81). Ground excitation in the transverse direction causes greater response in the straight bridge which in part is attributable to the greater lateral stability of the curved bridge resulting from the curved distribution of supporting columns and abutments.

E. INFLUENCE OF VERTICAL GROUND ACCELERATIONS

Vertical ground accelerations have an important effect on the vertical response of all bridges. They also significantly affect the horizontal response of curved bridges (Cases 1 and 5 of 5/14 South Connector Overcrossing) because of the coupling between the horizontal and vertical modes of vibration.

Vertical ground accelerations cause large vertical oscillatory motions in the deck resulting in a removal of the vertical compressive reaction at the expansion joints. This in turn removes the horizontal Coulomb friction force which affects the horizontal resistance of the structure. The vertical accelerations also change the axial load in the columns and therefore affect the ultimate bending strength of the columns due to yielding interaction effect.

F. INFLUENCE OF COULOMB FRICTION AT EXPANSION JOINTS

The results of modal analyses presented for 5/14 South Connector Overcrossing (Figs. 17 and 18) clearly showed that infinite friction at the expansion joints substantially increased the lateral stiffnesses of the bridge when compared to the case with zero friction. The case of zero friction is considered to be more practical and realistic, because the Coulomb friction force corresponding to the friction coefficient normally existing at expansion joints is relatively small when compared to the axial force which would result from the arch action of the deck in the infinite friction case.

Because Coulomb friction force is relatively small and acts only at a limited number of expansion joints, its influence on the total bridge response is small.

G. INFLUENCE OF OVERALL STRUCTURAL ARRANGEMENT

Although longitudinal restrainer ties are effective in reducing the amplitude of seismic response, they do not affect the dynamic characteristics of the bridge. This was illustrated by the essential similarity between the shapes of time history response of Case 1 and 2 of 5/14 South Connector Overcrossing. Longitudinal restrainers are secondary structural elements which are effective only when the primary horizontal seismic response is sufficiently large to activate them.

From the results presented for 5/14 South Connector Overcrossing and the curved Figueroa Undercrossing Connector, it is concluded that the major influence on the seismic response characteristics is the arrangement of supporting columns of each deck segment. For sufficient earthquake resistance of a bridge structure, each deck segment should be supported by at least 2 columns so that its lateral stiffness is reasonably adequate, and that stable oscillatory response of the structure can be achieved.

IX CONCLUSIONS AND RECOMMENDATIONS

A. CONCLUSIONS

Based on the analytical results presented for the bridge structures studied in this investigation, the following conclusions may be deduced:

(1) The basic mathematical models and the analytical procedures as well as computer programs developed in this investigation provide a rational and effective means for determining the seismic response of curved and straight, multiple-span, long, highway overcrossings of the type considered herein.

(2) The collapse of 5/14 South Connector Overcrossing during the San Fernando earthquake was due to the inability of the longitudinal restrainer ties to constrain the joint separation at expansion joints Nos. 1 and 2 thus causing the central spans between these expansion joints to fall off their supports resulting in the collapse of the spans along with column No. 4.

(3) The proposed curved Figueroa Street Undercrossing Connector can safely withstand severe ground shaking of an intensity similar to those experienced during the San Fernando earthquake; however, some flexural damage is likely to occur at the bases of the columns supporting the central spans.

(4) The strength and distribution of bridge columns in each deck segment between expansion joints are the primary factors which most significantly affect the seismic response characteristics of the bridge structure. Thus expansion joints in the deck should be located in a manner such that each substructural segment posses a reasonably adequate amount of lateral stiffnesses.

(5) Under severe shaking conditions similar to those experienced

during the San Fernando earthquake, the maximum horizontal acceleration response at the deck level substantially exceeded those values specified by the "Bridge Planning and Design Manual", California State Division of Highways [5] which was in effect prior to the San Fernando earthquake of February 9, 1971.

(6) Because of their effectiveness in reducing the amplitude of the seismic displacement response of curved bridges, longitudinal restrainer ties should be provided at expansion joints. They should have sufficiently high strength and ductility to constrain the joint separation within the limit of the supporting ledge dimension. Vertical restrainers should also be provided at expansion joints in order to prevent the lifting of spans from their supports.

(7) For curved bridges, a bigger expansion joint gap appears to be desirable to prevent biased seismic response and to avoid a shake-down situation under severe shaking conditions.

(8) The required expansion joint ledge dimension and joint gap can be estimated from the maximum horizontal displacement obtained from a seismic analysis and the geometry of the transverse fundamental mode of vibration.

(9) Under moderate to severe shaking, the overall seismic response of a bridge structure is considerably affected by vertical ground accelerations. The effect is particularly noticeable for curved bridges in which coupling exists between vertical and horizontal modes of vibration. Thus the vertical component of ground acceleration should always be included in a seismic analysis.

(10) For a normally proportioned bridge structure under moderate to severe shaking, flexural yielding is more likely to occur at the base of the columns. Yielding in columns should be avoided under moderate shaking but can be permitted to

limited levels under severe shaking conditions. It is therefore important that column reinforcement should be sized and detailed in such a manner that the columns can develop full yield strengths and maintain their yield capacities throughout their specified ranges of ductility. Flexural yielding should be prevented from developing at the top of columns in order to avoid damage penetrating the deck.

(11) The effect of Coulomb friction forces on the seismic response of the type of bridges considered is small, for the friction coefficient values normally existing at the expansion joints. Thus for linear analysis, a mathematical model assuming zero friction at expansion joints can be used as a realistic linear bridge model.

(12) Linear seismic response analysis provides a reasonable estimate of the maximum displacement response of a bridge system; however, substantial error may result in predicting the internal forces in the structure. Thus it is essential that nonlinear seismic analysis be carried out for major bridge structures under severe shaking conditions.

B. RECOMMENDATIONS

Due to the unique dynamic characteristics of bridge structures considered in this investigation, it is recommended that major highway overcrossings should be designed on the basis of dynamic analyses using appropriate nonlinear mathematical models to represent the structural systems. The dynamic analysis should include both vertical and horizontal components of ground motions suitable to the respective site. In determining the seismic response of long bridges, the "out-of-phasing" of the multiple support input ground motions may, in some cases, be significant. Therefore, every effort should be made to measure spatial correlations as well as time correlations for strong ground motions.

BIBLIOGRAPHY

1. Iwasaki, T., Penzien, J., Clough, R., "Literature Survey - Seismic Effects on Highway Bridges", EERC 71-11, Earthquake Engineering Research Center, University of California, Berkeley, November, 1972.

2. Iwasaki, T., Penzien, J., Clough, R., "An Investigation of the Effectiveness of Existing Bridge Design Methodology in Providing Adequate Structural Resistance to Seismic Disturbance, Phase I: Literature Survey", FHWA-RD-73-13, Federal Highway Administration, November, 1972.

3. Penzien, J., "Seismic Effects on Bridges, San Fernando Earthquake, 9, February, 1971", III Congreso Nacional de Ingenieria Sismica, Acapulco, Gro. Mexico, November, 1971.

4. Jennings, P. C., "Engineering Features of the San Fernando Earthquake, February 9, 1971", EERL 71-20, Earthquake Engineering Research Laboratory, California Institute of Technology, June, 1971.

5. "Bridge Planning and Design Manual", California State Division of Highways, Vol. 1 - Design Specifications, March, 1968.

6. Elliot, A. L., "The San Fernando Earthquake: A Lesson in Highway and Bridge Design", Civil Engineering, ASCE, Special Issue, September, 1972.

7. Scordelis, A. C., "Analysis of Simply Supported Box Girder Bridges", SESM 66-17, Structural Engineering and Structural Mechanics Division (SESM), University of California, Berkeley, October, 1966.

8. Scordelis, A. C., "Analysis of Continuous Box Girder Bridges", SESM 67-15, SESM, University of California, Berkeley, November, 1967.

9. Bouwkamp, J. G., Scordelis, A. C., Wasti, S. T., "Structural Behavior of a Two Span Reinforced Concrete Box Girder Bridge Model", SESM 71-5, SESM, University of California, Berkeley, April, 1971.

10. Godden, W. G., Aslam, M., "Model Studies of Skew Box-Girder Bridges", SESM 71-26, SESM, University of California, Berkeley, December, 1971.

11. Przemieniecki, J. S., "THEORY OF MATRIX STRUCTURAL ANALYSIS", McGraw-Hill, 1968.

12. Morris, D. L., "Curved Beam Stiffness Coefficients", Journal of the Structural Division, ASCE, Vol. 94, No. ST5, May, 1968.

13. Tezcan, S. S., Ovunc, B., "Analysis of Plane and Space Frameworks with Curved Members", International Association for Bridge and Structural Engineering Memoires, Vol. 25, 1965.

14. Wilson, E. L., "Earthquake Analysis of Reactor Structures", Symposium on Seismic Analysis of Pressure Vessel and Piping Components, San Francisco, Edited by D. H. Pai, ASME, 1971.

15. Penzien, J., "Soil-Pile Foundation Interaction", EARTHQUAKE ENGINEERING, R. L. Wiegel, Coordinating Editor, Prentice-Hall, 1970.

16. Minami, T., "Elasto-Plastic Earthquake Response of Soil-Building Systems", EERC 72-3, EERC, University of California, Berkeley, August, 1972.

17. Richart, E. F., Hall, J. R., and Woods, R. D., "VIBRATIONS OF SOILS AND FOUNDATIONS", Prentice-Hall, 1970.

18. Clough, R. W., "Analysis of Structural Vibrations and Dynamic Response", Japan - U. S. Seminar on Matrix Method of Structural Analysis and Design, August 25-30, 1969, Tokyo, Japan, 1969.

19. Wilson, E. L., Penzien, J., "Evaluation of Orthogonal Damping Matrices", International Journal for Numerical Methods in Engineering, Vol. 4, pp 5-10, January, 1972.

20. Blume, J. A., Newmark, N. M., and Corning, L. H., "DESIGN OF MULTIPLE-STORY REINFORCED CONCRETE BUILDING FOR EARTHQUAKE MOTIONS", Portland Cement Association, 1961.

21. Mattock, A. H., Kriz, L. B., and Hognestad, E., "Rectangular Concrete Stress Distribution in Ultimate Strength Design", ACI, Vol. 32, No. 8, February, 1961.

22. Bresler, B., "Design Criteria for Reinforced Columns Under Axial Load and Biaxial Bending", ACI, Vol. 32, No. 5, November, 1960.

23. Pannell, F. N., "Failure Surfaces for Members in Compression and Biaxial Bending", ACI, Vol. 60, No. 1, January, 1963.

24. Hill, R., "THE MATHEMATICAL THEORY OF PLASTICITY", Oxford, Clarendon Press, 1950.

25. Chi, H. M., and Powell, G. H., "Computational Procedure for Inelastic Finite Element Analysis", SESM 73-2, SESM, University of California, Berkeley, January, 1973.

26. Bathe, K. J., "Solution Methods for Large Generalized Eigenvalue Problems in Structural Engineering", SESM 71-20, SESM, University of California, Berkeley, November, 1971.

27. Clough, R. W., "Earthquake Response of Structures", EARTHQUAKE ENGINEERING, R. L. Wiegel, Coordinating Editor, Prentice Hall, 1970.

28. Newmark, N. M., "A Method of Computation for Structural Dynamics", Proceeding ASCE, Vol. 85, No. EM3, 1959.

29. Nickell, R. E., "On the Stability of Approximate Operators in Problems of Structural Dynamics", International Journal of Solids and Structures, Vol. 7, 1971.

30. Bathe, K. J., Wilson, E. L., "Stability and Accuracy Analysis of Direct Integration Methods", International Journal of Earthquake Engineering and Structural Dynamics, Vol. 1, 1973.

31. Wilson, E. L., "SAP - A General Structural Analysis Program", SESM 70-20, SESM, University of California, Berkeley, September, 1970.

32. "Report on Evaluation and Modification of Earthquake Resistance Design Criteria", California State Division of Highways, March 1971.

33. Chopra, A. K., Bertero, V. V., and Mahin, S., "Response of the Olive View Medical Center Main Building During the San Fernando Earthquake", Proceedings, Fifth World Conference on Earthquake Engineering, Rome, June, 1973.

APPENDIX I

FORTRAN IV LISTING OF PROGRAM YIELD

```
      PROGRAM YIELD (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT)
C
C
C     PROGRAM TO COMPUTE THE ULTIMATE AXIAL FORCE AND BIAXIAL BENDING
C     MOMENT FOR REINFORCED CONCRETE COLUMN OF ARBITRARY CROSS-SECTION
C
C
      COMMON/INPAR/ FC,E,EO,SY,TOLP,TOLE,NITRN
      DIMENSION T(3),HEAD(12)
C
      COMMON A(3000)
      MTOT=2000
C
   10 CALL SECOND(T(1))
C
C     INPUT PROGRAM CONTROL DATA AND MATERIAL PROPERTIES
C
      READ (5,1000) HEAD,NLOOP,NNODE,NBAR,NAS,NP,NE
      IF (NLOOP.EQ.0) STOP
      WRITE(6,2000) HEAD,NLOOP,NNODE,NBAR,NAS,NP,NE
C
      NITRN=30
      TOLP=1.0E-06
      TOLE=1.0E-04
C
      READ (5,1010) FC,E,EO,SY
      WRITE(6,2010) FC,E,EO,SY
C
C     COMPUTE ULTIMATE LOAD CAPACITIES
C
      N1=1
      N2=N1+NLOOP
      N3=N2+NP
      N4=N3+NE
      N5=N4+NE
      N6=N5+NAS
C
      NPE=NP*NE
      N61=N6+NPE
      N62=N61+NPE
      N63=N62+NPE
      N64=N63+NPE
      N65=N64+NPE
      N66=N65+NPE
      N67=N66+NPE
      N68=N67+NPE
C
      MNODE=NNODE+NLOOP
      N7=N68+MNODE
      N8=N7+MNODE
      N9=N8+MNODE
      N9=N8+MNODE
      N10=N9+MNODE
      N11=N10+MNODE
      N12=N11+MNODE
C
      N13=N12+NBAR
      N14=N13+NBAR
      N15=N14+NBAR
      N16=N15+NBAR
      N17=N16+NBAR*3
C
      NLOOP3=NLOOP*3
      N18=N17+NLOOP3
      N19=N18+NLOOP3
      N20=N19+NLOOP3
      N21=N20+NLOOP3
C
      N22=N21+NLOOP3
      N23=N22+NLOOP3
      N24=N23+NLOOP3
      N25=N24+NLOOP3
      N26=N25+NLOOP3
      N27=N26+NLOOP3
      N28=N27+NLOOP3
      N29=N28+NLOOP3
      N30=N29+NLOOP3
      IF (N30.LE.MTOT) GO TO 100
      WRITE(6,2020) N30
      STOP
C
  100 CALL ULOAD (A(N1),A(N2),A(N3),A(N4),A(N5),A(N6),A(N61),
     1               A(N62),A(N63),A(N64),A(N65),A(N66),A(N67),
     2               A(N68),A(N7),A(N8),A(N9),A(N10),A(N11),A(N12),
     3               A(N13),A(N14),A(N15),A(N16),A(N17),A(N18),A(N19),
     4               A(N20),A(N21),A(N22),A(N23),A(N24),A(N25),A(N26),
     5               A(N27),A(N28),A(N29),NLOOP,NP,NE,NAS,MNODE,NNODE,
     6               NBAR )
C
      CALL SECOND (T(2))
      T(3)=T(2)-T(1)
      WRITE(6,2030) T(3)
      GO TO 10
C
 1000 FORMAT (12A6/6I5)
 1010 FORMAT (4F10.0)
C
 2000 FORMAT (1H1,12A6////
     *  28H NUMBER OF BOUNDARY LOOPS  =  ,I5//
     *  28H NUMBER OF TOTAL NODES     =  ,I5//
     *  28H NUMBER OF STEEL BARS      =  ,I5//
     *  28H NUMBER OF STEEL AREAS     =  ,I5//
     *  28H NUMBER OF AXIAL LOAD STEPS=  ,I5//
     *  28H NUMBER OF MOMENT RATIOS   =  ,I5///)
 2010 FORMAT (1H0,26HMATERIAL PROPERTIES OF THE SECTION..///
     *  39H ULTIMATE CONCRETE COMPRESSIVE STRESS =  ,F14.4//
     *  39H YOUNG*S MODULUS OF STEEL BARS        =  ,F14.4//
     *  39H ULTIMATE CONCRETE COMPRESSIVE STRAIN =  ,F14.4//
     *  39H YIELDING STRESS OF STEEL BARS        =  ,F14.4)
 2020 FORMAT (1H1,23HSTORAGE EXCEEDED---NEED  ,I8/
     *  21H EXECUTION TERMINATED  )
 2030 FORMAT (////23H TOTAL EXECUTION TIME =  ,F8.2)
      END
```

```
      SUBROUTINE ULOAD (LNODE,RP,RE,DI,AS,P,UMX,UMY,UM ,SETA,DIST,TOP,
     *                 TOE,X,Y,ID,AC,ACX,ACY,XS,YS,IAS,IDS,ES,XA,YA,XB,YB,
     *                    ACA,ACAX,ACAY,ACB,ACBX,ACBY,ACAB,ACABX,ACABY,
     *                    NLOOP,NP,NE,NAS,MNODE,NNODE,NBAR)
C
C     COMPUTE ULTIMATE LOAD CAPACITIES FOR SPECIFIED LOAD COMBINATIONS
C
      COMMON/INFAP/FC,E,EO,SY,TOLP,TOLE,NITRN
C
      DIMENSION LNODE(NLOOP),RP(NP),RE(NE),DI(NE),AS(NAS),P(NP,NE),
     *  UMX(NP,NE),UMY(NP,NE),UM (NP,NE),SETA(NP,NE),DIST(NP,NE),
     *  TOP(NP,NE),TOE(NP,NE),
     *  X(MNODE),Y(MNODE),ID(MNODE),AC(MNODE),ACX(MNODE),ACY(MNODE),
     *  XS(NBAR),YS(NBAR),IAS(NBAR),IDS(NBAR),ES(NBAR,3),
     *  XA(NLOOP,3),YA(NLOOP,3),XB(NLOOP,3),YB(NLOOP,3),
     *  ACA(NLOOP,3),ACAX(NLOOP,3),ACAY(NLOOP,3),ACB(NLOOP,3),
     *  ACBX(NLOOP,3),ACBY(NLOOP,3),ACAB(NLOOP,3),ACABX(NLOOP,3),
     *  ACABY(NLOOP,3)
C
      COMMON/VALLE/SRA,CRA,D,DP
      DIMENSION F(3),G(3),GX(3),GY(3),T(3)
C
C     INPUT CROSS-SECTIONAL DATA
C
      WRITE(6,2000)
      READ (5,1010) (LNODE(L),L=1,NLOOP)
      WRITE(6,2010) (L,LNODE(L),L=1,NLOOP)
C
      WRITE(6,2015)
      NI=1
      DO 100 L=1,NLOOP
      NF=NI+LNODE(L)-1
      WRITE(6,2016) L,LNODE(L)
      READ (5,1020) (X(I),Y(I),I=NI,NF)
      WRITE(6,2020) (I,X(I),Y(I),I=NI,NF)
      NF=NF+1
      X(NF)=X(NI)
      Y(NF)=Y(NI)
      NI=NF+1
  100 CONTINUE
C
      READ (5,1030) (AS(I),I=1,NAS)
      READ (5,1040) (XS(J),YS(J),IAS(J),J=1,NBAR)
      DO 150 J=2,NBAR
      J1=J-1
  150 IF (IAS(J).EQ.0) IAS(J)=IAS(J1)
C
      WRITE(6,2025)
      WRITE(6,2030) (I,AS(I),I=1,NAS)
      WRITE(6,2035)
      WRITE(6,2040) (J,XS(J),YS(J),IAS(J),J=1,NBAR)
C
C     INPUT LOAD POINTS TO BE CALCULATED
C
      READ (5,1050) (RP(I),I=1,NP)
      READ (5,1060) (RE(I),DI(I),I=1,NE)
      READ (5,1070) IPO,IMXO,IMYO
C
      L=8*NP*NE
      DO 155 I=1,L
  155 P(I)=0.
C
      ACT=0.
      NI=1
      DO 200 L=1,NLOOP
      NF=NI+LNODE(L)-1
      DO 160 I=NI,NF
      I1=I+1
      CALL AREA (X(I),Y(I),X(I1),Y(I1),AC(I),ACX(I),ACY(I))
  160 ACT=ACT+AC(I)
      NF=NF+1
      AC(NF)=0.
      ACX(NF)=0.
      ACY(NF)=0.
      NI=NF+1
  200 CONTINUE
C
      AST=0.
      DO 250 J=1,NBAR
      JS=IAS(J)
  250 AST=AST+AS(JS)
C
      POT=-SY*AST
      FCP=0.85*FC
      POC=FCP*ACT-POT
      RATIO=AST/ACT
      FRATIO=POT/POC
      WRITE(6,2050) ACT,POC,POT,AST,RATIO,FRATIO
      RATIO =FRATIO+0.1
C
C     ITERATION FOR COMPUTING LOAD COMBINATIONS
C
      DO 900 IE=1,NE
      RA=RE(IE)
      RA=RA*3.14159265359/180.
      REI=RA
      D=DI(IE)
      DO 800 IP=1,NP
C
      KKK=0
      IF (RP(IP).LT.RATIO) RP(IP)=RATIO
      IF (RP(IP).GT.0.9) RP(IP)=0.9
      RPUO=RP(IP)
      IF (RE(IE)-45.0) 297,297,298
  297 IND=-1
      EXEYO=TAN (REI)
      GO TO 299
C
  298 IND= 1
      EXEYO=TAN (3.14159265359/2.0-REI)
C
  299 ITRN=0
C
  300 ITRN=ITRN+1
      CRA=COS(RA)
      SRA=SIN(RA)
C
C     ID ARRAYS FOR NODES AND STEEL BARS
C
      YPMIN= 10000.
      YPMAX=-10000.
      NI=1
      DO 340 L=1,NLOOP
```

```
      NF=NI+LNODE(L)-1
      DO 330 I=NI,NF
      ID(I)=0
      YP=X(I)*SRA+Y(I)*CRA-D
      IF (YP.GT.0.) ID(I)=1
      IF (YP.LE.YPMAX) GO TO 310
      YPMAX=YP
      IMAX=I
  310 IF (YP.GE.YPMIN) GO TO 330
      YPMIN=YP
      IMIN=I
  330 CONTINUE
      NF=NF+1
      ID(NF)=ID(NI)
      NI=NF+1
  340 CONTINUE
      DMAX=YPMAX-YPMIN
      IF ((YPMAX.LE.0.).OR.(YPMIN.GE.0.)) GO TO 655
C
      YIELDY=ABS(SY*YPMAX/(EO*E))
      DO 350 J=1,NBAR
      IOS(J)=0
      ES(J,2)=0.
      ES(J,3)=0.
      XP=XS(J)*CRA-YS(J)*SRA
      YP=XS(J)*SRA+YS(J)*CRA-D
      IF (ABS(YP).GE.YIELDY) GO TO 345
      ES(J,2)=-EO*(YPMAX-YP)/(YPMAX*YPMAX)
      ES(J,3)= EO*XP/YPMAX
  345 IF (YP.GE.YIELDY) IOS(J)=1
      IF (YP.LE.-YIELDY) IOS(J)=-1
      ES(J,1)= EO*YP/YPMAX
  350 CONTINUE
C
C     LOCATE POSITION OF NEUTRAL AXIS
C
      NLOOP39=NLOOP*39
      DP=0.15*YPMAX
      DO 355 I=1,NLOOP39
  355 XA(I)=0.
C
      NI=1
      DO 390 L=1,NLOOP
      NF=NI+LNODE(L)-1
      DO 380 I=NI,NF
      I1=I+1
      IF (ID(I)-ID(I1)) 360,380,370
C
  360 CALL NEUTRAL (XA(L,1),YA(L,1),XA(L,2),YA(L,2),XA(L,3),YA(L,3),
     *                  X(I),Y(I),X(I1),Y(I1))
C
      CALL AREA (XA(L,1),YA(L,1),X(I1),Y(I1),ACA(L,1),ACAX(L,1),
     *            ACAY(L,1))
C
      CALL DIFFRNT (ACA(L,2),ACAX(L,2),ACAY(L,2),
     *                  ACA(L,3),ACAX(L,3),ACAY(L,3),
     *                  XA(L,1),YA(L,1),XA(L,2),YA(L,2),XA(L,3),YA(L,3),
     *                  X(I1),Y(I1),0.0,0.0,0.0,0.0)
      GO TO 380
C
  370 CALL NEUTRAL (XB(L,1),YB(L,1),XB(L,2),YB(L,2),XB(L,3),YB(L,3),
     *                  X(I),Y(I),X(I1),Y(I1))
C
      CALL AREA (X(I),Y(I),XB(L,1),YB(L,1),ACB(L,1),ACBX(L,1),ACBY(L,1))
C
      CALL DIFFRNT (ACB(L,2),ACBX(L,2),ACBY(L,2),
     *                  ACB(L,3),ACBX(L,3),ACBY(L,3),
     *                  X(I ),Y(I ),0.0,0.0,0.0,0.0,
     *                  XB(L,1),YB(L,1),XB(L,2),YB(L,2),XB(L,3),YB(L,3))
  380 CONTINUE
      NF=NF+1
      NI=NF+1
C
      CALL AREA (XB(L,1),YB(L,1),XA(L,1),YA(L,1),
     *            ACAB(L,1),ACABX(L,1),ACABY(L,1))
C
      CALL DIFFRNT (ACAB(L,2),ACABX(L,2),ACABY(L,2),
     *                  ACAB(L,3),ACABX(L,3),ACABY(L,3),
     *                  XB(L,1),YB(L,1),XB(L,2),YB(L,2),XB(L,3),YB(L,3),
     *                  XA(L,1),YA(L,1),XA(L,2),YA(L,2),XA(L,3),YA(L,3))
C
  390 CONTINUE
C
C     COMPUTE ULTIMATE LOADS
C
      PU=0.0
      PUX=0.0
      PUY=0.0
      NI=1
      DO 450 L=1,NLOOP
      NF=NI+LNODE(L)-1
      DO 440 I=NI,NF
      I1=I+1
      IF ((ID(I).LT.1).OR.(ID(I1).LT.1)) GO TO 440
      PU=PU+AC(I)
      PLX=PLX+ACY(I)
      PUY=PLY+ACX(I)
  440 CONTINUE
      NF=NF+1
      NI=NF+1
      PU=PU+ACA(L,1)+ACB(L,1)+ACAB(L,1)
      PUX=PLX+ACAY(L,1)+ACBY(L,1)+ACABY(L,1)
      PUY=PLY+ACAX(L,1)+ACBX(L,1)+ACABX(L,1)
  450 CONTINUE
      PU=FCP*PU
      PUX=FCP*PLX
      PUY=FCP*PUY
C
      DO 500 J=1,NBAR
      JS=IAS(J)
      ASJ=AS(JS)
      IF (IOS(J)) 460,470,480
  460 FSJ=-SY*ASJ
      GO TO 490
  470 FSJ=E*ES(J,1)*ASJ
      GO TO 490
  480 FSJ=SY*ASJ
  490 PU=PU+FSJ
      PLX=PLX+FSJ*YS(J)
      PUY=PLY+FSJ*XS(J)
  500 CONTINUE
C
C     CHECK CONVERGENCE
C
```

```
      RPU=PL/POC
  520 RORE=0.
      IF (IND) 525,525,527
  525 EXEY=PUY/PUX
      THETA=ATAN(EXEY)*180./3.14159265359
      GO TO 532
C
  527 EXEY=PUX/PUY
      THETA=ATAN (EXEY) *180./3.14159265359
      THETA=90.-THETA
  532 RORE=ABS (EXEY-EXEYO)
      RDRP=ABS (RPL-RPUO)
      ANGLE=RA*180./3.14159265359
C
      IF ((RORE.LE.TOLE).AND.(RDRP.LE.TOLP)) GO TO 700
C
C     COMPUTE NEW LOCATION OF NEUTRAL AXIS
C
      DO 550 I=1,3
      F(I)=0.
      G(I)=0.
      GX(I)=0.
      GY(I)=0.
  550 CONTINUE
      DO 600 L=1,NLCOP
      DO 600 I=2,3
      F(I)=F(I)+ACA(L,I)+ACB(L,I)+ACAB(L,I)
      GX(I)=GX(I)+ACAY(L,I)+ACBY(L,I)+ACABY(L,I)
      GY(I)=GY(I)+ACAX(L,I)+ACBX(L,I)+ACABX(L,I)
  600 CONTINUE
      DO 610 I=2,3
      F(I)=FCP*F(I)
      GX(I)=FCP*GX(I)
      GY(I)=FCP*GY(I)
  610 CONTINUE
C
      DO 620 J=1,NBAR
      IF (IC(J).NE.0) GO TO 620
      JS=IAS(J)
      ASJ=AS(JS)
      FSDJ=E*ES(L,2)*ASJ
      FSRAJ=E*ES(J,3)*ASJ
      F(2)=F(2)+FSDJ
      F(3)=F(3)+FSRAJ
      GX(2)=GX(2)+FSDJ*YS(J)
      GY(2)=GY(2)+FSDJ*XS(J)
      GX(3)=GX(3)+FSRAJ*YS(J)
      GY(3)=GY(3)+FSRAJ*XS(J)
  620 CONTINUE
      F(2)=F(2)/POC
      F(3)=F(3)/POC
C
  630 DRA=0.
      DRP=RP(IP)-RPU
      AREI=ABS(RE(IE))
      IF ((AREI.LT.1.0E-10).OR.(AREI.GE.90.)) GO TO 650
      DRE=EXEYO-EXEY
      IF (IND) 635,635,640
  635 G(2)=(GY(2)*PLX-GX(2)*PUY)/(PUX*PUX)
      G(3)=(GY(3)*PLX-GX(3)*PUY)/(PUX*PUX)
      GO TO 645
  640 G(2)=(GX(2)*PUY-GY(2)*PUX)/(PUY*PUY)
      G(3)=(GX(3)*PUY-GY(3)*PUX)/(PUY*PUY)
C
  645 FG=F(2)*G(3)-F(3)*G(2)
      IF (ABS(FG).LT.1.0E-10) GO TO 650
      DD=(DRP*G(3)-DRE*F(3))/FG
      DRA=(-DRP*G(2)+DRE*F(2))/FG
      DD=DD/2.0
      DRA=DRA/2.0
      GO TO 690
  650 IF (ABS(F(2)).GT.1.0E-15) GO TO 660
  655 KKK=1
      GO TO 700
  660 DD=DRP/F(2)
      DD=DD/2.0
  690 DMAY=DMAX/4.0
      IF (DD.GE.DMAY) DD=DMAX/5.
      IF (DD.LE.-DMAY) DD=-DMAX/5.
      D=D+DD
      ADRA=ABS(DRA)
      IF (ADRA.GE.0.1) DRA=DRA*0.1/ADRA
      RA=RA+DRA
C
  699 IF (ITRN.LT.NITRN) GO TO 300
C
C     OUTPUT RESULTS
C
  700 CONTINUE
      UM(IP,IE)=SQRT(PUX*PUX+PUY*PUY)
      SETA(IP,IE)=ANGLE
      DIST(IP,IE)=D
      TOP(IP,IE)=RDRP
      TOE(IP,IE)=RORE
      P(IP,IE)=PU
      UMX(IP,IE)=PLX
      UMY(IP,IE)=PLY
      IF (KKK.LT.1) GO TO 800
      D=0.
      RA=REI
  800 CONTINUE
  900 CONTINUE
C
      WRITE(6,2300)
      DO 950 J=1,NE
      DO 940 I=1,NP
      IF (I.EQ.1) GO TO 930
      WRITE(6,2400) RP(I)   ,P(I,J),UMX(I,J),UMY(I,J),UM (I,J),SETA(I,J)
     *                DIST(I,J),TOP(I,J),TOE(I,J)
      GO TO 940
  930 WRITE(6,2500) RE(J)   ,RP(I)   ,P(I,J),UMX(I,J),UMY(I,J),
     *                UM (I,J),SETA(I,J),DIST(I,J),TOP(I,J),TOE(I,J)
  940 CONTINUE
  950 CONTINUE
C
      I=IPO*IMXO*IMYO
      IF (I.EQ.0) GO TO 1000
      WRITE(6,2600)
      DO 990 J=1,NE
      UMO=UM(IPO,J)
      DO 980 I=1,NP
      SETA(I,J)=UMX(I,J)/UM(I,IMXO)
      DIST(I,J)=UMY(I,J)/UM(I,IMYO)
      TOP(I,J)=LP(I,J)/UMO
```

```
      IF (I.EQ.1) GC TO 970
      WRITE(6,2700)        RP(I),P(I,J),UMX(I,J),UMY(I,J),UM(I,J),
     *                   SETA(I,J),OIST(I,J),TOP(I,J)
      GO TO 980
  970 WRITE(6,2800) RE(J),RP(I),P(I,J),UMX(I,J),UMY(I,J),UM(I,J),
     *                   SETA(I,J),DIST(I,J),TOP(I,J)
  980 CONTINUE
  990 CONTINUE
 1000 CONTINUE
      RETUPN
C
 1010 FORMAT (4I5)
 1020 FORMAT (2F10.0)
 1030 FORMAT (8F10.C)
 1040 FORMAT (2F10.0,I5)
 1050 FORMAT (8F10.C)
 1060 FORMAT (2F10.C)
 1070 FORMAT (3I5)
 2000 FORMAT (1H1,3CHCOLUMN CROSSSECTIONAL OATA....    /)
 2010 FORMAT (5H LCCP,10X,9HNUMBER OF   /
     *           5H  NC.,10X,9H  NODES     /(I5,I16))
 2015 FORMAT ( ///24HCOORDINATES OF NOOES....    /)
 2016 FORMAT (10H LCOP NO. .I3,5H  HAS,I5,8H   NOOES  /)
 2020 FORMAT (5H NOCE,10X,1HX,16X,1HY//(I5,2F16.5))
 2025 FORMAT (1H1,31HNUMBER OF OIFFERENT STEEL AREAS  // 5H  NO.,10X,
     *                 4HAREA   /)
 2030 FORMAT (I5,F16.5)
 2035 FCRMAT ( ///3CH COORDINATES OF STEEL BARS....   //
     *                  5H  BAR,10X,1HX,15X,1HY,15X,8HAREA NO.   /)
 2040 FORMAT (I5,2F16.5,I14)
 2050 FCRMAT (1H1/ 20H REFERENCE OATA.....  ///
     *  27H TOTAL APEA OF THE SECTION=  ,F16.5//
     =  27H ULTIMATE COMPRESSIVE LOAD=  ,F16.5//
     *  27H ULTIMATE TENSILE LOAD    =  ,F16.5//
     *  27H TOTAL REINFORCEMENT AREA =  ,F16.5//
     *  27H PEINFOPCEMENT RATIO      =  ,F16.5//
     *  27H TENSILE LOAD RATIO       =  ,F16.5)
 2300 FORMAT (1H1,28HPRINT OF FINAL RESULTS......  /
     *  5X,2HRE, 8X.2HRP, 9X,2HPU,11X,3HMUX,10X,3HMUY,10X,3HUM ,10X,
     *  4HSETA, 9X,4HDIST,9X.3HTOP,10X,3HTOE  /)
 2400 FORMAT (1H ,1CX,F8.3,2X,8E13.4 )
 2500 FCRMAT (/1H ,F8.3,2X.F8.3, 2X,8E13.4 )
 2600 FCRMAT (1H1
     *  5X,2HPE, 8X,2HRP, 9X,2HPU,11X,3HMUX,10X,3HMUY,10X,3HUM ,10X,
     *  7HUMX/MXO,6X,7HUMY/MYO,6X,6HUM/UMO    /)
 2700 FORMAT (1H ,1CX,F8.3,2X,4E13.4,3F13.4)
 2800 FORMAT (/1H ,F8.3,2X,F8.3,2X,4E13.4,3F13.4)
      END


      SUBROUTINE NELTRAL (XA,YA,XAO,YAO,XARA,YARA,X,Y,XK,YK)
C
      CCMMCN/VALUE/SRA,CRA,D,OP
C
      XX=XK-X
      YY=YK-Y
      AXX=ABS(XX)
      AYY=ABS(YY)
      IF ((AXX.LT.1.0E-15).AND.(AYY.GT.1.0E-15)) GO TO 400
      IF ((AXX.GT.1.0E-15).AND.(AYY.LT.1.0E-15)) GO TO 500
      IF ((AXX.LT.1.CE-15).ANO.(AYY.LT.1.0E-15)) GO TO 600
C
      OB=XX*SRA+YY*CRA
      IF (ABS(OB).GT.1.0E-15) GO TO 300
      STOP
C
  300 XA=X*YK-XK*Y
      YA=(-XA*SRA+(C+OP)*YY)/OB
      XA=(XA*CRA+(D+OP)*XX)/OB
      XAD=0.85*XX/CB
      YAO=0.85*YY/CB
      XARA=(-(X*YK-XK*Y)*SRA-XA*(-XX*CRA+YY*SPA))/OB
      YARA=(-(X*YK-XK*Y)*CRA-YA*(-XX*CRA+YY*SPA))/CB
      RETURN
C
  400 XA=X
      YA=(O+DP-X*SPA)/CRA
      XAO=0.
      YAO=0.85/CRA
      XARA=C.
      YARA=-(X (O+CP)*SRA)/(CRA*CPA)
      RETURN
C
  500 XA=(U+DP-Y*CPA)/SRA
      YA=Y
      XAD=0.85/SRA
      YAO=0.
      XARA=(Y-(D+OP)*CRA)/(SRA*SRA)
      YARA=C.
      RETUPN
C
  600 XA=X
      YA=Y
      RETURN
      END


      SUBROUTINE AREA (X.Y,X1,Y1,A,AX,AY)
C
      XX=X-X1
      YY=Y+Y1
      A=XX*YY/2.C
      AX=XX*(X*(YY+Y)+X1*(YY+Y1))/6.0
      AY=XX*(YY*YY-Y*Y1)/6.0
      RETURN
      END


      SUBROLTINE DIFFRNT (AD,AXD,AYD.ARA,AXRA,AYRA,
     *                         XA,YA,XAD,YAO,XARA,YARA,
     *                         XB,YB,XBD,YBD,XBRA,YBRA)
C
      XX=XA-XB
      YY=YA+YB
      XXD=XAD-XBD
      YYD=YAO+YBD
      XXRA=XARA-XBRA
      YYRA=YARA+YBRA
```

```
ULOA 484  C
ULOA 485        AD=(XXD*YY+XX*YYD)/2.0
ULOA 486        ARA=(XXRA*YY+XX*YYRA)/2.0
ULOA 487        AXD=(XXD*(XA*(YA+YY)+XB*(YY+YB))+
ULOA 488       *      XX*(XAD*(YY+YA)+XBD*(YY+YB)+XA*(YYD+YAD)+XB*(YYD+YBD)))/6.0
ULOA 489        AXRA=(XXRA*(XA*(YA+YY)+XB*(YY+YB))+
ULOA 490       *  XX*(XARA*(YA+YY)+XBRA*(YY+YB)+XA*(YYRA+YARA)+XB*(YYRA+YBRA)))/6.
ULOA 491        AYD=(XXD*(YY*YY-YA*YB)+XX*(2.0*YY*YYD-YAD*YB-YA*YBD))/6.0
ULOA 492        AYRA=(XXRA*(YY*YY-YA*YB)+XX*(2.0*YY*YYRA-YARA*YB-YA*YBRA))/6.0
ULOA 493        RETURN
ULOA 494        END
```

APPENDIX II

FORTRAN IV LISTING OF PROGRAM BSAP

```
      PROGRAM BSAP (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE1,TAPE2
     .             ,TAPE3,TAPE4,TAPE7,TAPE8,TAPE9,TAPE10,TAPE11,TAPE12)
C
C
C     BSAP....A MODIFIED VERSION OF ...SAP... PROGRAM
C             FOR STATIC AND DYNAMIC LINEAR ELASTIC ANALYSIS
C             OF 3-DIMENSIONAL BRIDGE STRUCTURAL SYSTEMS.
C
C
      COMMON/JUNK/ HED(12),JUK(193)
      COMMON/ELPAR/ NPAR(14),NUMNP,MBAND,NELTYP,N1,N2,N3,N4,N5,MTOT,NEQ
      COMMON/EM/ UCCU(1058)
      COMMON/MISC/ NBLOCK,NEQB,LL,NF,LB,NSVV,NFN,NGM,NAT,N7,NOT
C
      DIMENSION T(7)
C
C     PROGRAM CAPACITY CONTROLLED BY THE FOLLOWING TWO STATEMENTS...
C
      COMMON A(20000)
      MTOT=20000
C
C     PROGRAM CONTROL DATA
C
    5 CALL SECOND (T(1))
      READ (5,100) HED,NUMNP,NELTYP,NDYN
      IF (NUMNP.EQ.0) STOP
      WRITE(6,200) HED,NUMNP,NELTYP,NDYN
C
C     ANALYSIS TYPE CONTROL DATA
C
      READ (5,101) LL,NF,NSVV,NFN,NGM,NAT,N7,NOT
      IF (NSVV.LT.10) NSVV=10
      IF (NDYN.EQ.0) WRITE(6,201) LL
      IF (NDYN.GT.0) WRITE(6,202) NF,NSVV
      IF (NDYN.EQ.2) WRITE(6,203) NFN,NGM,NAT,N7,NOT
C
C     INPUT JOINT DATA----ID ARRAY STORED ON TAPE 8
C
      N1=1
      N2=N1+6*NUMNP
      N3=N2+NUMNP
      N4=N3+NUMNP
      N5=N4+NUMNP
      IF (N5.GT.MTOT) CALL ERROR(N5-MTOT)
      CALL INPUTM (A(N1),A(N2),A(N3),A(N4),NUMNP,NEQ,NEQB,NDYN,NGM)
C
C     FORM ELEMENT STIFFNESSES----STIFF. ON TAPE 2, STRESS ON TAPE 1
C
      CALL SECOND(T(2))
      CALL ELSTIF  (NUMEL)
C
C     COMPUTE MAX. PERMISSIBLE NUMBER OF EQUATIONS IN A BLOCK
C
      CALL SECOND (T(3))
      CALL BLOCFC (NEQB,NL,NDYN)
      NBLOCK=(NEQ-1)/NEQB+1
      IF (NEQB.GT.NEQ) NEQB=NEQ
      WRITE(6,204) NEQ,MBAND,NEQB,NBLOCK
C
C     INPUT NODAL LOADS AND MASSES----STORED ON TAPE 3
C
      N3=N2+NEQB*LL
      N4=N3+6*LL
      CALL SECOND (T(4))
      CALL INL  (A(N1),A(N2),A(N3),A(N4),NUMNP,NEQB,LL,NF,NEQ)
C
C     FORM TOTAL STIFFNESS---ON TAPE 4, MASS MATRIX---ON TAPE 9
C
      NE2B=2*NEQB
      N2=N1+NEQB*MBAND
      N3=N2+NEQB*LL
      N4=N3+4*LL
      NN2=N1+NE2B*MBAND
      NN3=NN2+NE2B*LL
      NN4=NN3+4*LL
      CALL SECOND (T(5))
      CALL ADDSTF  (A(N1),A(NN2),A(NN3),A(NN4),NUMEL,NBLOCK,NE2B,LL,
     .                 MBAND,NL,NEQ)
C
      CALL SECOND (T(6))
C
C     CALL FOR APPROPRIATE SOLUTION TYPE
C
      IF (NDYN.EQ.0) CALL STATIC
      IF (NDYN.EQ.1) CALL MODES
      IF (NDYN.EQ.2) CALL HISTRY
      IF (NDYN.EQ.3) CALL RESPEC
C
      CALL SECOND (T(7))
C
      DO 40 I=1,6
   40 T(I)=T(I+1)-T(I)
      T(7)=T(1)+T(2)+T(3)+T(4)+T(5)+T(6)
      WRITE(6,205) T
C
      GO TO 5
C
  100 FORMAT (12A6/3I5)
  101 FORMAT (8I5)
  200 FORMAT (1H1,12A6///
     .  28H  NUMBER OF NODAL POINTS    =  ,I5//
     .  28H  NUMBER OF ELEMENT TYPES   =  ,I5//
     .  28H  ANALYSIS TYPE CODE NO.    =  ,I5//)
  201 FORMAT (28H  NUMBER OF LOAD CASES      =  ,I5//)
  202 FORMAT (28H  NUMBER OF FREQUENCIES     =  ,I5//
     .            28H  EIGEN-SUBSPACE PARAMETER =  ,I5//)
  203 FORMAT (28H  NUMBER OF TIME FUNCTIONS =  ,I5//
     .            28H  GROUND MOTION INPUT CODE =  ,I5//
     .            28H  NUMBER OF DELAYED TIMES  =  ,I5//
     .            28H  NUMBER OF TIME STEP DT   =  ,I5//
     .            28H  DT MULTIPLIER FOR OUTPUT =  ,I5//)
  204 FORMAT (34H1 TOTAL NUMBER OF EQUATIONS         =  ,I5,
     .           /34H  BANDWIDTH                         =  ,I5,
     .           /34H  NUMBER OF EQUATIONS IN A BLOCK =  ,I5,
     .           /34H  TOTAL NUMBER OF BLOCKS            =  ,I5)
  205 FORMAT (/12H OVERALL LOG  //
     .             33H  NODAL POINT INPUT..............  ,F8.2//
     .             33H  FORM ELEMENT STIFFNESSES.......  ,F8.2//
     .             33H  DETERMINE EQUATION BLOCK SIZE.  ,F8.2//
     .             33H  INPUT NODAL LOADS AND MASSES..  ,F8.2//
     .             33H  FORM TOTAL STIFFNESS..........  ,F8.2//
     .             33H  PARTICULAR SOLUTION (ABOVE)...  ,F8.2//
     .            /33H  TOTAL SOLUTION TIME...........  ,F8.2//)
```

```
      END
```

```
      SUBROUTINE ERROR(N)
      WRITE (6,2000) N
 2000 FORMAT (// 20H STORAGE EXCEEDED BY  I6)
      STOP
      END
```

```
      SUBROUTINE INPUTM(ID,X,Y,Z,NUMNP,NEQ,NEQG,NDYN,NGM)
C
C     NODAL POINT INPUT AND GENERATION.
C
      DIMENSION X(1),Y(1),Z(1),ID(NUMNP,6) ,NID(6)
      ND=6
      WRITE(6,200)
      WRITE (6,204)
      KO=1
      DO 70 I=1,6
   70 NID(I)=0
   11 READ (5,100) N,(ID(N,I),I=1,6),X(N),Y(N),Z(N),KN
      WRITE(6,203) N,(ID(N,I),I=1,6),X(N),Y(N),Z(N),KN
      DO 60 I=1,6
      IF (ID(N,I)) 61,62,63
   61 IF (ID(N,I).EQ.-2) GO TO 65
      NID(I)=-1
      ID(N,I)=1
      GO TO 60
   62 IF (NID(I).EQ.-1) ID(N,I)=1
      GO TO 60
   63 NID(I)=0
      GO TO 60
   65 ID(N,I)=-1
   60 CONTINUE
      IF(KO.EQ.1) GO TO 12
C
C     CHECK IF GENERATION NEEDED
C
      IF (KN) 10,10,20
   12 KO=0
   10 CONTINUE
      NUMINT=1
      GO TO 15
C
C     GENERATE NEW NODES
C
   20 NUMINT=(N-NI)/KN
      DX=(X(N)-X(NI))/NUMINT
      DY=(Y(N)-Y(NI))/NUMINT
      DZ=(Z(N)-Z(NI))/NUMINT
      DO 21 J=1,NUMINT
      NN=NI+J*KN
      X(NN)=X(NN-KN)+DX
      Y(NN)=Y(NN-KN)+DY
      Z(NN)=Z(NN-KN)+DZ
C
C     SET JOINT DOF CODE... SAME AS FIRST JOINT IN A SERIES
C
      DO 22 JJ=1,6
      IF(ID(NI,JJ)-1) 24,26,25
C
C     GENERATE NEW MASTER NODES
C
   25 ID(NN,JJ)=ID(NI,JJ)+J*KN
      GO TO 22
   26 ID(NN,JJ)=ID(NI,JJ)
      GO TO 22
   24 IF (ID(NI,JJ).NE.-2) GO TO 23
      ID(NN,JJ)=ID(NI,JJ)
      GO TO 22
```

```
INPU  62      23 ID(NN,JJ)=0
INPU  63      22 CONTINUE
INPU  64      21 CONTINUE
INPU  65      15 NI=N
INPU  66  C
INPU  67  C      CHECK FOR LAST NODAL POINT
INPU  68  C
INPU  69         IF(NUMNP-NI) 13,13,11
INPU  70      13 CONTINUE
INPU  71  C
INPU  72  C      PRINT ALL NOCAL DATA
INPU  73  C
INPU  74         WRITE (6,202)
INPU  75         WRITE (6,204)
INPU  76         DO 50 N=1,NUMNP
INPU  77      50 WRITE(6,203) N,(ID(N,I),I=1,6),X(N),Y(N),Z(N)
INPU  78  C
INPU  79  C       NUMBER UNKNOWNS AND SET MASTER NODE NEGATIVE.
INPU  80  C
INPU  81         NEQ=0
INPU  82         DO 40 I=1,NUMNP
INPU  83         DO 40 J=1,ND
INPU  84         IF (ID(I,J).LT.0) GO TO 40
INPU  85         IF(ID(I,J)-1) 37,38,39
INPU  86      37 NEQ=NEQ+1
INPU  87         ID(I,J)=NEQ
INPU  88         GO TO 40
INPU  89      38 ID(I,J)=0
INPU  90         GO TO 40
INPU  91      39 ID(I,J)=-ID(I,J)
INPU  92      40 CONTINUE
INPU  93  C
INPU  94         NEQG=NEQ
INPU  95         IF (NCYN.NE.2) GO TO 90
INPU  96         IF (NCM.NE.2) GO TO 90
INPU  97  C
INPU  98         DO 80 I=1,NUMNP
INPU  99         DO 80 J=1,ND
INPU 100         IF (ID(I,J).NE.-1)  GO TO 80
INPU 101         NEQG=NEQG+1
INPU 102         ID(I,J)=NEQG
INPU 103      80 CONTINUE
INPU 104  C
INPU 105         WRITE(6,205)
INPU 106         WRITE(6,204)
INPU 107         DO 89 N=1,NUMNP
INPU 108      89 WRITE(6,203) N,(ID(N,I),I=1,6),X(N),Y(N),Z(N)
INPU 109  C
INPU 110      90 CONTINUE
INPU 111         REWIND 8
INPU 112         WRITE (8) ID
INPU 113  C
INPU 114         RETURN
INPU 115     100 FORMAT (7I5,3F10.0,I5)
INPU 116     104 FORMAT (I5,4F10.4)
INPU 117  C
INPU 118     200 FORMAT (26H1NODAL POINT DATA AS INPUT//)
INPU 119     202 FORMAT (26H1COMPLETE NODAL POINT DATA  //  )
INPU 120     203 FORMAT (I5,6I5,3F13.3,I5)
INPU 121     204 FORMAT (5HONODE 6X 24HBOUNDARY CONDITION CODES 13X
INPU 122       .  23HNODAL POINT COORDINATES / 7H NUMBER 2X 1HX 4X 1HY 4X 1HZ 3X
INPU 123       . 2HXX 3X 2HYY 3X 2HZZ12X 1HX 12X 1HY 12X 1HZ/)
```

```
INPU 124     205 FORMAT (26H1INTERNAL NODAL POINT DATA  // )
INPU 125  C
INPU 126         END
```

```
      SUBROUTINE ELSTF(NUMEL)
C
      COMMON /ELPAR/ NPAR(14),NUMNP,MBAND,NELTYP,N1,N2,N3,N4,N5,MTOT,NEQ
C
      MBAND=0
      NUMEL=0
      REWIND 1
      REWIND 2
      DO 900 M=1,NELTYP
C
      READ  (5,1001) NPAR
      WRITE (1) NPAR
      NUMEL=NUMEL+NPAR(2)
      MTYPE=NPAR(1)
C
      GO TO (1,2,3,4,5)  MTYPE
C
C     THREE DIMENSIONAL TRUSS ELEMENTS
C
    1 CALL TRUSS
      GO TO 900
C
C     THREE DIMENSIONAL BEAM ELEMENTS
C
    2 CALL BEAM
      GO TO 900
C
C     THREE DIMENSIONAL CURVED BEAM ELEMENTS
C
    3 CALL CBEAM
      GO TO 900
C
C     BOUNDARY ELEMENTS
C
    4 CALL BOUND
      GO TO 900
C
C     EXPANSION JOINT ELEMENTS
C
    5 CALL EXPJO
C
  900 CONTINUE
      RETURN
 1001 FORMAT (14I5)
      END


      SUBROUTINE WRITET(MBAND,NDIF)
      COMMON/ELPAR/ NPAR(14),NUMNP,MBANK,NELTYP,N1,N2,N3,N4,N5,MTOT,NEQ
      COMMON /EM/LM(24),ND,NS,S(24,24),P(24,4),XM(24),ST(12,24),TT(12,4)
C
C     CALCULATION OF BAND WIDTH AND WRITES ELEMENT MATRICES ON TAPES
C
      MIN=100000
      MAX=0
      DO 450 L=1,ND
      IF (LM(L).EQ.0) GO TO 450
      IF (LM(L).GT.NEQ) GO TO 450
      IF (LM(L).GT.MAX) MAX=LM(L)
      IF (LM(L).LT.MIN) MIN=LM(L)
  450 CONTINUE
      NDIF=MAX-MIN+1
      IF (NDIF.GT.MBAND) MBAND=NDIF
C
      WRITE (1) NS,ND,LM,ST,TT
      WRITE (2) LM,ND,NS,S,P,XM
C
      RETURN
      END
```

```
      SUBROUTINE BLOCEQ (NEQG,NL,NOYN)
C
C     DETERMINE APPROPRIATE NEQB FOR EACH TYPE OF ANALYSIS
C
      COMMON/ELPAR/ NPAR(14),NUMNP,MBAND,NELTYP,N1,N2,N3,N4,N5,MTOT,NEQ
      COMMON/MISC/ NBLOCK,NEQB,LL,NF,LB,NSVV,NFN,NGM,NAT,NT,NOT
      DIMENSION N(3),NB(4)
C
      IF (NDYN.GT.0) GO TO 10
C
C     STATIC ANALYSIS
C
      IF (LL.GE.1) GO TO 5
      WRITE(6,2000)
      STOP
    5 NL=LL
      NEQB=(MTOT-4*LL)/(MBAND+LL+1)/2
      NSB=(MBAND+LL)*NEQB
      NSBB=NEQB*LL+(2*(MBAND-1)/NEQB)
      IF (NSBB.LT.NSB) NSBB=NSB
      NB(1)=(MTOT-NSBB)/(LL+MBAND+1)
      IF (NEQB.GT.NB(1)) NEQB=NB(1)
      RETURN
C
C     DYNAMIC ANALYSIS ---- MODES AND FREQUENCIES
C
   10 NL=0
      LL=1
      NEQB=(MTOT-4*LL)/(MBAND+LL+1)/2
      NBLOCK=(NEQ-1)/NEQB+1
      IF (NBLOCK.GT.1) GO TO 12
      N(1)=NEQ*MBAND+(NF+3)*(NEQ+5)+1
      IF (N(1).GT.MTOT) CALL ERROR (N(1)-MTOT)
      GO TO 30
   12 NV=2*NF
      IF (NF.GT.NSVV) NV=NF+NSVV
      NB(1)=MTOT/(2*MBAND+1)
      NB(2)=(MTOT-3*NV*(NV+1))/(2*NV+1)
      NB(3)=(MTOT-6*NV)/(MBAND+1)
      DO 15 I=1,3
   15 IF (NEQB.GT.NB(I)) NEQB=NB(I)
   20 NBLOCK=(NEQ-1)/NEQB+1
      NSB=(MBAND-2)/NEQB+1
      IF (NSB.GE.NBLOCK) NSB=NBLOCK-1
      NB(4)=(MTOT-2*NV-1)/(MBAND+NV+NV*(NSB+1)+1)
      IF (NEQB.LE.NB(4)) GO TO 30
      NEQB=NB(4)
      GO TO 20
   30 IF (NDYN.EQ.1) RETURN
C
C     DYNAMIC ANALYSIS ---- RESPONSE SPECTRUM
C
      IF (NDYN.EQ.2) GO TO 40
      IF(NDYN.EQ.3) LL=NF
      NB(1)=(MTOT-6*NF)/(NF+3)
      NB(2)=(MTOT-6*NUMNP-6*(NF+1))/(NF+2)
      DO 35 I=1,2
      IF (NEQB.GT. NB(I)) NEQB=NB(I)
   35 CONTINUE
      RETURN
C
C     DYNAMIC ANALYSIS ---- TIME HISTORY RESPONSE
C
   40 NOS=(NT-1)/NOT
      N(2)=NFN*(NF+1)+NAT+NT+1
      N(1)=NELTYP+NF*(NOS+8)+1
      MAX=(MTOT-N(2))/2
      IF (NT.GT.MAX) MAX=NT
      N(2)=N(2)+MAX
      N(3)=MTOT
      IF (N(1).GT.N(3)) N(3)=N(1)
      IF (N(2).GT.N(3)) N(3)=N(2)
      IF (N(3).GT.MTOT) CALL ERROR (N(3)-MTOT)
      NB(1)=(MTOT-NFN*NF*NAT-2*LL )/(2*NFN+NF+LL+1)
      NB(2)=(MTOT-6*NUMNP)/(NFN+1)
      NB(3)=(MTOT-NELTYP-8*NF)/(1+NBLOCK*NF)
      DO 45 I=1,3
      IF (NEQB.GT.NB(I)) NEQB=NB(I)
   45 CONTINUE
C
      NCN=NGM-1
      IF (NCN) 70, 60, 50
   50 IF (NEQG.GT.NEQ) GO TO 55
      WRITE(6,2001)
      STOP
   55 LL=NEQG-NEQ
      NB(1)=(MTOT-4*LL)/(MBAND+LL+1)/2
      NSB=(MBAND+LL)*NEQB
      NSBB=NEQB*LL+(2*(MBAND-1)/NEQB)
      IF (NSBB.LT.NSB) NSBB=NSB
      NB(2)=(MTOT-NSBB)/(LL+MBAND+1)
      NB(3)=(MTOT-2*LL-5*NUMNP)/(LL+2)
      NB(4)=(MTOT-NFN*NF*NAT-2*LL)/(2*NFN+NF+LL+1)
      DO 57 I=1,4
      IF (NEQB.GT.NB(I)) NEQB=NB(I)
   57 CONTINUE
      GO TO 70
C
   60 NB(1)=(MTOT-2*LL)/(NFN+1)/2
      IF (NEQB.GT.NB(1)) NEQB=NB(1)
C
   70 RETURN
C
 2000 FORMAT (////
     .  53H FOR STATIC ANALYSIS LL MAY NOT BE INPUT AS ZERO....
     .  22H EXECUTION TERMINATED. )
 2001 FORMAT (////
     .  60H JOINT DOF CODES ARE NOT SPECIFIED CORRECTLY FOR JOINTS      /
     .  60H HAVING MULTIPLE GROUND MOTION INPUT...EXECUTION TERMINATED.)
C
      END
```

```
      SUBROUTINE INL(ID,B,TR,TMASS,NUMNP,NEQB,LL,NF,NEQ)
C
C     INPUT NODAL LOADS AND MASSES
C
      DIMENSION ID(NUMNP,6),B(NEQB,LL),TR(6,LL),TMASS(NEQB)
      COMMON / JUNK / R(6),TXM(6)
C
      NT=3
      REWIND NT
      KSHF=0
      IF (NF.NE.0) GO TO 720
      WRITE (6,2002)
      GO TO 740
  720 WRITE (6,2003)
  740 DO 750 I=1,NEQB
      TMASS(I)=0.
      DO 750 K=1,LL
  750 B(I,K)=0.0
C
      DO 900 NN=1,NUMNP
C
      DO 100 I=1,6
      TXM(I)=0.
      DO 100 J=1,LL
  100 TR(I,J)=0.0
C
      IF(NN.EQ.1) GO TO 300
  150 IF(N.NE.NN) GO TO 400
      DO 200 I=1,6
      IF (L) 180,180,190
  180 TXM(I)=R(I)
      GO TO 200
  190 TR(I,L)=R(I)
  200 CONTINUE
  300 READ (5,1001) N,L,R
      IF (N.EQ.0) GO TO 150
      WRITE(6,2001) N,L,R
      GO TO 150
C
  400 DO 800 J=1,6
      II=ID(NN,J)
      IF (II.GT.NEQ) GO TO 800
      II=II-KSHF
      IF (II) 800,800,500
  500 DO 600 K=1,LL
  600 B(II,K)=TR(J,K)
      TMASS(II)=TXM(J)
  610 IF(II.NE.NEQB) GO TO 800
      WRITE (NT) B,TMASS
      KSHF=KSHF+NEQB
      DO 700 I=1,NEQB
      TMASS(I)=0.
      DO 700 K=1,LL
  700 B(I,K)=0.0
  800 CONTINUE
  900 CONTINUE
      WRITE (NT) B,TMASS
C
      RETURN
 1001 FORMAT (2I5,7F10.4)
 2001 FORMAT (2I5,7F12.3)
 2002 FORMAT (23H1.....NODAL POINT LOADS // 10H NODE LOAD 27X
     . 14HAPPLIED LOADS             / 10H  NO. CASE  8X  2HFX  10X
     .   2HFY 10X 2HFZ 10X 2HMX 10X 2HMY 10X 2HMZ )
 2003 FORMAT (23H1....NODAL POINT MASSES  // 10H NODE LOAD 27X
     . 19HCONCENTRATED MASSES /10H  NO. CASE  8X  2HMX  10X
     .   2HMY 10X 2HMZ 10X 3HI/X 9X 3HI/Y 9X 3HI/Z )
      END
```

```
      SUBROUTINE ADCSTF(A,B,STR,TMASS,NUMEL,NBLOCK,NE28,LL,MBAND,NL,NEQ)
C
C     FORMS GLOBAL EQUILIBRIUM EQUATIONS IN BLCCKS
C
      DIMENSION AINE28,MBAND),B(NE28,LL),STR(4,LL),TMASS(NE28),STIF(722)
      COMMCN/EM/LMI24),ND,NS,S(24,24),PI24,4),XM(24),ST(12,24),TT(12,4)
      COMMCN/MISC/ NBLOC,NEQBL,LLL,NF,LB,NSVV,NFN,NGM,NAT,NT,NOT
      EQUIVALENCE (STIF,LM)
      NECH=NE28/2
      K=NFQB+1
      X=NBLCCK
      MB=SQRT(X)
      MB=MB/2+1
      NEBB=MB*NE28
      MM=1
C
      NSHIFT=0
      REWINC 3
      REWINC 4
      REWINC 9
C
C     READ ELEMENT LOAD MULTIPLIERS
C
      DC 40 L=1,LL
      DC 40 I=1,4
   40 STR(I,L)=C.
C
      IF (NL.EC.0) GO TO 65
      WRITE (6,2000)
      DO 50 L=1,LL
      READ  (5,10C2)   ISTR(I,L),I=1,4)
   50 WRITE (6,2002) L,ISTR(I,L),I=1,4)
C
   65 CONTINUL
      WRITE (8) STR
C
C     FORM EQUATIONS IN BLOCKS      ( 2 BLOCKS AT A TIME)
C
      DO 1000 M=1,NBLOCK ,2
      DO 1CC I=1,NE28
      DO 100 J=1,MBAND
  100 A(I,J)=0.
      READ (3) I(B(I,L),I=1,NEQB),L=1,LL),(TMASS(I),I=1,NEQB)
      IF (M.EC.NBLCCK) GO TO 200
      READ (3) I(B(I,L),I=K,NE28),L=1,LL),(TMASS(I),I=K,NE28)
  200 CONTINUE
C
      REWINC 7
      REWINC 2
      NA=7
      NUML=NUM7
      IF (MM.NE.1) GO TO 75
      NA=2
      NUME=NUMEL
      NUM7 =0
C
   75 DO 7CC N=1,NUME
      READ (NA)  STIF
      DO 6OC I=1,NC
      IF (LM(I).GT.NEQ) GO TO 600
      LMN=1-LM(I)
      II=LM(I)-NSHIFT
      IF (II.LE.0.OR.II.GT.NE28) GO TO 600
      TMASS(II)=TMASS(II)+XM(I)
      IF (NL) 22C,220,210
  210 CONTINUE
      DO 3CC L=1,LL
      DO 300 J=1,4
  300 B(II,L)=B(II,L)+P(I,J)*STR(J,L)
  220 DO 5CC J=1,NC
      JJ=LM(J)
      IF (JJ.LE.NEQ) GO TO 350
      L=JJ-NEQ
      B(II,L)=B(II,L)+S(I,J)
      GO TO 500
  350 JJ=JJ+LMN
      IF(JJ) 5CC,5CC,400
  400 A(II,JJ)=A(II,JJ)+S(I,J)
  500 CONTINUE
  600 CONTINUE
C
C     DETERMINE IF STIFFNESS IS TO BE PLACED ON TAPE 7
C
      IF (MM.GT.1) GO TO 700
      DC 65C I=1,NC
      IF (LM(I).GT.NEQ) GO TO 650
      II=LM(I) -NSHIFT
      IF(II.GT.NE28.AND.II.LE.NEBB) GO TO 660
  650 CONTINUE
      GO TC 700
  660 WRITE (7) STIF
      NUM7=NUM7+1
C
  700 CONTINUE
      WRITE(4) I(A(I,J),I=1,NEQB),J=1,MBAND),((B(I,L),I=1,NEQB),L=1,LL)
      IF (NGM.EC.2)
     .WRITE(12)I(A(I,J),I=1,NEQB),J=1,MBAND),I(B(I,L),I=1,NEQB),L=1,LL)
      WRITE(9) I(TMASS(I),I=1,NEQB),(I(B(I,L),I=1,NEQB),L=1,LL)
      IF(M.EG.NBLCCK) GO TO 1000
      WRITE(4) I(A(I,J),I=K,NE28),J=1,MBAND),((B(I,L),I=K,NE28),L=1,LL)
      IF (NGM.EC.2)
     .WRITE(12)I(A(I,J),I=K,NE28),J=1,MBAND),((B(I,L),I=K,NE28),L=1,LL)
      WRITE(9) (TMASS(I),I=K,NE28),I(B(I,L),I=K,NE28),L=1,LL)
      IF (MP.EQ.MP) MM=0
      MM=MM+1
 1000 NSHIFT=NSHIFT+NE28
C
      RETURN
 1002 FORMAT I4F10.C)
 2000 FORMAT I1CHISTRUCTURE 12X 25HELEMENT LOAD MULTIPLIERS /
     . 10H LOAD CASE 9X 1HA 9X 1HB 9X 1HC 9X 1HD/)
 2002 FORMAT I16,7X,4F10.3)
      END
```

```
      SUBROUTINE STATIC
C
      COMMON / MISC / NBLOCK,NEQB,LL,NF,LB,NSVV
      COMMON /JUNK/ XXX(4),NDYN,JUK(200)
      DIMENSION T(4)
      COMMON /ELPAR/ NPAR(14),NUMNP,MBAND,NELTYP,N1,N2,N3,N4,N5,MTOT,NEQ
      COMMON A(1)
C
      CALL SECOND (T(1))
C
C     SOLVE FOR DISPLACEMENT UNKNOWNS
C              ( SOLN ON TAPE 2.  TAPES 3 AND 7 USED FOR TEMP STORAGE)
C
      NSB=(MBAND+LL)*NEQB
      NSBB=NEQB*LL*(2+(MBAND-1)/NEQB)
      IF(NSBB.LT.NSB) NSBB=NSB
      N4=N3+NSBB
      CALL USOL(A(N1),A(N3),A(N4),NEQB,MBAND,LL,NBLOCK,NSB,4,3,7,2,2)
C
C     PRINT DISPLACEMENT
C
      CALL SECOND (T(2))
C
      N2=N1+NUMNP*6
      N3=N2+6*LL
      CALL PRINTD(A(N1),A(N2),A(N3),NEQB,NUMNP,LL,NBLOCK,NEQ,2,0)
C
      CALL SECOND (T(3))
C
C     COMPUTE STRESSES
C
   30 N2=N1+4*LL
      N3=N2+NEQB*LL
      LB=(MTOT-N3)/(NEQ +12)
      NDYN=0
      CALL STRESS(A(N1),A(N2),A(N3),NEQB,LB,LL,NEQ,NBLOCK)
C
      CALL SECOND (T(3))
C
      T(1)=T(2)-T(1)
      T(2)=T(3)-T(2)
      T(3)=T(4)-T(3)
      WRITE (6,1000) T(1),T(2),T(3)
C
      RETURN
 1000 FORMAT (23H1....TIME LOG (SECONDS)  ///
     .            33H  EQUATION SOLVING............... ,F8.2//
     . 33H  PRINT DISPLACEMENTS............  , F8.2 //
     . 33H  COMPUTE STRESSES..............   , F8.2 //)
       END
```

```
      SUBROUTINE USOL (A,B,MAXB,NEQB,MB,LL,NBLOCK,NSB,NORG,NBKS,NT1,
     .                                                           NT2,NEST)
      DIMENSION A(NSB),B(NSB),MAXB(NEQB)
C
      NC=MB+LL
      NBR=(MB-1)/NEQB+1
      INC=NEQB-1
      NMB=NEQB*MB
      N2=NT2
      N1=NT1
      REWIND NORG
      REWIND NBKS
C
C     REDUCE EQUATIONS BLOCK-BY-BLOCK
C
      DO 900 N=1,NBLOCK
      IF (N.GT.1.AND.NBR.EQ.1) GO TO 110
      IF (NBR.LE.1) GO TO 105
      REWIND N1
      REWIND N2
  105 NI=N1
      IF(N.EQ.1) NI=NORG
      READ (NI) A
  110 DO 300 I=1,NEQB
      D=A(I)
      IF(D) 115,300,120
  115 M=NEQB*(N-1)+I
      WRITE (6,116) M,D
  116 FORMAT (33H0SET OF EQUATIONS MAY BE SINGULAR   /
     .  26H DIAGONAL TERM OF EQUATION   I8, 8H  EQUALS   1PE12.4)
C
  120 II=I
      DO 125 J=2,NC
      II=II+NEQB
  125 A(II)=A(II)/D
C
      DO 130 J=I,NMB,NEQB
      IF (A(J).NE.0.) MAXB(I)=J
  130 CONTINUE
C
      JL=I+1
      IF (JL.GT.NEQB) GO TO 300
      II=I
      DO 200 J=JL,NEQB
      II=II+NEQB
      IF(II.GT.NMB) GO TO 200
      C=A(II)
      IF (C.EQ.0.0) GO TO 200
      C=C*A(I)
C
      KK=J
      MAX=MAXB(I)
      DO 150 JJ=II,MAX,NEQB
      A(KK)=A(KK)-C*A(JJ)
  150 KK=KK+NEQB
C
      KK=J +NMB
      JJ=I +NMB
      DO 175 L=1,LL
      A(KK)=A(KK)-C*A(JJ)
      KK=KK+NEQB
```

```
      WRITE (N8KS) A,MAX8
C
C     SUBSTITUTE INTO REMAINING EQUATIONS
C
      OC 800 NN=1,NBR
      IF(N+NN.GT.NBLOCK) GO TO 800
      NI=N1
      IF(N.EQ.1) NI=NORG
      IF(NN.EQ.NBR) NI=NORG
      READ (NI) B
      IL=1+NN*NEQB*NEQ8
      DO 70C I=1,NEQB
      II=IL
      DO 69C K=1,NEQ8
      IF (II.GT.NMB) GU TO 690
      C=A(II)
      IF (C.EQ.0.0) GO TO 690
      C=C*A(K)
      MAX=MAXB(K)
C
      KK=I
      DO 64C JJ=II,MAX,NEQ8
      B(KK)=B(KK)-C*A(JJ)
  640 KK=KK+NEQB
      KK=I+NMB
      JJ=K+NMB
      DC 65C L=1,LL
      B(KK)=B(KK)-C*A(JJ)
      KK=KK+NEQB
  650 JJ=JJ+NEQB
  690 II=II-INC
  700 IL=IL+NEQB
C
      IF(NBR.NE.1) GO TO 750
      DO 74C I=1,NSB
  740 A(I)=B(I)
      GC TO 800
  750 WRITE (N2) B
  800 CONTINUE
      M=N1
      NI=N2
  900 N2=M
C
C     BACKSUBSTITUTION - RESULTS ON TAPE NRST
C
      LS=LL*NEQB
      NEB=NEQ8*(NBR+1)
      NUM=NBR*NEQ8
      MAX=NEB*LL
      DO 905 I=1,MAX
  905 B(I)=C.
      REWINO NRST
C
      OC 1000 N=1,NBLOCK
      BACKSPACE NBKS
      READ (N8KS) A,MAXB
      BACKSPACE N8KS
      DO 910 L=1,LL
      K=L*NEB
      DO 91C J=1,NLM
      I=K-NEQ8
      B(K)=B(I)
  910 K=K-1
C
      I=NMB
      DO 920 L=1,LL
      K=(L-1)*NEB
      DO 92C J=1,NEQB
      I=I+1
      K=K+1
  920 B(K)=A(I)
C
      DO 955 I=1,NEQ8
      J=NEQB+1-I
      MAX=MAX8(J)
      IF (A(J).EQ.0.) GO TO 955
      DO 95C L=1,LL
      KK=J+(L-1)*NEB
      JJ=KK+1
      IL=J+NEQB
      C=B(KK)
      DO 940 II=IL,MAX,NEQB
      C=C-A(II)*B(JJ)
  940 JJ=JJ+1
  950 B(KK)=C
  955 CONTINUE
C
      I=0
      DO 96C L=1,LL
      K=(L-1)*NEB
      DC 96C J=1,NEQB
      K=K+1
      I=I+1
  960 A(I)=B(K)
C
      WRITE (NRST) (A(I),I=1,LS)
 1000 CONTINUE
      RETURN
C
 2000 FORMAT (33HOSET OF EQUATIONS MAY BE SINGULAR    /
     .  26H DIAGONAL TERM OF EQUATION   I6, 6H  EQUALS   1PE12.4)
      END
```

```
      SUBROUTINE PRMAT(A,NR,NC,MAX,TITLE)
C
      DIMENSION A(MAX,1)
C
      PRINT 1002,TITLE
      DO 100 J=1,NC,8
      JH=J+7
      IF (JH-NC) 75,75,50
   50 JH=NC
   75 PRINT 1000,(N,N=J,JH)
      DO 100 I=1,NR
  100 PRINT 1001,I,(A(I,K),K=J,JH)
C
      RETURN
C
 1000 FORMAT (///8X,8I14)
 1001 FORMAT (I4,4X,8E14.7)
 1002 FORMAT (/////8H MATRIX ,A10)
C
      END



      SUBROUTINE PRINTD(ID,D,B,NEQB,NUMNP,LL,NBLOCK,NEQ,NT,NF)
C
      DIMENSION ID(NUMNP,6),B(NEQB,LL),D(6,LL)
C
      REWIND NT
      IF(NF.GT.0) READ (NT)
      REWIND 8
      READ (8) ID
      M=NEQ
      NN=NEQB*NBLOCK
      IF(NF.EQ.0) WRITE (6,2003)
      IF(NF.GT.0) WRITE (6,2005)
      N=NUMNP
C
      DO 500 KK=1,NUMNP
C
      III=6
      JJJ=6
      I=6
      DO 250 II=1,6
      DO 100 L=1,LL
  100 D(I,L)=0.
      IF(M.GT.NN) GO TO 150
      IF (M.EQ.0) GO TO 150
      READ (NT) B
      NN=NN-NEQB
  150 NID=ID(N,I)
      IF (NID.GT.NEQ) GO TO 250
      IF (NID.LT.0) III=III-1
      IF (NID.EQ.0) JJJ=JJJ-1
      IF (NID.LT.1) GO TO 250
      K=M-NN
      M=M-1
C
      DO 200 L=1,LL
  200 D(I,L)=B(K,L)
  250 I=I-1
C
      IF (III.EQ.0) GO TO 400
      IF (JJJ.EQ.0) GO TO 450
      WRITE (6,2004) N,(L,(D(I,L),I=1,6),L=1,LL)
      GO TO 500
C
  400 WRITE (6,2006) N
      GO TO 500
C
  450 WRITE (6,2007) N
C
  500 N=N-1
C
      RETURN
C
 2003 FORMAT (40H1.......NODE DISPLACEMENTS AND ROTATIONS//
     .  5H NODE  5H LOAD  11X  1HX  11X  1HY  11X  1HZ  9X  2HXX
     .  9X  2HYY  9X  2HZZ/5H   NO,5H   NO /)
 2004 FORMAT (1H0,I4,I5,1P3E12.3,3E11.2/(I10,3E12.3,3E11.2))
 2005 FORMAT (19H1.......MODE SHAPES  //
     .  5H0NODE  5H MODE  11X  1HX  11X  1HY  11X  1HZ  9X  2HXX
     .   9X  2HYY  9X  2HZZ/5H   NO,5H   NO /)
 2006 FORMAT (1H0,I4,5X,18H...SLAVE JOINT....  )
 2007 FORMAT (1H0,I4,5X,24H...FIXED JOINT (ALL 0.0) )
C
      END



      SUBROUTINE STRESS (STR,B,D,NEQB,LB,LL,NEQ,NBLOCK)
C
      DIMENSION D(NEQ,LB),B(NEQB,LL),STR(4,LL)
      COMMON/EM/ND,NS,LM(24),S(24,24),P(24,4),XM(24),SA(12,24),TT(12,4)
      COMMON /ELPAR/ NPAR(14),NUMNP,MBAND,NELTYP,N1,N2,N3,N4,N5,MTOT,NEQ
      COMMON / JUNK / MM,L,K,NTAG,NDYN,SIG(12),EXM3(188)
C
      READ (8) STF
      NT=(LL-1)/LB +1
      LH=0
      REWIND 3
C
      DO 1000 II=1,NT
C
      LT =LH+1
      LLT=1-LT
      LH=LT+LB- 1
      IF(LH.GT.LL) LH=LL
C
C     MOVE DISPLACEMENTS INTO CORE FOR LB LOAD CONDITIONS
C
      REWIND 2
      IF(NDYN.EQ.3) READ (2)
      NQ=NEQB*NBLOCK
      DO 200 NN=1,NBLOCK
      READ (2) B
      N=NEQB
      IF (NN.EQ.1) N=NEQ-NQ+NEQB
      NQ=NQ-NEQB
      DO 200 J=1,N
      I=NQ+J
      DO 200 L=LT,LH
```

```
      K=L+LLT
  200 D(I,K)=B(J,L)
      LK=LM-LT+1
C
C     CALCULATE STRESSES FOR ALL ELEMENTS FOR LB LOAD CONDITIONS
C
      REWIND 1
      DO 1000 M=1,NELTYP
C
      IF(NDYN.EQ.0) WRITE (6,2000)
      IF(NDYN.EQ.3) WRITE (6,2001)
C
      READ (1) NPAR
      IF(NDYN.EQ.3) WRITE (3) NPAR
      MTYPE=NPAR(1)
      NPAR(1)=0
C
C     CALCULATE STRESSES
C
  250 NUME=NPAR(2)
      NTAG=0
      DO 1000 MM=1,NUME
      READ (1) NS,NC,LM,SA,TT
      WRITE (6,2003)
      K=0
      DO 950 L=LT,LM
      K=K+1
C
      DO 300 N=1,NS
      SIG(N)=TT(N,1)*STR(1,L)+TT(N,2)*STR(2,L)+TT(N,3)*STR(3,L)+
     .  TT(N,4)*STR(4,L)
      DO 300 J=1,NC
      JJ=LM(J)
      IF (JJ.GT.NEQ) GO TO 300
      IF (JJ) 300,300,290
  290 SIG(N)=SIG(N)+SA(N,J)*D(JJ,K)
  300 CONTINUE
C
C     CALCULATE ADDITIONAL STRESSES AND PRINT RESULTS
C
      GO TO (1,2,3,4,5) MTYPE
C     THREE DIMENSIONAL TRUSS ELEMENTS
C
    1 CALL TRUSS
      GO TO 900
C
C     THREE DIMENSIONAL BEAM ELEMENTS
C
    2 CALL BEAM
      GO TO 900
C
C
C     THREE DIMENSIONAL CURVED BEAM ELEMENTS
C
    3 CALL CBEAM
      GO TO 900
C
C     BOUNDARY ELEMENTS
C
    4 CALL BOUND
      GO TO 900
C
C     EXPANSION JOINT ELEMENTS
C
    5 CALL EXPJB
C
  900 CONTINUE
      IF(NDYN.EQ.3) WRITE (3) NS,SIG
  950 CONTINUE
 1000 CONTINUE
C
      RETURN
 2000 FORMAT (25H1 STATIC STRESS PRINT OUT )
 2001 FORMAT (28H1 RESPONSE SPECTRUM ANALYSIS //
     . 38H  STRESS PRINT OUT IN INDIVIDUAL MODES  )
 2003 FORMAT (/)
      END
```

```
      SUBROUTINE  MODES
C
C
      COMMON / MISC / NBLOCK,NEQB,LL,NF,LB,NSVV
      COMMON /ELPAR/ NPAR(14),NUMNP,MBAND,NELTYP,N1,N2,N3,N4,N5,MTOT,NEQ
      DIMENSION T(3)
      COMMON A(1)
C
C     EVALUATION OF MODE SHAPES AND FREQUENCIES
C
      CALL SECOND (T(1))
C
      IF (NSVV.LT.10) NSVV=10
      CALL DYNAM (NEQ,MBAND,NBLOCK,NEQB,NF,NSVV,MTOT)
C
C     PRINT MODE SHAPES IN JOINT ORDER
C
C
      CALL SECOND (T(2))
C
      N2=1+6*NUMNP
      N3=N2+6*NF
      CALL PRINTD (A(1),A(N2),A(N3),NEQB,NUMNP,NF,NBLOCK,NEQ,7,NF)
C
      CALL SECOND (T(3))
      T(1)=T(2)-T(1)
      T(2)=T(3)-T(2)
      WRITE (6,3000) T(1),T(2)
C
      RETURN
C
 3000 FORMAT (23H1....TIME LOG (SECOND)  ///
     . 33H  MODE SHAPES AND FREQUENCIES...   ,F8.2//
     . 33H  PRINT MODE SHAPES..............  ,F8.2//)
C
      END
```

```
      SUBROUTINE DYNAM (NEQ,MBAND,NBLOCK,NEQB,NF,NSVV,MTOT)
C
C      COMPUTE SPECFIED NUMBER OF SMALLEST EIGENVALUES AND ASSOCIATED
C      VECTORS IN THE GENERALIZED EIGENVALUE PROBLEM....BY K.J. BATHE
C
       COMMON /TAPES/NSTIF,NRED,NL,NR,NT,NMASS
       COMMON A(1)
C
       NSTIF=4
       NMASS=9
       NRED=10
       NL=2
       NR=3
       NT=7
       WRITE (6,1000) NEQ,MBAND,NBLOCK,NEQB,NF
C
       IF (NBLOCK.GT.1) GO TO 300
       NIM=2
       NC=NF+NIM
       NWA=NEQ*MBAND
       N2=1+NWA
       N3=N2+NEQ
       N4=N3+NEQ
       N5=N4+NEQ
       N6=N5+NEQ
       N7=N6+NEQ*NC
       N8=N7+NEQ*NC
       N9=N8+NC
       N10=N9+NC
       N11=N10+NC
       N12=N11+NC
       N13=N12+NC
C
       IF (MTOT-N13) 100,200,200
  100  PH=7HSECANTC
       WRITE (6,1010) PH,N13
       STOP
C
  200  CALL SECANTC (A(1),A(N2),A(N3),A(N4),A(N5),A(N6),A(N7),A(N8),A(N9
     1),A(N10),A(N11),A(N12),NEQ,MBAND,NWA,NF,NC)
       DO 250 I=1,NF
       II=N8+I-1
  250  A(I)=A(II)
       GO TO 600
C
  300  NWA=NEQB*MBAND
       NV=2*NF
       IF (NF.GT.NSVV) NV=NF+NSVV
       NWV=NV*NEQB
       NTB=(MBAND-2)/NEQB+1
       IF (NTB.GE.NBLOCK) NTB=NBLOCK-1
       NWVV=NWV*(NTB+1)
       N=2*NWA+NEQB
       N2=NWV+NEQB+NV
       N3=NWA+NWV+NWVV+NEQB+2*NV
       N4=NV*(3*NV+2*NEQB+3)+NEQB
       N5=NWA+6*NV+NEQB
C
       IF (N2.GT.N) N=N2
       IF (N3.GT.N) N=N3
       IF (N4.GT.N) N=N4
       IF (N5.GT.N) N=N5
       IF (MTOT-N) 400,500,500
  400  PH=7HSSPACEP
       WRITE (6,1010) PH,N
       STOP
C
  500  CALL SSPACEP (NEQ,MBAND,NBLOCK,NEQB,NF,NV,NWA,NWV,NWVV,NTB)
C
  600  RETURN
C
 1000 FORMAT (1H1,30HFREQUENCY ANALYSIS INFORMATION  //
     1           /30H NO OF EQUATIONS ..............  I4
     2           /30H 1/2 BANDWIDTH OF A ...........  I4
     3           /30H NO OF BLOCKS .................  I4
     4           /30H NO OF EQNS PER BLOCK .........  I4
     5           /30H NO OF FREQUENCIES REQD .......  I4  )
 1010 FORMAT (30H0FOR EXECUTION OF SUBROUTINE  ,A7,30H MTOT NEED TO BE I
     .NCREASE TO  ,I6)
C
       END
```

```
SECA   1            SUBROUTINE SECANTD (A,B,V,MAXA,W,VV,WW,ROOT,TIM,ERRVL,ERRVR,
SECA   2          .                      NITE,N,MA,NWA,NRODT,NC)
SECA   3  C
SECA   4            COMMON /TAPES/NSTIF,NRED,NL,NR,NT,NMASS
SECA   5            DIMENSION A(NWA),B(N),V(N),VS(1),W(N),VV(N,NC),WW(N,NC),ROOT(NC),
SECA   6          .          TIM(NC),ERRVL(NC),ERRVR(NC)
SECA   7            INTEGER NITE(NC),MAXA(NC)
SECA   8  C
SECA   9  C         FOLLOWING TOLERANCES ARE SET FOR CDC 6400.....
SECA  10  C
SECA  11            ACTOL=1.0E-04
SECA  12            RCBTOL=1.0E-06
SECA  13            RTOL=1.0E-10
SECA  14            RQTOL=1.0E-12
SECA  15            SCALE=2.0**SCO
SECA  16  C
SECA  17            NTF=5
SECA  18            IITEM=10
SECA  19            NITEM=40
SECA  20  C
SECA  21            REWIND NMASS
SECA  22            READ (NMASS) B
SECA  23  C
SECA  24            ETA=2.0
SECA  25            NOV=0
SECA  26            JR=1
SECA  27            NSK=0
SECA  28            NWA=N*MA
SECA  29            ISC=1000
SECA  30  C
SECA  31            CALL SECOND (TIM1)
SECA  32            RA=0.0
SECA  33            RR=0.0
SECA  34            CALL BANDET (A,B,V,MAXA,N,NWA,RA,NSCH,DETA,ISC,1)
SECA  35            FA=DETA
SECA  36            FB=FA
SECA  37            DETB=DETA
SECA  38  C
SECA  39  C         FIND LOWER BOUND ON SMALLEST EIGENVALUE
SECA  40  C
SECA  41            WRITE (6,1010)
SECA  42            DO 100 I=1,N
SECA  43     100    W(I)=B(I)
SECA  44            RT=0.0
SECA  45            IITE=0
SECA  46            KK=3
SECA  47     110    IITE=IITE+1
SECA  48            DO 120 I=1,N
SECA  49     120    V(I)=W(I)
SECA  50            CALL BANDET (A,B,V,MAXA,N,NWA,RA,NSCH,DETA,ISC,KK)
SECA  51            KK=2
SECA  52            RQT=0.0
SECA  53            DO 130 I=1,N
SECA  54     130    RQT=RQT+W(I)*V(I)
SECA  55            DO 180 I=1,N
SECA  56     180    W(I)=B(I)*V(I)
SECA  57            RQB=0.0
SECA  58            DO 140 I=1,N
SECA  59     140    RQB=RQB+W(I)*V(I)
SECA  60            RQ=RQT/RQB
SECA  61            WRITE (6,1004) RQ
SECA  62            BS=SQRT(RQB)
SECA  63            TOL=ABS(RQ-RT)/RQ
SECA  64            IF (TOL.LT.RQBTOL) GO TO 150
SECA  65            DO 160 I=1,N
SECA  66     160    W(I)=W(I)/BS
SECA  67            RT=RQ
SECA  68            IF (IITE.LT.IITEM) GO TO 110
SECA  69  C
SECA  70     150    DO 170 I=1,N
SECA  71     170    V(I)=V(I)/BS
SECA  72            RB=RQ*(1.0-AMIN1(0.1,100*TOL))
SECA  73            IS=0
SECA  74     230    CALL BANDET (A,B,V,MAXA,N,NWA,RB,NSCH,DETB,ISC,1)
SECA  75            WRITE (6,1020) RB,NSCH
SECA  76            FB=DETB
SECA  77            IF (NSCH.EQ.0) GO TO 300
SECA  78            IS=IS+1
SECA  79            IF (IS.LE.NTF) GO TO 240
SECA  80            WRITE (6,1030)
SECA  81            STOP
SECA  82     240    RB=RB/(NSCH+1)
SECA  83            GO TO 230
SECA  84  C
SECA  85  C         ITERATION FOR INDIVIDUAL ROOT
SECA  86  C
SECA  87     300    WRITE (6,1040)
SECA  88            NITE(JR)=1
SECA  89            WRITE (6,1050) JR,NITE(JR),RA,DETA,FA,ETA,ISC
SECA  90            NITE(JR)=2
SECA  91            WRITE (6,1050) JR,NITE(JR),RB,DETB,FB,ETA,ISC
SECA  92  C
SECA  93  C         STOP WHEN REQUIRED NO. OF ROOTS SMALLER THAN RC AND NOV=0 FOUND
SECA  94  C
SECA  95     310    IF (NSCH.GE.NROOT) GO TO 900
SECA  96  C
SECA  97            DIF=FB FA
SECA  98            IF (DIF.NE.0.0) GO TO 320
SECA  99            WRITE (6,1060)
SECA 100            GO TO 900
SECA 101     320    DEL=FB*(RB-RA)/DIF
SECA 102            RC=RB ETA*DEL
SECA 103            TOL=RCBTOL*RC
SECA 104            IF (ABS(RC-RB).GT.TOL) GO TO 330
SECA 105            WRITE (6,1070)
SECA 106            ROOT(JR)=RB
SECA 107            GO TO 400
SECA 108  C
SECA 109     330    CALL BANDET (A,B,V,MAXA,N,NWA,RC,NSCH,DETC,ISC,1)
SECA 110            FC=DETC
SECA 111            NITE(JR)=NITE(JR)+1
SECA 112            IF (JR.EQ.1) GO TO 340
SECA 113            JJ=JR-1
SECA 114            DO 350 K=1,JJ
SECA 115     350    FC=FC/(RC-ROOT(K))
SECA 116     340    WRITE (6,1050) JR,NITE(JR),RC,DETC,FC,ETA,ISC
SECA 117  C
SECA 118  C         START INVERSE ITERATIONS
SECA 119  C
SECA 120            NES=0
SECA 121            IF (JR.EQ.1) GO TO 380
SECA 122            DO 360 I=1,JJ
SECA 123     360    IF (ROOT(I).LT.RC) NES=NES+1
```

```
  380   NOV=NSCH-NES
        IF (NOV.EQ.0) GO TO 370
        WRITE (6,1080) NOV
        ROOT(JR)=RC
        IF (NOV.GT.1) NSK=1
C
        GO TC 400
  370   RR=RA
        FR=FA
        DETR=DETA
        RA=RB
        FA=FB
        DETA=DETB
        RB=RC
        FB=FC
        DETB=DETC
C
C       RESET ETA IF NECESSARY
C
        TOL=RB*ACTOL
        IF (ABS(RA-RB).LT.TOL) ETA=ETA*2
        IF (NITE(JR).LE.NITEM) GO TO 310
        WRITE (6,1015) NITE(JR),JR
        GO TC 90C
C
C       CHECK FOR STCRAGE
C
  400   IF (JR.LE.NC) GO TO 405
        WRITE (6,1090)
        GO TC 900
C
  405   NCR=JR-1
        CALL SECCND (TIM3)
        WRITE (6,1100) NOR
        IF (JR.EC.1) GO TO 410
        DO 420 I=1,N
  420   V(I)=1.0
        KK=3
  410   DC 430 I=1,N
  430   W(I)=B(I)*V(I)
        IS=0
        R7A=C.0
        GO TC 510
C
C       INVERSE ITERATION
C
  440   NITE(JR)=NITE(JR)+1
        DO 450 I=1,N
  450   V(I)=W(I)
        CALL BANDET (A,B,V,MAXA,N,NWA,RC,NSCH,DETC,ISC,KK)
        IF (IS.EQ.1) GO TO 460
        KK=2
        RQT=C.0
        DO 470 I=1,N
  470   RQT=RQT+W(I)*V(I)
        DO 475 I=1,N
  475   W(I)=B(I)*V(I)
        RQB=C.0
        DO 480 I=1,N
  480   RQB=RQB+W(I)*V(I)
        RQ=RQT/RQB
        RT=RCCT(JR)+RQ
        WRITE (6,1110) JR,NITE(JR),RT,RQ
        TOL=RT*RCTOL
        IF (ABS(RT-RTA).GT.TOL) GO TO 510
        IS=1
        GO TC 440
C
  510   RTA=RT
        BS=SCRT(RQB)
        DO 490 I=1,N
  490   W(I)=W(I)/BS
        IF (NOR.EQ.C) GO TO 550
        DO 520 K=1,NCR
        AL=C.C
        DO 530 I=1,N
  530   AL=AL+VV(I,K)*W(I)
        DO 540 I=1,N
  540   W(I)=W(I)-AL*WW(I,K)
  520   CONTINUE
C
  550   IF (NITE(JR).LE.NITEM) GO TO 440
        WRITE (6,1015) NITE(JR),JR
        GO TC 90C
C
  460   RQT=C.0
        ERRT=RQB
        DO 57C I=1,N
  570   RQT=RQT+V(I)*W(I)
        DO 56C I=1,N
  560   W(I)=P(I)*V(I)
        RQB=0.0
        DO 580 I=1,N
  580   RQB=RQB+V(I)*W(I)
C
C       OBTAIN A RATHER LARGE ERRCR BOUND
C
        RQ=RQT/RQB
        ROOT(JR)=RCCT(JR)+RQ
        ERR=SQR7(ERRT/RQB)
        ERRVL(JR)=RCC7(JR)-ERR
        ERRVR(JR)=RCCT(JR)+ERR
C
        BS=SCRT(RQB)
        DO 590 I=1,N
        W(I)=W(I)/BS
  590   V(I)=V(I)/BS
        DO 6CC I=1,N
        WW(I,JR)=W(I)
  60C   VV(I,JR)=V(I)
C
        CALL SECCND (TIM2)
        TIM3=TIM2-TIM3
        WRITE (6,112C) TIM3
        TIM(JR)=TIM2-TIM1
        TIM1=TIM2
C
C       DECICE STRTEGY FOR ITERATION TOWARD NEXT ROOT
C
        TOL=RTOL*ROOT(JR)
        IF (NOV.GT.C) GO TO 700
        IF (ABS(RCCT(JR)-RB).GT.TCL) GC TO 710
        IF (RA.GT.0.C) GO TO 720
        RA=RB/2.
```

```
      CALL BANDET (A,B,V,MAXA,N,NWA,RA,NSCH,DETA,ISC,1)
      FA=DETA
20    RB=RA
      FB=FA
      DETB=DETA
      RA=RR
      FA=FR
      DETA=DETR
      GO TO 710

00    IF (ROOT(JR).GT.RC) NSK=1
      IF (NSK.EQ.1) GO TO 730
      IF (ABS(RC-ROOT(JR)).LT.TCL) GO TO 740
      IF (ABS(ROOT(JR)-RB).LT.TCL) GO TO 750
      RA=RB
      FA=FB
      DETA=DETB
50    RB=RC
      FB=FC
      DETB=DETC
      GO TO 710
40    IF (ABS(ROOT(JR)-RB).GT.TCL) GO TO 710
      IF (RA.GT.0.0) GO TO 760
      RA=RB/2.
      CALL BANDET (A,B,V,MAXA,N,NWA,RA,NSCH,DETA,ISC,1)
      FA=DETA
50    RB=RA
      FB=FA
      DETB=DETA
      RA=RR
      FA=FR
      DETA=DETR
10    FA=FA/(RA-ROOT(JR))
      FR=FR/(RR-ROOT(JR))
      JR=JR+1
      ETA=2.0
      GO TO 300

30    IF (RA.GT.0.0) GO TO 780
      RA=RB/2.
      CALL BANDET (A,B,V,MAXA,N,NWA,RA,NSCH,DETA,ISC,1)
      FA=DETA
80    IF (ABS(ROOT(JR)-RB).GT.TCL) GO TO 770
      RB=RA
      FB=FA
      DETB=DETA
      RA=RR
      FA=FR
      DETA=DETR
70    FA=FA/(RA-ROOT(JR))
      FB=FB/(RB-ROOT(JR))
      FR=FR/(RR-ROOT(JR))
      IF (ROOT(JR).LE.RC) NOV=NOV-1
      JR=JR+1
      NITE(JR)=0
      ROOT(JR)=RC
      IF (NOV.GT.0) GO TO 400
      NSK=0
      ETA=2.0
      GO TO 300

00    NROOT=JR-1
```

```
SECA 310          IF (NROOT.EQ.0) RETURN
SECA 311          WRITE (6,1130)
SECA 312          WRITE (6,1004) (ROOT(J),J=1,NROOT)
SECA 313          WRITE (6,1140)
SECA 314          WRITE (6,1006) (NITE(J),J=1,NROOT)
SECA 315          WRITE (6,1150)
SECA 316          WRITE (6,1008) (TIM(J),J=1,NROOT)
SECA 317          WRITE (6,1160)
SECA 318          WRITE (6,1004) (ERRVL(J),J=1,NROOT)
SECA 319          WRITE (6,1004) (ERRVR(J),J=1,NROOT)
SECA 320    C
SECA 321    C     ARRANGE EIGENVALUES AND VECTORS IN ASCENDING ORDER
SECA 322    C
SECA 323          IF (JR.EQ.2) GO TO 950
SECA 324          JR=JR-2
SECA 325     910  IS=0
SECA 326          DO 920 I=1,JR
SECA 327          IF (ROOT(I+1).GE.ROOT(I)) GO TO 920
SECA 328          IS=IS+1
SECA 329          RT=ROOT(I+1)
SECA 330          ROOT(I+1)=ROOT(I)
SECA 331          ROOT(I)=RT
SECA 332          DO 930 K=1,N
SECA 333          RT=VV(K,I+1)
SECA 334          VV(K,I+1)=VV(K,I)
SECA 335     930  VV(K,I)=RT
SECA 336     920  CONTINUE
SECA 337          IF (IS.GT.0) GO TO 910
SECA 338    C
SECA 339     950  WRITE (6,1170)
SECA 340          NROOT=NSCH
SECA 341          DO 960 I=1,NROOT
SECA 342          IF (ROOT(I).LE.0.0) GO TO 960
SECA 343          ROOT(I)=SQRT(ROOT(I))
SECA 344     960  CONTINUE
SECA 345          WRITE (6,3000) NT
SECA 346          IF (NT.NE.7) NT=7
SECA 347          REWIND NT
SECA 348          WRITE (NT) (ROOT(I),I=1,NROOT)
SECA 349    C
SECA 350    C     PRINT FREQUENCIES AND MODE SHAPES
SECA 351    C
SECA 352          WRITE (6,2000)
SECA 353          DO 990 I=1,NROOT
SECA 354          PERIOD=6.2831853/ROOT(I)
SECA 355          WRITE (6,2001) I,ROOT(I),PERIOD
SECA 356     990  ROOT(I)=PERIOD
SECA 357    C
SECA 358          WRITE (NT) ((VV(I,J),I=1,N),J=1,NROOT)
SECA 359          RETURN
SECA 360    C
SECA 361    1002  FORMAT (1H ,12E11.4)
SECA 362    1004  FORMAT (1H0,6E20.12)
SECA 363    1006  FORMAT (1H0,6I20)
SECA 364    1008  FORMAT (1H0,6F20.2)
SECA 365    1010  FORMAT (1H1,63HINVERSE ITERN GIVES FOLLOWING APPROXIMATN TO LOWES
SECA 366         1T EIGENVALUE  )
SECA 367    1015  FORMAT (41H0WE ABANDON ITERN BECAUSE NO OF ITERN IS I3,9H FOR ROW
SECA 368         1T I3 )
SECA 369    1020  FORMAT (5HORB = E20.12,7H NSCH = I4)
SECA 370    1030  FORMAT (30H0WE BETTER CHECK THE MATRICES  )
SECA 371    1040  FORMAT (1H1,4X,4HROOT,4X,4HNITE,18X,2HRC,15X,12HDET (A-RC*B),15X,
```

```
           /2HFC,13X,3HETA,4X,3HISC)
 1050 FORMAT (1H0,4X,I4,4X,I4,8X,3E22.14,F7.2,I6)
 1060 FORMAT (42HCTHE DEFLATED POLYNOMIAL HAS NO MORE ROOTS  )
 1070 FORMAT (29HC(RC-RB) IS SMALLER THAN TOL )
 1080 FORMAT (16HCWE JUMPED OVER I4,16H UNKNOWN ROOT(S)  )
 1090 FORMAT (1H1,36HNO MORE STORAGE FOR VECTORS  WE QUIT  )
 1100 FORMAT (1H0,24X,4HROOT,18X,2HRC,18X,4HNCR=,I2)
 1110 FORMAT (1H0,4X,I4,4X,I4,8X,2F22.14)
 1120 FORMAT (20H0TIME FOR INV ITERN F5.2)
 1130 FORMAT (20HCTHE EIGENVALUES ARE  /)
 1140 FORMAT (42HCNO OF ITERATIONS FOR EACH EIGENVALUE ARE   /)
 1150 FORMAT (30HCTIME USED FOR EACH EIGENVALUE /)
 1160 FORMAT (43HCFOLLOWING ARE ERROR BOUNDS ON EIGENVALUES )
 1170 FORMAT (1H1,62HWE ACCEPT FOLLOWING FREQUENCIES AND MODES ARRANGED
     . IN ORDER     )
 2000 FORMAT (39H1......PRINT OF FREQUENCIES AND PERIODS//
     .  6X,4HMODE,9X,11HFREQUENCIES,13X,7HPERIODS /
     .  6X,4H NO ,9X,11H (RAD/SEC) ,13X,7H (SEC)  /)
 2001 FORMAT (I10,2F20.4)
 3000 FORMAT (////46H EIGENVALUE AND EIGENVECTOR ARE STORED IN TAPE,I3)
C
      END


      SUBROUTINE BANDET (A,B,V,MAXA,NN,NWA,RA,NSCH,DET,ISCALE,KK)
C
      COMMON /TAPES/NSTIF,NRED,NL,NR,NT,NMASS
      DIMENSION A(NWA),B(1),V(1),MAXA(1)
C
C     TRIANGULARIZE BANDED STIFFNESS MATRIX
C
      NR=NN-1
      IF (KK-2) 100,700,800
C
  100 TOL=1.0E-07
      RTOL=1.0E-10
      SCALE=2.0**900
      NTF=3
      IS=1
  120 REWIND NSTIF
      READ (NSTIF) A
      DO 140 I=1,NN
  140 A(I)=A(I)-RA*B(I)
  160 IF (NWA.EQ.NN) GO TO 230
      DO 200 N=1,NR
      IH=N+NWA-NN
  210 IF (A(IH)) 220,215,220
  215 IH=IH-NN
      GO TO 210
  220 MAXA(N)=IH
      PIV=A(N)
      IF (PIV) 221,226,221
  221 IL=N+NN
      L=N
      DO 240 I=IL,IH,NN
      L=L+1
      C=A(I)
      IF (C) 225,240,225
  225 C=C/PIV
      IF (ABS(C).LT.TOL) GO TO 235
```

```
  226 IS=IS+1
      IF (IS.LE.NTF) GO TO 245
      WRITE (6,1000) NTF
      STOP
  245 RA=RA*(1.0-RTOL)
      GO TO 120
  235 J=L-I
      DO 260 K=I,IH,NN
  260 A(K+J)=A(K+J)-C*A(K)
      A(I)=C
  240 CONTINUE
  200 CONTINUE
  230 IF (A(NN).NE.0.0) GO TO 280
      AA=ABS(A(1))
      DO 290 I=2,NR
  290 AA=AA+ABS(A(I))
      A(NN)=-(AA/NR)*1.0E-16
C
  280 NSCH=0
      ISC=0
      DET=1.0
      DO 300 I=1,NN
      IF (ABS(DET).LT.SCALE) GO TO 320
      DET=DET/SCALE
      ISC=ISC+1
  320 DET=DET*A(I)
  300 IF (A(I).LT.0.) NSCH=NSCH+1
C
      IF (ISCALE.LT.1000) GO TO 340
      ISCALE=ISC
      GO TO 900
  340 IF (ISC-ISCALE) 350,900,370
  350 DET=DET/SCALE
      GO TO 900
  370 DET=DET*SCALE
      GO TO 900
C
  700 IL=NN
      DO 400 N=1,NR
      C=V(N)
      V(N)=C/A(N)
      IF (NWA-NN) 410,400,410
  410 IL=IL+1
      IH=MAXA(N)
      K=N
      DO 420 I=IL,IH,NN
      K=K+1
  420 V(K)=V(K)-C*A(I)
  400 CONTINUE
      V(NN)=V(NN)/A(NN)
C
  800 IF (NWA-NN) 430,900,430
  430 N=NN
      DO 440 L=2,NN
      N=N-1
      IL=N+NN
      IH=MAXA(N)
      K=N
      DO 460 I=IL,IH,NN
      K=K+1
  460 V(N)=V(N)-A(I)*V(K)
  440 CONTINUE
```

```
SECA 492   900   RETURN
SECA 493   1000  FORMAT (1H1,20H TRIANG FACTORIZATN I3,32H TIMES ABONDONED,CHECK
SECA 494        1MATRICES  )
SECA 495         END
```

```
SSPA   1         SUBROUTINE SSPACEB (NEQ,MBAND,NBLOCK,NEQB,NF,NV,NWA,NWV,NWVV,NTB)
SSPA   2   C
SSPA   3         COMMON /TAPES/NSTIF,NRED,NL,NR,NT,NMASS
SSPA   4         COMMON A(1)
SSPA   5         LOGICAL CHECK
SSPA   6   C
SSPA   7         NITEM=12
SSPA   8         CHECK=.TRUE.
SSPA   9   C
SSPA  10   C     FACTORIZE STIFFNESS MATRIX
SSPA  11   C
SSPA  12         N2=1+NWA
SSPA  13         N3=N2+NWA
SSPA  14         CALL SECOND (TIM1)
SSPA  15         CALL DECOMP (A(1),A(N2),A(N3),NEQB,MBAND,NBLOCK,NWA,NTB,NSCH,NEQ)
SSPA  16         CALL SECOND (TIM2)
SSPA  17   C
SSPA  18   C     ESTABLISH STARTING TRANSFORMATION VECTORS ON TAPE NR
SSPA  19   C
SSPA  20         N2=1+NWV
SSPA  21         N3=N2+NEQB
SSPA  22         CALL INVECT (A(1),A(N2),A(N3),NBLOCK,NEQB,NV)
SSPA  23         CALL SECOND (TIM3)
SSPA  24         TIM1=TIM2-TIM1
SSPA  25         TIM2=TIM3-TIM2
SSPA  26         WRITE (6,1000) TIM1
SSPA  27         WRITE (6,1010) TIM2
SSPA  28   C
SSPA  29   C     PERFORM SUBSPACE ITERATION
SSPA  30   C
SSPA  31         DO 100 I=1,NV
SSPA  32    100  A(I)=0.0
SSPA  33         NITE=0
SSPA  34    200  NITE=NITE+1
SSPA  35         WRITE (6,1020) NITE
SSPA  36         CALL SECOND (TIM1)
SSPA  37         N1=1+2*NV
SSPA  38         N2=N1+NWA
SSPA  39         N3=N2+NWV
SSPA  40         N4=N3+NWVV
SSPA  41         CALL REDBAK (A(N1),A(N2),A(N3),A(N4),NEQB,NV,NWA,NWV,NWVV,NTB,
SSPA  42        . NBLOCK)
SSPA  43   C
SSPA  44   C     SOLVE SUBSPACE EIGENVALUE PROBLEM
SSPA  45   C
SSPA  46         N2=1+NV
SSPA  47         N3=N2+NV
SSPA  48         N4=N3+NV*NV
SSPA  49         N5=N4+NV*NV
SSPA  50         N6=N5+NV*NV
SSPA  51         N7=N6+NWV
SSPA  52         N8=N7+NWV
SSPA  53         N9=N8+NV
SSPA  54         CALL SECOND (TIM2)
SSPA  55         CALL EIGSOL (A(1),A(N2),A(N3),A(N4),A(N5),A(N6),A(N7),A(N8),A(N9)
SSPA  56        . ,NF,NV,NBLOCK,NEQB,NITE)
SSPA  57         CALL SECOND (TIM3)
SSPA  58         TIM1=TIM3-TIM1
SSPA  59         TIM2=TIM3-TIM2
SSPA  60         WRITE (6,1030) TIM1
SSPA  61         WRITE (6,1040) TIM2
```

```
SSPA  62  C
SSPA  63          IF (NITE.LT.NITEM) GO TO 200
SSPA  64  C
SSPA  65  C
SSPA  66  C       PRINT FREQUENCIES
SSPA  67  C
SSPA  68          WRITE (6,1050)
SSPA  69          WRITE (6,1060) (A(I),I=1,NF)
SSPA  70          WRITE (6,1070)
SSPA  71          NF1=NV+1
SSPA  72          NF2=NV+NF
SSPA  73          WRITE (6,1060) (A(I),I=NF1,NF2)
SSPA  74  C
SSPA  75  C       FREQUENCIES AND MODE SHAPES STORED ON TAPE NR
SSPA  76  C
SSPA  77          WRITE (6,2020) NR
SSPA  78  C
SSPA  79          IF (.NOT.CHECK) GO TO 600
SSPA  80  C
SSPA  81  C       APPLY STURM SEQUENCE CHECK
SSPA  82  C
SSPA  83          CALL SECOND (TIM1)
SSPA  84          N2=1+NV
SSPA  85          N3=N2+NV
SSPA  86          N4=N3+NWA
SSPA  87          N5=N4+NEQB
SSPA  88          N6=N5+NV
SSPA  89          N7=N6+NV
SSPA  90          N8=N7+NV
SSPA  91          CALL SCHECK (A(1),A(N2),A(N3),A(N4),A(N5),A(N6),A(N7),A(N8),NWA,
SSPA  92         . NEQB,NBLOCK,NF,NV,SHIFT,NEI)
SSPA  93          WRITE (6,1085) SHIFT
SSPA  94          N2=1+NWA
SSPA  95          N3=N2+NWA
SSPA  96          CALL DECOMP (A(1),A(N2),A(N3),NEQB,MBAND,NBLOCK,NWA,NTB,NSCH,NEQ)
SSPA  97          IF (NSCH.EQ.NEI) GO TO 500
SSPA  98          NSCH=NSCH-NEI
SSPA  99          WRITE (6,1090) NSCH
SSPA 100          GO TO 540
SSPA 101   500    WRITE (6,1100) NSCH
SSPA 102   540    CALL SECOND (TIM2)
SSPA 103          TIM2=TIM2-TIM1
SSPA 104          WRITE (6,1110) TIM2
SSPA 105  C
SSPA 106          WRITE (6,2010)
SSPA 107          WRITE (6,2011)
SSPA 108  C
SSPA 109          REWIND NR
SSPA 110          READ (NR) (A(I),I=1,NF)
SSPA 111          DO 545 N=1,NBLOCK
SSPA 112   545    READ (NR)
SSPA 113  C
SSPA 114          DO 550 I=1,NF
SSPA 115          II=NF+I-1
SSPA 116          A(II)=A(I)
SSPA 117          A(I)=6.28318529/A(I)
SSPA 118   550    WRITE (6,2012) I,A(II),A(I)
SSPA 119  C
SSPA 120   600    RETURN
SSPA 121  C
SSPA 122   1000   FORMAT (////46H0TIME FOR STIFFNESS FACTORIZATION.............
SSPA 123         .            F6.2)
```

```
SSPA 124   1010   FORMAT (46HOTIME FOR GENERATION OF INITIAL TR-VECTORS....  F6.2)
SSPA 125   1020   FORMAT (1H1,11HNO OF ITERN  I4//)
SSPA 126   1030   FORMAT (30HOTIME USED IN ITERN STEP......  F6.2)
SSPA 127   1040   FORMAT (30HOTIME FOR EIGENVALUE SOLN.....  F6.2)
SSPA 128   1050   FORMAT (1H1,26HTHE FINAL EIGENVALUES ARE /)
SSPA 129   1060   FORMAT (1H0,6E22.14)
SSPA 130   1070   FORMAT (//1H ,50HREL. TOL. REACHED BY EIGENVALUES ARE  /)
SSPA 131   1085   FORMAT (///1H0,37HSTURM SEQUENCE CHECK APPLIED AT SHIFT  E22.14)
SSPA 132   1090   FORMAT (1CHOTHERE ARE I4,21H EIGENVALUES MISSING  )
SSPA 133   1100   FORMAT (20HOWE FOUND THE LOWEST I4,12H EIGENVALUES  )
SSPA 134   1110   FORMAT (36HOTIME FOR STURM SEQUENCE CHECK.......  F6.2)
SSPA 135   2010   FORMAT (39H1......PRINT OF FREQUENCIES AND PERIODS  )
SSPA 136   2011   FORMAT (//6X,4HMODE,9X,11HFREQUENCIES,13X,7HPERIODS /
SSPA 137         .            6X,4H NO ,9X,11H (RAD/SEC) ,13X,7H (SEC)  /)
SSPA 138   2012   FORMAT (I10,2F20.8)
SSPA 139   2020   FORMAT (////61H .....FREQUENCIES AND MODE SHAPES IN BLOCK ARE ST
SSPA 140         .RED IN TAPE  ,I3////)
SSPA 141         . END


SSPA 142          SUBROUTINE DECOMP (A,B,MAXB,NEQB,MBAND,NBLOCK,NWA,NTB,NSCH,NEQ)
SSPA 143  C
SSPA 144          COMMON /TAPES/NSTIF,NRED,NL,NR,NT,NMASS
SSPA 145          DIMENSION A(NWA),B(NWA),MAXB(NEQB)
SSPA 146  C
SSPA 147  C       FACTORIZE STIFFNESS MATRIX
SSPA 148  C
SSPA 149          NEQB1=NEQB-1
SSPA 150          N1=NL
SSPA 151          N2=NR
SSPA 152          REWIND NSTIF
SSPA 153          REWIND NRED
SSPA 154          REWIND N1
SSPA 155          REWIND N2
SSPA 156          NSCH=0
SSPA 157  C
SSPA 158          DO 600 N=1,NBLOCK
SSPA 159          IF (N.NE.1) GO TO 10
SSPA 160          READ (NSTIF) A
SSPA 161          GO TO 110
SSPA 162    10    IF (NTB.EQ.1) GO TO 110
SSPA 163          REWIND N1
SSPA 164          REWIND N2
SSPA 165          READ (N1) A
SSPA 166  C
SSPA 167  C       FACTORIZE LEADING BLOCK.
SSPA 168  C
SSPA 169   110    DO 300 I=1,NEQB1
SSPA 170          PIV=A(I)
SSPA 171          IF (PIV) 120,115,130
SSPA 172   115    II=(N-1)*NEQB+I
SSPA 173          IF (II.GT.NEQ) GO TO 520
SSPA 174          WRITE (6,1000) II
SSPA 175          STOP
SSPA 176   120    NSCH=NSCH+1
SSPA 177   130    IM=I+NWA-NEQB
SSPA 178   140    IF (A(IM)) 160,150,160
SSPA 179   150    IM=IM-NEQB
SSPA 180          GO TO 140
SSPA 181   160    MAXB(I)=IM
```

```
      JL=I+1
      II=I
      DO 200 J=JL,NEQB
      II=II+NEQB
      IF (II-NWA) 170,170,300
  170 C=A(II)
      IF (C) 180,200,180
  180 C=C/PIV
      KK=J
      MAX=MAXB(I)
      DO 250 JJ=II,MAX,NEQB
      A(KK)=A(KK)-C*A(JJ)
  250 KK=KK+NEQB
  200 A(II)=C
  300 CONTINUE
      IF (A(NEQB)) 80,60,70
   60 II=N*NEQB
      IF (II.GT.NEQ) GO TO 520
      WRITE (6,1000)II
      STOP
   80 NSCH=NSCH+1
   70 DO 50 J=NEQB,NWA,NEQB
   50 IF (A(J).NE.0.0) MAXB(NEQB)=J
C
C     CARRY OVER INTO TRAILING BLOCKS
C
      DO 400 NN=1,NTB
      IF ((NN+N).GT.NBLOCK) GO TO 400
      NI=N1
      IF ((N.EQ.1).OR.(NN.EQ.NTB)) NI=NSTIF
      READ (NI) B
      IL=1+NN*NEQB*NEQB
      DO 420 I=1,NEQB
      II=IL
      DO 440 K=1,NEQB
      IF (II-NWA) 410,410,440
  410 C=B(II)
      IF (C) 430,440,430
  430 C=C/A(K)
      MAX=MAXB(K)
      KK=I
      DO 460 JJ=II,MAX,NEQB
      B(KK)=B(KK)-C*A(JJ)
  460 KK=KK+NEQB
      A(II)=C
  440 II=II-NEQB1
  420 IL=IL+NEQB
      IF (NTB.NE.1) GO TO 480
      WRITE (NRED) A,MAXB
      DO 500 I=1,NWA
  500 A(I)=B(I)
      GO TO 600
  480 WRITE (N2) B
  400 CONTINUE
      M=N1
      N1=N2
      N2=M
  520 WRITE (NRED) A,MAXB
  600 CONTINUE
C
      RETURN
 1000 FORMAT (22H0PIVOT IS ZERO IN ROW  I4)
```

```
      END
```

```
      SUBROUTINE INVECT (VA,XM,IEQ,NBLOCK,NEQB,NV)
C
      COMMON /TAPES/NSTIF,NRED,NL,NR,NT,NMASS
      DIMENSION VA(NEQB,NV),XM(NEQB),IEQ(1)
C
      NV1=NV-1
      KK=1
      IND=0
   90 NBV=KK*((NV1-1)/NBLOCK+1)
      IF (NBV.GT.NEQB) NBV=NEQB
      IF (NBV.EQ.NEQB) IND=1
      NBVN=0
      ICOUNT=0
      LL=0
C
      REWIND NMASS
      REWIND NSTIF
   60 READ (NMASS) XM
      READ (NSTIF) (VA(I),I=1,NEQB)
      ICOUNT=ICOUNT+1
      DO 20 I=1,NEQB
      IF (VA(I).EQ.0.0) GO TO 20
      VA(I)=XM(I)/VA(I)
   20 CONTINUE
C
      NNV=NEQB/NBV
      DO 40 L=1,NBV
      RT=0.0
      NN=L*NNV
      DO 34 I=1,NN
      IF (VA(I).LT.RT) GO TO 34
      RT=VA(I)
      IJ=I
   34 CONTINUE
      DO 30 I=NN,NEQB
      IF (VA(I).LE.RT) GO TO 30
      RT=VA(I)
      IJ=I
   30 CONTINUE
      IF (VA(IJ).NE.0.0) GO TO 32
      NBVN=NBVN+1
      GO TO 40
   32 LL=LL+1
      IEQ(LL)=(ICOUNT-1)*NEQB+IJ
      IF (LL.GE.NV1) GO TO 50
      VA(IJ)=0.0
   40 CONTINUE
      IF (IND.EQ.1) GO TO 45
      IF ((NBVN.EQ.0).OR.(ICOUNT.EQ.NBLOCK)) GO TO 45
      NBV=KK*((NV1-LL-1)/(NBLOCK-ICOUNT)+1)
      IF (NBV.GT.NEQB) NBV=NEQB
      NBVN=0
C
   45 IF (ICOUNT.LT.NBLOCK) GO TO 60
      IF (IND.EQ.1) GO TO 47
      KK=2*KK
      GO TO 90
```

```
SSPA 302   47     WRITE (6,1000)
SSPA 303          STOP
SSPA 304  C
SSPA 305   50     REWIND NMASS
SSPA 306          REWIND NR
SSPA 307          DO 100 L=1,NBLOCK
SSPA 308          READ (NMASS) XM
SSPA 309          DO 120 I=1,NEQB
SSPA 310          VA(I,1)=XM(I)
SSPA 311          DO 120 J=2,NV
SSPA 312   120    VA(I,J)=0.0
SSPA 313          DO 140 K=2,NV
SSPA 314          II=IEQ(K-1)
SSPA 315          NLE=(L-1)*NEQB
SSPA 316          NRI=L*NEQB
SSPA 317          IF (II-NLE) 140,140,160
SSPA 318   160    IF (NRI-II) 140,180,180
SSPA 319   180    II=II-NLE
SSPA 320          VA(II,K)=1.
SSPA 321   140    CONTINUE
SSPA 322          WRITE (NR) VA
SSPA 323   100    CONTINUE
SSPA 324  C
SSPA 325          RETURN
SSPA 326   1000   FORMAT (42H0WE DO NOT HAVE ENOUGH FINITE EIGENVALUES   )
SSPA 327          END


SSPA 328          SUBROUTINE REDBAK (A,VA,VV,MAXB,NEQB,NV,NWA,NWV,NWVV,NTB,NBLOCK)
SSPA 329  C
SSPA 330          COMMON /TAPES/NSTIF,NRED,NL,NR,NT,NMASS
SSPA 331          DIMENSION A(NWA),VA(NWV),VV(NWVV),MAXB(NEQB)
SSPA 332  C
SSPA 333          NEB=NTB*NEQB
SSPA 334          NEBT=NEB*NEQB
SSPA 335  C
SSPA 336  C       REDUCE VECTORS ON TAPE NR
SSPA 337  C
SSPA 338          REWIND NRED
SSPA 339          REWIND NR
SSPA 340          REWIND NL
SSPA 341          REWIND NT
SSPA 342          READ (NRED) A,MAXB
SSPA 343          ISV=NTB+1
SSPA 344          LL=0
SSPA 345          DO 10 L=1,ISV
SSPA 346          READ (NL) VA
SSPA 347          K=0
SSPA 348          KK=LL
SSPA 349          DO 20 J=1,NV
SSPA 350          DO 30 I=1,NEQB
SSPA 351          K=K+1
SSPA 352          KK=KK+1
SSPA 353   30     VV(KK)=VA(K)
SSPA 354   20     KK=KK+NEB
SSPA 355   10     LL=LL+NEQB
SSPA 356          ISA=1
SSPA 357  C
SSPA 358   500    DO 100 I=1,NEQB
SSPA 359          IL=I+NEQB
```

```
SSPA 360          MAX=MAXB(I)
SSPA 361          J=0
SSPA 362          DO 120 II=IL,MAX,NEQB
SSPA 363          J=J+1
SSPA 364          C=A(II)
SSPA 365          IF (C) 110,120,110
SSPA 366   110    KK=I+J
SSPA 367          JJ=I
SSPA 368          DO 140 L=1,NV
SSPA 369          VV(KK)=VV(KK)-C*VV(JJ)
SSPA 370          KK=KK+NEBT
SSPA 371   140    JJ=JJ+NEBT
SSPA 372   120    CONTINUE
SSPA 373   100    CONTINUE
SSPA 374          DO 200 I=1,NEQB
SSPA 375          C=A(I)
SSPA 376          IF (C) 180,200,180
SSPA 377   180    KK=I
SSPA 378          DO 210 L=1,NV
SSPA 379          VV(KK)=VV(KK)/C
SSPA 380   210    KK=KK+NEBT
SSPA 381   200    CONTINUE
SSPA 382          IF (ISA.EQ.NBLOCK) GO TO 400
SSPA 383          READ (NRED) A,MAXB
SSPA 384          ISA=ISA+1
SSPA 385  C
SSPA 386  C       STORE REDUCED VECTORS ON TAPE NT
SSPA 387  C
SSPA 388          K=0
SSPA 389          KK=0
SSPA 390          DO 240 J=1,NV
SSPA 391          DO 220 I=1,NEQB
SSPA 392          K=K+1
SSPA 393          KK=KK+1
SSPA 394   220    VA(K)=VV(KK)
SSPA 395   240    KK=KK+NEB
SSPA 396          WRITE (NT) VA
SSPA 397          K=1
SSPA 398          DO 310 J=1,NV
SSPA 399          DO 300 I=1,NEB
SSPA 400          KK=K+NEQB
SSPA 401          VV(K)=VV(KK)
SSPA 402   300    K=K+1
SSPA 403   310    K=K+NEQB
SSPA 404          IF (ISV.EQ.NBLOCK) GO TO 500
SSPA 405          READ (NR) VA
SSPA 406          ISV=ISV+1
SSPA 407          KK=NEB
SSPA 408          K=0
SSPA 409          DO 330 J=1,NV
SSPA 410          DO 320 I=1,NEQB
SSPA 411          K=K+1
SSPA 412          KK=KK+1
SSPA 413   320    VV(KK)=VA(K)
SSPA 414   330    KK=KK+NEB
SSPA 415          GO TO 500
SSPA 416  C
SSPA 417  C       BACKSUBSTITUTE VECTORS ON TAPE NT
SSPA 418  C
SSPA 419   400    BACKSPACE NRED
SSPA 420          ISA=1
SSPA 421   420    DO 600 I=1,NEQB
```

```
SSPA 422          J=NEQB+1-I
SSPA 423          MAX=MAXB(J)
SSPA 424          IF (A(J)) 440,600,440
SSPA 425   440    KK=J
SSPA 426          DO 620 L=1,NV
SSPA 427          JJ=KK+I
SSPA 428          IL=J+NEQB
SSPA 429          C=VV(KK)
SSPA 430          DO 640 II=IL,MAX,NEQB
SSPA 431          C=C-A(II)*VV(JJ)
SSPA 432   640    JJ=JJ+1
SSPA 433          VV(KK)=C
SSPA 434   620    KK=KK+NEBT
SSPA 435   600    CONTINUE
SSPA 436          KK=0
SSPA 437          K=0
SSPA 438          DO 660 J=1,NV
SSPA 439          DO 670 I=1,NEQB
SSPA 440          K=K+1
SSPA 441          KK=KK+1
SSPA 442   670    VA(K)=VV(KK)
SSPA 443   660    KK=KK+NEB
SSPA 444          WRITE (NL) VA
SSPA 445          IF (ISA.EQ.NBLOCK) GO TO 800
SSPA 446          BACKSPACE NRED
SSPA 447          READ (NRED) A,MAXB
SSPA 448          BACKSPACE NRED
SSPA 449          ISA=ISA+1
SSPA 450          BACKSPACE NT
SSPA 451          READ (NT) VA
SSPA 452          BACKSPACE NT
SSPA 453          K=NEBT
SSPA 454          DO 700 J=1,NV
SSPA 455          DO 720 I=1,NEB
SSPA 456          KK=K-NEQB
SSPA 457          VV(K)=VV(KK)
SSPA 458   720    K=K-1
SSPA 459   700    K=K+NEBT+NEB
SSPA 460          K=0
SSPA 461          KK=0
SSPA 462          DO 740 J=1,NV
SSPA 463          DO 760 I=1,NEQB
SSPA 464          K=K+1
SSPA 465          KK=KK+1
SSPA 466   760    VV(KK)=VA(K)
SSPA 467   740    KK=KK+NEB
SSPA 468          GO TO 420
SSPA 469   800    RETURN
SSPA 470          END


SSPA 471          SUBROUTINE EIGSOL (DL,RTOLV,AR,BR,VEC,VL,VR,D,XM,NF,NV,NBLOCK,
SSPA 472         . NEQB,NITE)
SSPA 473    C
SSPA 474          COMMON /TAPES/NSTIF,NRED,NL,NR,NT,NMASS
SSPA 475          DIMENSION AR(NV,NV),BR(NV,NV),VEC(NV,NV),VL(NEQB,NV),VR(NEQB,NV)
SSPA 476          DIMENSION D(NV),DL(NV),RTOLV(NV),XM(NEQB)
SSPA 477    C
SSPA 478          NITEM=12
SSPA 479          RTOL=1.0E-04
SSPA 480          TOLJ=1.0E-12
SSPA 481          REWIND NMASS
SSPA 482          REWIND NT
SSPA 483          REWIND NR
SSPA 484    C
SSPA 485    C     FIND PROJECTIONS OF MASS AND STIFFNESS OPERATORS
SSPA 486    C
SSPA 487          DO 100 I=1,NV
SSPA 488          DO 100 J=I,NV
SSPA 489          AR(I,J)=0.0
SSPA 490   100    BR(I,J)=0.0
SSPA 491          DO 200 N=1,NBLOCK
SSPA 492          BACKSPACE NL
SSPA 493          READ (NL) VL
SSPA 494          BACKSPACE NL
SSPA 495          READ (NR) VR
SSPA 496          READ (NMASS) XM
SSPA 497          DO 220 I=1,NV
SSPA 498          DO 220 J=I,NV
SSPA 499          ART=0.0
SSPA 500          DO 230 K=1,NEQB
SSPA 501   230    ART=ART+VL(K,I)*VR(K,J)
SSPA 502   220    AR(I,J)=AR(I,J)+ART
SSPA 503          DO 240 I=1,NEQB
SSPA 504          XMM=XM(I)
SSPA 505          DO 240 J=1,NV
SSPA 506   240    VR(I,J)=VL(I,J)*XMM
SSPA 507          WRITE (NT) VR
SSPA 508          DO 260 I=1,NV
SSPA 509          DO 260 J=I,NV
SSPA 510          BRT=0.0
SSPA 511          DO 280 K=1,NEQB
SSPA 512   280    BRT=BRT+VL(K,I)*VR(K,J)
SSPA 513   260    BR(I,J)=BR(I,J)+BRT
SSPA 514   200    CONTINUE
SSPA 515          DO 290 I=1,NV
SSPA 516          DO 290 J=1,I
SSPA 517          AR(I,J)=AR(J,I)
SSPA 518   290    BR(I,J)=BR(J,I)
SSPA 519    C
SSPA 520    C     SOLVE EIGENVALUE PROBLEM
SSPA 521    C
SSPA 522          CALL JACOBI (AR,BR,VEC,D,VL,NV,TOLJ)
SSPA 523          DO 295 J=1,NV
SSPA 524          XMM=SQRT(BR(J,J))
SSPA 525          DO 295 K=1,NV
SSPA 526   295    VEC(K,J)=VEC(K,J)/XMM
SSPA 527    C
SSPA 528    C     ARRANGE EIGENVALUES
SSPA 529    C
SSPA 530          NV1=NV-1
SSPA 531   440    IS=0
SSPA 532          DO 400 I=1,NV1
SSPA 533          IF (D(I+1).GE.D(I)) GO TO 400
SSPA 534          IS=IS+1
SSPA 535          BT=BR(I+1,I+1)
SSPA 536          DT=D(I+1)
SSPA 537          BR(I+1,I+1)=BR(I,I)
SSPA 538          D(I+1)=D(I)
SSPA 539          BR(I,I)=BT
SSPA 540          D(I)=DT
SSPA 541          DO 420 K=1,NV
```

```
      TEMP=VEC(K,I+1)
      VEC(K,I+1)=VEC(K,I)
  420 VEC(K,I)=TEMF
  400 CONTINUF
      IF (IS.GT.0) GO TO 440
C
C     CHECK FOR CONVERGENCE
C
      DO 300 I=1,NV
      DIF=ABS(DL(I)-D(I))
  300 RTOLV(I)=DIF/D(I)
      WRITE (6,1040)
      WRITE (6,1000) (RTOLV(I),I=1,NV)
      DO 320 I=1,NF
      IF (RTOLV(I).GT.RTOL) GO TO 340
  320 CONTINUE
      WRITE (6,1050) RTOL
      NITE=NITEM
      GO TC 35C
  340 IF (NITE.LT.NITEM) GO TO 360
      WRITE (6,106C)
  35C DO 354 I=1,NV
      DL(I)=D(I)
      IF (D(I).LE.0.0) STOP
  354 D(I)=SQRT(D(I))
      M=NT
      NT=NL
      NL=M
      M=NR
      NR=NL
      NL=M
C
      REWIND NR
      WRITE (NR) ((D(I),I=1,NF)
      GO TC 43C
C
C     CALCULATE APPROXIMATE EIGEN DIRECTIONS
C
  36C DO 410 I=1,NV
  410 DL(I)=D(I)
      REWIND NR
  43C REWIND NT
      DO 460 N=1,NBLOCK
      READ (NT) VR
      DO 480 J=1,NV
      DO 480 I=1,NEQB
      TEMP=0.0
      DO 500 K=1,NV
  50C TEMP=TEMP+VR(I,K)*VEC(K,J)
  480 VL(I,J)=TEMP
  460 WRITE (NR) VL
C
      RETURN
C
 1000 FORMAT (12E11.4)
 1040 FORMAT (//32HOREL TOL REACHED ON EIGENVALUES  )
 1050 FORMAT (//25HOCONVERGENCE FOR RTOL           E10.4)
 1060 FORMAT (//31HOWE ACCEPT CURRENT ITERN VALUES  )
      END
```

```
      SUBROUTINE JACOBI (A,B,X,EIGV,D,N,RTOL)
C
      DIMENSION A(N,N),B(N,N),X(N,N),EIGV(N),D(N)
C
      NSMAX=15
      DO 10 I=1,N
      D(I)=A(I,I)/B(I,I)
   10 EIGV(I)=D(I)
      IF (N.EQ.1) RETURN
      DO 30 I=1,N
      DO 20 J=1,N
   20 X(I,J)=0.
   30 X(I,I)=1.0
      NSWEEP=0
      NR=N-1
C
C     START ITERATION
C
   40 NSWEEP=NSWEEP+1
      WRITE (6,1000) NSWEEP
      EPS=(0.01**NSWEEP)**2
      DO 50 J=1,NR
      JJ=J+1
      DO 50 K=JJ,N
      TT=A(J,K)*A(J,K)
      TB=A(J,J)*A(K,K)
      EPTOLA=ABS(TT/TB)
      TT=B(J,K)*B(J,K)
      TB=B(J,J)*B(K,K)
      EPTOLB=TT/TB
      IF ((EPTOLA.LT.EPS).AND.(EPTOLB.LT.EPS)) GO TO 50
      AKK=A(K,K)*B(J,K)-B(K,K)*A(J,K)
      AJJ=A(J,J)*B(J,K)-B(J,J)*A(J,K)
      AB=A(J,J)*B(K,K)-A(K,K)*B(J,J)
      CHECK=(AB*AB+4.0*AKK*AJJ)/4.0
      IF (CHECK) 60,70,70
   60 WRITE (6,1004) CHECK
      STOP
   70 SQCH=SQRT(CHECK)
      D1=AB/2.0+SQCH
      D2=AB/2.0-SQCH
      DEN=D1
      IF (ABS(D2).GT.ABS(D1)) DEN=D2
      IF (DEN) 90,80,90
   80 CA=0.
      CG=-A(J,K)/A(K,K)
      GO TO 100
   90 CA=AKK/DEN
      CG=-AJJ/DEN
C
C     PERFORM GENERALIZED ROTATION
C
  100 IF (N-2) 95,180,95
   95 JP1=J+1
      JM1=J-1
      KP1=K+1
      KM1=K-1
C
      IF (JM1-1) 120,110,110
  110 DO 105 I=1,JM1
      AJ=A(I,J)
```

```
SSPA 662          BJ=B(I,J)
SSPA 663          AK=A(I,K)
SSPA 664          BK=B(I,K)
SSPA 665          A(I,J)=AJ+CG*AK
SSPA 666          B(I,J)=BJ+CG*BK
SSPA 667          A(I,K)=AK+CA*AJ
SSPA 668    105   B(I,K)=BK+CA*BJ
SSPA 669   C
SSPA 670    120   IF (KP1-N) 130,130,140
SSPA 671    130   DO 125 I=KP1,N
SSPA 672          AJ=A(J,I)
SSPA 673          BJ=B(J,I)
SSPA 674          AK=A(K,I)
SSPA 675          BK=B(K,I)
SSPA 676          A(J,I)=AJ+CG*AK
SSPA 677          B(J,I)=BJ+CG*BK
SSPA 678          A(K,I)=AK+CA*AJ
SSPA 679    125   B(K,I)=BK+CA*BJ
SSPA 680   C
SSPA 681    140   IF (JP1-KM1) 150,150,180
SSPA 682    150   DO 160 I=JP1,KM1
SSPA 683          AJ=A(J,I)
SSPA 684          BJ=B(J,I)
SSPA 685          AK=A(I,K)
SSPA 686          BK=B(I,K)
SSPA 687          A(J,I)=AJ+CG*AK
SSPA 688          B(J,I)=BJ+CG*BK
SSPA 689          A(I,K)=AK+CA*AJ
SSPA 690    160   B(I,K)=BK+CA*BJ
SSPA 691    180   AK=A(K,K)
SSPA 692          BK=B(K,K)
SSPA 693          A(K,K)=AK+2*CA*A(J,K)+CA*CA*A(J,J)
SSPA 694          B(K,K)=BK+2*CA*B(J,K)+CA*CA*B(J,J)
SSPA 695          A(J,J)=A(J,J)+2*CG*A(J,K)+CG*CG*AK
SSPA 696          B(J,J)=B(J,J)+2*CG*B(J,K)+CG*CG*BK
SSPA 697          A(J,K)=0.0
SSPA 698          B(J,K)=0.0
SSPA 699   C
SSPA 700   C      UPDATE EIGENVECTORS
SSPA 701   C
SSPA 702          DO 190 I=1,N
SSPA 703          XJ=X(I,J)
SSPA 704          XK=X(I,K)
SSPA 705          X(I,J)=XJ+CG*XK
SSPA 706    190   X(I,K)=XK+CA*XJ
SSPA 707   C
SSPA 708     50   CONTINUE
SSPA 709   C
SSPA 710          DO 220 I=1,N
SSPA 711    220   EIGV(I)=A(I,I)/B(I,I)
SSPA 712          WRITE (6,1005)
SSPA 713          WRITE (6,1002) (EIGV(I),I=1,N)
SSPA 714   C
SSPA 715   C      CHECK FOR CONVERGENCE
SSPA 716   C
SSPA 717          DO 240 I=1,N
SSPA 718          TOL=RTOL*D(I)
SSPA 719          DIF=ABS(EIGV(I)-D(I))
SSPA 720    240   IF (DIF.GT.TOL) GO TO 300
SSPA 721   C
SSPA 722   C      CHECK IF ALL OFF-DIAG ELEMENTS ARE SATISFACTORILY SMALL
SSPA 723   C
SSPA 724          EPS=RTOL**2
SSPA 725          DO 260 J=1,NR
SSPA 726          JJ=J+1
SSPA 727          DO 260 K=JJ,N
SSPA 728          TT=A(J,K)*A(J,K)
SSPA 729          TB=A(J,J)*A(K,K)
SSPA 730          EPSA=ABS(TT/TB)
SSPA 731          TT=B(J,K)*B(J,K)
SSPA 732          TB=B(J,J)*B(K,K)
SSPA 733          EPSB=TT/TB
SSPA 734          IF ((EPSA.LT.EPS).AND.(EPSB.LT.EPS)) GO TO 260
SSPA 735          GO TO 300
SSPA 736    260   CONTINUE
SSPA 737   C
SSPA 738          DO 310 I=1,N
SSPA 739          DO 310 J=I,N
SSPA 740          B(J,I)=B(I,J)
SSPA 741    310   A(J,I)=A(I,J)
SSPA 742          RETURN
SSPA 743   C
SSPA 744    300   DO 320 I=1,N
SSPA 745    320   D(I)=EIGV(I)
SSPA 746          IF (NSWEEP.LT.NSMAX) GO TO 40
SSPA 747          DO 330 I=1,N
SSPA 748          DO 330 J=I,N
SSPA 749          B(J,I)=B(I,J)
SSPA 750    330   A(J,I)=A(I,J)
SSPA 751          RETURN
SSPA 752   C
SSPA 753   1000   FORMAT (1H0,14HNO OF SWEEP =  I4)
SSPA 754   1002   FORMAT (     12E11.4)
SSPA 755   1004   FORMAT (8H0CHECK =  E20.14)
SSPA 756   1005   FORMAT (24H0CURRENT EIGENVALUES ARE   )
SSPA 757          END


SSPA 758          SUBROUTINE SCHECK (DL,RTOLV,A,XM,BUP,BLO,BUPC,NEIV,NMA,NEQB,
SSPA 759         . NBLOCK,NF,NV,SHIFT,NEI)
SSPA 760   C
SSPA 761          COMMON /TAPES/NSTIF,NRED,NL,NR,NT,NMASS
SSPA 762          DIMENSION A(NMA),XM(NEQB),BUP(NV),BLO(NV),BUPC(NV),DL(NV),
SSPA 763         1RTOLV(NV)
SSPA 764          INTEGER NEIV(NV)
SSPA 765   C
SSPA 766          RTOL=1.0E-06
SSPA 767          RTOL=1.0E-04
SSPA 768          FTOL=1.0E-02
SSPA 769   C
SSPA 770          DO 100 I=1,NV
SSPA 771          BUP(I)=DL(I)*(1.0+FTOL)
SSPA 772    100   BLO(I)=DL(I)*(1.0-FTOL)
SSPA 773          NROOT=0
SSPA 774          DO 120 I=1,NF
SSPA 775    120   IF (RTOLV(I).LT.RTOL) NROOT=NROOT+1
SSPA 776          IF (NROOT.GE.1) GO TO 200
SSPA 777          WRITE (6,1010)
SSPA 778          STOP
SSPA 779   C
SSPA 780   C      FIND UPPER BOUNDS ON EIGENVALUE CLUSTERS
SSPA 781   C
```

```
  200 DO 240 I=1,NROOT
  240 NEIV(I)=1
      IF (NROOT.NE.1) GO TO 260
      BUPC(1)=BUP(1)
      LM=1
      L=1
      I=2
      GO TO 295
  260 L=1
      I=2
  270 IF (BUP(I-1).LE.BLO(I)) GO TO 280
      NEIV(L)=NEIV(L)+1
      I=I+1
      IF (I.LE.NROOT) GO TO 270
  280 BUPC(L)=BUP(I-1)
      IF (I.GT.NROOT) GO TO 290
      L=L+1
      I=I+1
      IF (I.LE.NROOT) GO TO 270
      BUPC(L)=BUP(I-1)
  290 LM=L
  295 IF (BUP(I-1).LE.BLO(I)) GO TO 300
      IF (RTOLV(I).GT.RTOL) GO TO 300
      BUPC(L)=BUP(I)
      NEIV(L)=NEIV(L)+1
      NROOT=NROOT+1
      IF (NROOT.EQ.NV) GO TO 300
      I=I+1
      GO TO 295
C
C     FIND SHIFT
C
  300 WRITE (6,1020)
      WRITE (6,1005) (BUPC(I),I=1,LM)
      WRITE (6,1030)
      WRITE (6,1006) (NEIV(I),I=1,LM)
      LL=LM-1
      IF (LM.EQ.1) GO TO 310
  330 DO 320 I=1,LL
  320 NEIV(L)=NEIV(L)+NEIV(I)
      L=L-1
      LL=LL-1
      IF (L.NE.1) GO TO 330
      WRITE (6,1040)
      WRITE (6,1006) (NEIV(I),I=1,LM)
  310 DO 340 I=1,LM
      IF (NEIV(I).GE.NROOT) GO TO 350
  340 CONTINUE
  350 SHIFT=BUPC(I)
      NEI=NEIV(I)
C
C     SHIFT MATRIX
C
      REWIND NSTIF
      REWIND NMASS
      REWIND NRED
      DO 400 L=1,NBLOCK
      READ (NSTIF) A
      READ (NMASS) XM
      DO 420 I=1,NEQB
  420 A(I)=A(I)-SHIFT*XM(I)
      WRITE (NRED) A
  400 CONTINUE
      I=NSTIF
      NSTIF=NRED
      NRED=I
      RETURN
C
 1005 FORMAT (1H0,6E20.12)
 1006 FORMAT (1H0,6I20)
 1010 FORMAT (30H0CONVERGENCE FOR NO EIGENVALUE  )
 1020 FORMAT (37H0UPPER BOUNDS ON EIGENVALUE CLUSTERS  )
 1030 FORMAT (34H0NO OF EIGENVALUES IN EACH CLUSTER  )
 1040 FORMAT (42H0NO OF EIGENVALUES LESS THAN UPPER BOUNDS  )
      END
```

```
      SUBROUTINE HISTRY
C
C     TIME HISTORY RESPONSE ANALYSIS TO APPLIED DYNAMIC LOADS AND/OR
C     RIGID OR MULTIPLE GROUND MOTION INPUTS
C
      COMMON/MISC/ NBLOCK,NEQB,LL,NF,LB,NSVV,NFN,NGM,NAT,NT,NOT
      COMMON/ELPAR/ NPAR(14),NUMNP,MBAND,NELTYP,N1,N2,N3,N4,N5,MTOT,NEQ
      COMMON/EM/ AT(1058)
      COMMON/JUNK/ NAR8,NGN
      COMMON/DYN/ NTT,NOTT,DAMP,DT
      DIMENSION T(7)
      COMMON A(1)
C
C     MODE SHAPES AND FREQUENCIES
C
      CALL SECOND (T(1))
      CALL DYNAM (NEQ,MBAND,NBLOCK,NEQB,NF,NSVV,MTOT)
C
C     READ HISTORY ANALYSIS CONTROL DATA
C
      READ (5,1000) DT,DAMP
      NTT=NT
      NOTT=NOT
C
C     ADJUST NUMBER OF FREQUENCIES TO BE SMALLER THAN 5*DT
C
      N1=1
      N2=0
      TT=DT*5.
      DO 200 I=1,NF
      IF (A(I).GT.TT) N2=I
  200 CONTINUE
      NF=N2
      WRITE(6,2000) NFN,NGM,NAT,NT,NOT,DT,DAMP,NF
      NGN=NGM
      IF (NF.NE.0) GO TO 300
      WRITE(6,4000)
      STOP
  300 IF (NAT.EQ.0) NAT=1
C
C     FORM DYNAMIC LOAD VECTORS
C
      CALL SECOND (T(2))
C
C     A.  DYNAMIC FORCING FUNCTIONS
C
      N2=N1+6*NUMNP
      N3=N2+NFN*NEQB
      N4=N3+NFN*NEQB
      IF (N4.GT.MTOT) CALL ERROR (N4-MTOT)
      CALL LOAD1 (A(N1),A(N2),A(N3),NUMNP,NEQB,NFN,NEQ)
      N2=NGM-1
      IF (N2) 600,500,400
C
C     B.  ADD MULTIPLE GROUND MOTION EFFECTS
C
  400 NSB=(MBAND+LL)*NEQB
      NSB8=NEQB*LL*(2+(MBAND-1)/NEQB)
      IF (NSB8.LT.NSB) NSB8=NSB
      N2=N1+NEQB*MBAND
      N3=N2+NEQB*LL
      N4=N3+NSB8
      CALL USOL(A(N1),A(N3),A(N4),NEQB,MBAND,LL,NBLOCK,NSB,12,3,10,11,11
     .)
C
      N2=N1+LL
      N3=N2+LL
      N4=N3+6*NUMNP
      N5=N4+NEQB
      N6=N5+NEQB*LL
      CALL MGMTN(A(N1),A(N2),A(N3),A(N4),A(N5),NEQB,LL,NUMNP,NBLOCK,NEQ)
      GO TO 600
C
C     C.  ADD RIGID GROUND MOTION EFFECTS
C
  500 N2=N1+6*NUMNP
      CALL EMID (A(N1),A(N2),NUMNP,NEQB,NEQ)
      N2=N1+NEQB*NFN
      N3=N2+NEQB*NFN
      N4=N3+NEQB
      N5=N4+NEQB
      IF (N5.GT.MTOT) CALL ERROR (N5-MTOT)
      CALL GMTN (A(N1),A(N2),A(N3),A(N4),NEQB,NFN,NBLOCK)
C
C     FORM MODAL LOAD VECTORS
C
  600 N2=N1+NFN*NF*NAT
      N3=N2+NEQB*NF
      N4=N3+NEQB*NFN
      N5=N4+NEQB*NFN
      NN6=N5+LL
      NN7=NN6+LL
      NN8=NN7+NEQB*LL
      IF (NN8.GT.MTOT) CALL ERROR (NN6-MTOT)
      N6=N2+NT*NFN
      MAX=(MTOT-N6)/2
      N7=N6+MAX
      N8=N6+NT
      IF (N8.GT.MTOT) CALL ERROR (N8-MTOT)
      CALL LOAD2 (A(N2),A(N3),A(N4),A(N5),A(NN6),A(NN7),
     .            A(N2),A(N6),A(N7),A(N8),NEQB,NF,NFN,NT,MAX,NBLOCK,NAT,
     .            LL)
C
C     NORMAL RESPONSE
C
      CALL SECOND (T(3))
      NOS=(NT-1)/NOT
      N2=N1+NF
      N3=N2+NT
      N4=N3+NF*NOS
      IF(N4.GT.MTOT) CALL ERROR(N4-MTOT)
      CALL RESPON (A(N1),A(N2),A(N3),NF,NT,NOS)
C
C     PRINT MODE SHAPES
C
      CALL SECOND (T(4))
      N2=N1+6*NUMNP
      N3=N2+6*NF
      CALL PRINTD (A(N1),A(N2),A(N3),NEQB,NUMNP,NF,NBLOCK,NEQ,7,NF)
C
C     DISPLACEMENT RESPONSE
C
      CALL SECOND (T(5))
```

```
      NSB=NEQB*NBLOCK
      N2=N1+8*NF
      N3=N2+NF*NDS
      IF(N3.GT.MTOT) CALL ERROR(N3-MTOT)
      CALL DISPLRS(A(N1),A(N1),A(N2),A(N2),NEQB,NF,NDS,NUMNP,NBLOCK,NSB,
     .             NEQ)
C
C     STRESS RESPONSE
C
      CALL SECOND (T(6))
      N2=N1+NELTYP
      N3=N2+8*NF
      N4=N3+NSB*NF
      N5=N3+NF*NDS
      IF(N4.GT.MTOT) CALL ERROR(N4-MTOT)
      IF(N5.GT.MTOT) CALL ERROR(N5-MTOT)
      CALL STRSC1(A(N1),A(N2),A(N3),A(N3),NF,NSB,NDS,NEQB,NBLOCK)
      CALL SECOND (T(7))
      TT=0.
      DO 100 I=1,6
      T(I)=T(I+1)-T(I)
  100 TT=TT + T(I)
      T(7)=TT
      WRITE (6,3000) T
C
      RETURN
C
 1000 FORMAT (2F10.0)
 2000 FORMAT (28H1NUMBER OF DYNAMIC INPUTS...= I5//
     .          28H GROUND MOTION INDICATOR...= I5//
     .          28H NUMBER OF ARRIVAL TIMES...= I5//
     .          28H NUMBER OF TIME STEPS......= I5//
     .          28H OUTPUT INTERVAL...........= I5//
     .          28H TIME INCREMENT............= 1PE9.2//
     .          28H DAMPING FACTOR............= 1PE9.2//
     .          15H ADJUSTED NUMBER OF /
     .          28H FREQUENCIES...............= ,I5)
 3000 FORMAT (22H1....TIME LOG (SECONDS) ///
     . 33H  MODE SHAPES AND FREQUENCIES...  ,F8.2 //
     . 33H  FORM DYNAMIC LOADS............  ,F8.2 //
     . 33H  MODAL RESPONSE................  ,F8.2 //
     . 33H  PRINT MODE SHAPES.............  ,F8.2 //
     . 33H  DISPLACEMENT OUTPUT...........  ,F8.2 //
     . 33H  STRESS OUTPUT.................  ,F8.2 //
     . 33H  TOTAL FOR RESPONSE ANALYSIS...  ,F8.2 //)
 4000 FORMAT (////50H  ALL PERIODS ARE SMALLER THAN 5*DT...STOP          )
C
      END



      SUBROUTINE LOAD1(ID,FF,IFF,NUMNP,NEQB,NFN,NFQ)
C
      DIMENSION ID(NUMNP,6),FF(NEQB,NFN),IFF(NEQB,NFN)
      COMMON / JUNK / NARB,NGM
C
C     READ ARBITRARY DYNAMIC LOADS
C
      NT=2
      REWIND NT
      REWIND 8
      READ (8) ID
C
      NNN=NEQB*NFN
      DO 10 I=1,NNN
      IFF(I)=0
   10 FF(I)=0.
C
      WRITE (6,2000)
      NARB=1
      READ (5,1000) NP,IC,IFN,IAT,P
      IF(IAT.EQ.0) IAT=1
      IF(NP.GT.0) GO TO 15
      NARB=0
      RETURN
   15 WRITE (6,2002) NP,IC,IFN,IAT,P
C
      NS=1
      NE=NEQB
      DO 500 NN=1,NUMNP
      DO 500 II=1,6
C
   20 N=ID(NN,II)
      IF (N.GT.NEQ) GO TO 500
      IF(N.LE.0) GO TO 300
      IF(N.GE.NS) GO TO 50
      WRITE (6,2001)
      STOP
C
   50 IF(N.LE.NE) GO TO 300
C
      WRITE (NT) FF,IFF
      NS=NS+NEQB
      NE=NE+NEQB
      DO 100 I=1,NNN
      IFF(I)=0
  100 FF(I)=0.
      GO TO 50
C
  300 IF(NP.EQ.NN.AND.IC.EQ.II) GO TO 350
      GO TO 500
  350 M=N-NS+1
      IF(N.LE.0) GO TO 400
      FF(M,IFN)=P
      IFF(M,IFN)=IAT
  400 READ (5,1000) NP,IC,IFN,IAT,P
      IF(IAT.EQ.0) IAT=1
      WRITE(6,2002) NP,IC,IFN,IAT,P
      GO TO 300
C
  500 CONTINUE
      WRITE (NT) FF,IFF
C
      RETURN
C
 1000 FORMAT (4I5,F10.2)
 2001 FORMAT (18HODATA OUT OF ORDER )
 2000 FORMAT (15H1DYNAMIC LOAD INPUT //
     . 57H  NODE   DISPLACEMENT  FUNCTION    ARRIVAL TIME    FUNCTION /
     . 60H  NUMBER   COMPONENT     NUMBER        NUMBER      MULTIPLIER /)
 2002 FORMAT (I6,2I11,I14,F15.3)
C
      END
```

```
      SUBRCUTINE MGPTN (NFNG,NATG,ID,XM,R,NEQB,LL,NUMNP,NBLOCK,NEQ)
C
C     SET LP QUASI-STATIC INFLUENCE MATRIX FOR MULTIPLE GROUND MOTIONS
C
      DIMENSION R(NEQB,LL),NFNG(LL),NATG(LL),XM(NEQB),ID(NUMNP,6)
C
      NT=11
      REWINC 10
      REWINO 9
      REWINC 8
      READ (8) IO
      L=0
      DO 10C I=1,LL
      NFNG(I)=0
  100 NATG(I)=0
      WRITE(6,2C00)
  150 L=L+I
      READ (5,1000) NP,IC,IFN,IAT
      IF (IAT.EC.0) IAT=1
      IF (NP.LE.0) CO TO 300
      WRITE(6,2C01) NP,IC,IFN,IAT
      JJ=ID(NP,IC)
      IF (JJ.GT.NEQ) GO TO 160
      WRITE(6,2CC2) NP,IC
      STOP
  160 JJ=JJ-NFQ
      NFNG(JJ)=IFN
      NATG(JJ)=IAT
C
      IF (L.LE.LL) CO TO 150
      WRITE(6,2CC3)
      STOP
C
  300 WRITE(10) NFNG,NATG
      DO 500 N=1,NBLOCK
      READ (9) XM
      BACKSPACE NT
      READ (NT) R
      BACKSPACE NT
      DC 40C I=1,NEQB
      DO 4CC L=1,LL
      R(I,L)=-R(I,L)*XM(I)
  400 CONTINUE
      WRITE (10) R
  500 CONTINUE
C
      RETURA
C
 1000 FORMAT (4I5)
 2000 FORMAT (34H1MULTIPLE GROUND MOTION INPUT OATA  //
     . 46H  NODE   OISPLACEMENT  FUNCTION   ARRIVAL TIME  /
     . 46H   NO.       COMPONENT     NUMBER        NUMBER    /)
 2001 FORMAT (I6,2I11,I14)
 2002 FORMAT (//19H THE INPUT NODE NC. ,I3,22H  DISPL. COMPONENT NO.,I3,
     . 40HARE NCT COMPATIBLE WITH JOINT INPUT OATA  /
     . 24H ..EXECUTION TERMINATEO.   )
 2003 FORMAT (//62H NO. OF GROUND INPUT POINTS .GT. THAT SPECIFIED BY JO
     .INT INPUT  / 24H ..EXECUTION TERMINATEO.  )
C
      ENO

      SUBROUTINE GMTN (FF,IFF,XM,MASS,NEQB,NFN,NBLOCK)
C
      COMMON / JUNK / NARB,NGM,JFN(3),JAT(3)
      DIMENSION FF(NEQB,NFN),IFF(NEQB,NFN),MASS(NEQB),XM(NEQB)
C
C     RIGIO GROUNO MOTION EFFECTS
C
      JT=4
      IT=2
      REWINC IT
      REWINC JT
      REWINO 3
      REWINC 9
C
      READ (5,1CC0) JFN,JAT
      OO 100 I=1,3
      IF(JAT(I)) 5C,50,100
   50 JAT(I)=I
  100 CONTINUE
      WRITE (6,200C) JFN,JAT
C
      NNN=NFN*NEQB
      DC 500 N=I,NBLOCK
C
      READ (3) MASS
      READ (9) XM
C
      IF(NARB.EQ.0) GO TO 200
      READ (IT) FF,IFF
      GC TO 300
  200 OO 25C I=I,NNN
      FF(I)=0.
  250 IFF(I)=0
C
  300 DC 400 I=1,NEQB
      J=MASS(I)
      IF (J.LE.C) GC TO 400
      JJ=JFN(J)
      IF(JJ.LE.C) GC TO 400
      FF(I,JJ)=-XM(I)
      IFF(I,JJ)=JAT(J)
  400 CONTINUE
C
  5C0 WRITE (JT) FF,IFF
C
      RETURN
C
 1000 FORMAT (6I5)
 20C0 FORMAT (////30H1...GROUND MOTION INPUT KEY...///
     . 23X,9HDIRECTION /
     . 22X,11HX    Y    Z  //
     . 20H  FUNCTICN NUMBER....,I3,2I5/
     . 20H  ARRIVAL TIME...... ,I3,2I5)
C
      END
```

```
      SUBROUTINE LOAD2 (FI,FF,IFF,NFNG,NATG,R,
     .                  PP,T,P,PD,NEQB,NF,NFN,NT,MAX,NBLOCK,NAT,LG)
C
      COMMON / EM / AT(1058)
      DIMENSION FI(NEQB,NF),FF(NEQB,NFN),IFF(NEQB,NFN),PP(NFN,1),T(1),
     .          P(1),PD(NT),NFNG(LG),NATG(LG),R(NEQB,LG)
      COMMON / DYN / QT,NOT,DAMP,DT
      COMMON / JUNK / NARB,NGM,HED(12)
      COMMON A(1)
C
C     TRANSFORM NODAL TO MODAL LOADS
C
      MTREAD=1
      IF ((NARB.EQ.0).AND.(NGM.NE.1)) MTREAD=0
      READ (5,1002) (AT(I),I=1,NAT)
      WRITE (6,2004) (I,AT(I),I=1,NAT)
      MT=2
      IF (NGM.EQ.1) MT=4
      REWIND MT
      REWIND 10
      IF (NGM.EQ.2)  READ (10) NFNG,NATG
      NE=NAT*NF*NFN
      DO 10 I=1,NE
   10 A(I)=0.
      KK=NF*NFN
C
      DO 500 N=1,NBLOCK
      BACKSPACE 7
      READ (7) FI
      BACKSPACE 7
      IF (MTREAD.NE.0) READ (MT) FF,IFF
      IF (NGM.EQ.2)  READ (10) R
      NN=-KK
C
      DO 200 I=1,NF
      DO 200 J=1,NFN
      NN=NN+1
      IF (NGM.LT.2) GO TO 100
C
      DO 80 K=1,LG
      IF (NFNG(K).NE.J) GO TO 80
      LL=NATG(K)
      IF (LL.EQ.0) GO TO 80
      DO 50 L=1,NEQB
      II=NN+LL*KK
      A(II)=A(II)+FI(L,I)*R(L,K)
   50 CONTINUE
   80 CONTINUE
C
C
  100 IF (MTREAD.EQ.0) GO TO 200
      DO 150 L=1,NEQB
      LL=IFF(L,J)
      IF(LL.EQ.0) GO TO 150
      K=NN+LL*KK
      A(K)=A(K) + FI(L,I)*FF(L,J)
  150 CONTINUE
  200 CONTINUE
  500 CONTINUE
C
C     READ TIME FUNCTIONS AND GENERATE
C     EQUAL INTERVAL FUNCTIONS
C
      DO 335 I=1,NFN
C
      READ (5,1000) NLP,SFTR,TINCR,HED
      IF(SFTR.EQ.0.) SFTR=1.0
      WRITE (6,2000) I,HED,NLP,SFTR
      IF(NLP.LE.MAX) GO TO 300
      L=2*(NLP-MAX)
      CALL ERROR(L)
  300 IF (TINCR.LE.0.) GO TO 305
      DO 303 L=1,NLP
      TL=L-1
  303 T(L)=TINCR*TL
      READ (5,1003) (P(L),L=1,NLP)
      GO TO 308
C
  305 READ (5,1001) (T(L),P(L),L=1,NLP)
C
  308 WRITE(6,2001) (T(L),P(L),L=1,NLP)
C
      TIME=T(1)
      L=1
      K=1
  310 L=L+1
      DDT=T(L)-T(L-1)
      DDP=P(L)-P(L-1)
      IF(DDT) 315,310,320
  315 WRITE (6,2003)
  320 SLOPE=DDP/DDT
  323 IF (T(L)-TIME) 310,325,325
  325 PP(I,K)=P(L-1)+(TIME-T(L-1))*SLOPE
      PP(I,K)=PP(I,K)*SFTR
  330 TIME=TIME+DT
      K=K+1
      IF (NT-K) 335,323,323
  335 CONTINUE
C
C     GENERATE MODAL LOAD VECTORS
C
      MT=4
      REWIND MT
      LL=NF*NFN
C
      DO 900 K=1,NF
      DO 550 I=1,NT
  550 PD(I)=0.
      INC=(K-1)*NFN
C
      DO 800 J=1,NAT
      LT=AT(J)/DT + 1
      N=0
C
      DO 600 NN=LT,NT
      N=N+1
      DO 600 I=1,NFN
      II=INC+I
  600 PD(NN)=PD(NN) + A(II)*PP(I,N)
C
  800 INC=INC+LL
```

```
HIST 480  C
HIST 481     900 WRITE (MT) PC
HIST 482  C
HIST 483         RETURN
HIST 484  C
HIST 485   1000 FORMAT (I5,F10.0,F5.0,12A5)
HIST 486   1001 FORMAT (12F6.0)
HIST 487   1002 FORMAT (8F10.2)
HIST 488   1003 FORMAT (10F7.4,10X)
HIST 489   2000 FORMAT (     26H1.... TIME FUNCTION NUMBER ,I2,6X,12HHEADING ... ,
HIST 490       . 12A5//6X,23HNUMBER OF LOAD POINTS = I4,/
HIST 491       . 6X,23HSCALE FACTOR..........= ,F10.3//)
HIST 492   2001 FORMAT (5(19H   TIME    INPUT      )/(5(F7.3,F8.3,4X)))
HIST 493   2003 FORMAT (15H0BAO LOAD DATA )
HIST 494   2004 FORMAT (//////14H  DELAY  TIMES //10X,TH DELAY /
HIST 495       . 16H  NUMBER     TIME  / (I6,F10.2))
HIST 496         END



HIST 497         SUBROUTINE RESPON (W,P,X,NF,NT,NDS)
HIST 498  C
HIST 499         DIMENSION W(NF),P(NT),X(NF,NDS)
HIST 500         COMMON / DYN / MT,NOT,DAMP,DT
HIST 501         COMMON / JUNK / C1,C2,C3,C4,C5,K,NOUT,F,DISP,
HIST 502        .                VEL,ACEL,A,B,I,N
HIST 503  C
HIST 504  C      EVALUATION OF NORMAL RESPONSE
HIST 505  C
HIST 506         REWINC 7
HIST 507         REWINC 4
HIST 508         READ (7) W
HIST 509         C1=DT/2.
HIST 510         C2=C1*DT/3.
HIST 511         C3=C2*2.
HIST 512  C
HIST 513         DO 260 N=1,NF
HIST 514         READ (4) P
HIST 515         K=1
HIST 516         NOUT=NOT+1
HIST 517         C4=W(N)**2
HIST 518         C5=2.*DAMP*W(N)
HIST 519         F=1.+C1*C5+C2*C4
HIST 520         DISP=0.0
HIST 521         VEL=0.0
HIST 522         ACEL=P(1)
HIST 523  C
HIST 524         DO 260 I=2,NT
HIST 525         A=VEL+C1*ACEL
HIST 526         B=DISP+DT*VEL+C3*ACEL
HIST 527         ACEL=(P(I)-C5*A-C4*B)/F
HIST 528         VEL=A+C1*ACEL
HIST 529         DISP=B+C2*ACEL
HIST 530         IF(NOUT-I) 260,250,260
HIST 531     250 X(N,K)=DISP
HIST 532         K=K+1
HIST 533         NOUT=NOUT+NOT
HIST 534     260 CONTINUE
HIST 535  C
HIST 536         REWIND 4
HIST 537         WRITE (4) X
HIST 538  C
HIST 539         RETURN
HIST 540  C
HIST 541         END



HIST 542         SUBROUTINE DISPLKS (ID,F,FI,X,NEQB,NF,NDS,NUMNP,NBLOCK,NSB,NEQ)
HIST 543  C
HIST 544         DIMENSION ID(NUMNP,6),F(B,NF),FI(NSB ,NF),X(NF,NDS)
HIST 545         COMMON / JUNK / NP,IC(6),D(8),L,  II,MSB,NS,NE,N,DDT,M,
HIST 546        .                TIME,J,K,MM,DD,KD(3,8),XUM,IEQ,NRC
HIST 547        .                ,DM(8),TP(8)
HIST 548         COMMON / DYN / NT,NOT,DAMP,DT
HIST 549  C
HIST 550  C      EQN. NUMBERS OF SELECTED DISPL. COMPONENTS.
HIST 551  C
HIST 552         REWINC 9
HIST 553         REWINC 8
HIST 554         READ (8) ID
HIST 555         L=0
HIST 556         NUM=0
HIST 557         READ (5,2000) KKK,ISP
HIST 558         WRITE (6,1005)
HIST 559     100 READ (5,2000) NP,IC
HIST 560         WRITE (6,2001) NP,IC
HIST 561         IF(NP.GT.0) GO TO 110
HIST 562         IF(L.EQ.0) GO TO 200
HIST 563         WRITE (9) KD,L
HIST 564         NUM=NUM+1
HIST 565         GO TO 200
HIST 566     110 DO 150 I=1,6
HIST 567         II=IC(I)
HIST 568         IF(II.EQ.0) GO TO 150
HIST 569         IF (ID(NP,II).LE.NEQ) GO TO 120
HIST 570         WRITE (6,4001) NP,II
HIST 571         GO TO 150
HIST 572     120 L=L+1
HIST 573         KD(1,L)=NF
HIST 574         KD(2,L)=II
HIST 575         KD(3,L)=ID(NP,II)
HIST 576         IF(ID(NP,II).LE.0) L=L-1
HIST 577         IF(L.LT.8) GO TO 150
HIST 578         WRITE (9) KD,L
HIST 579         NUM=NUM+1
HIST 580         L=0
HIST 581     150 CONTINUE
HIST 582         GO TO 100
HIST 583  C
HIST 584  C      APPROPRIATE MODE SHAPE COMPONENTS
HIST 585  C
HIST 586     200 IF(NUM.EQ.0) RETURN
HIST 587         WRITE (6,4000) KKK,ISP
HIST 588         REWINC 3
HIST 589         REWINC 9
HIST 590         REWINC 7
HIST 591         READ (7)
HIST 592         NE=NSB
HIST 593         NS=NE+1-NEQB
HIST 594         DO 300 I=1,NBLOCK
HIST 595         READ (7) ((FI(J,K),J=NS,NE),K=1,NF)
```

```
      NS=NS-NEQB
  300 NE=NE-NEQB
C
      DO 400 N=1,NUM
      READ (9) KC,L
C
      DO 350 I=1,L
      II=KD(3,I)
      DO 350 J=1,NF
  350 F(I,J)=FI(II,J)
  400 WRITE (3) L,KC,F
C
C     COMPUTE AND OUTPUT HISTORY OF VALUES
C
  410 DT=NOT*DT
C
      CALL DISPLY (X,F,NF,NDS,NUM,1,KKK,2,ISP)
C
  900 RETURN
C
 1005 FORMAT (35H1DISPLACEMENT COMPONENTS FOR WHICH /
     . 26H  TIME HISTORY IS REQUIRED  //
     . 31H  NODE  DISPLACEMENT COMPONENTS  /)
 2000 FORMAT (7I5)
 2001 FORMAT (I5,4X,6I3)
 4000 FORMAT (/16H OUTPUT TYPE.....I1/
     .              16H PLOT SPACING....I1)
 4001 FORMAT (///5H NODE,I5,25H   DISPLACEMENT COMPONENT  ,I5,6H   IS  /
     .  60H FOR GROUND MOTION INPUT....OUTPUT DISREGARD.                )
C
      END



      SUBROUTINE DISPLY (X,F,NF,NDS,NUM,NN,KKK,ISD,ISP)
C
C     S/R TO PRINT/PLOT RESPONSE VALUES
C     ISD=1..STRESS     =2..DISPL
C     KKK=1..PRINT      =2..PLOT             =3..MAXIMUMS
C
      DIMENSION X(NF,NDS),F(8,NF),NUM(NN)
      COMMON / JUNK / KD(3,8),TM(8),DM(8),D(8)
      COMMON / DYN / NT,NOT,DAMP,DT
      COMMON / FLPAR / NPAR(14)
C
      REWIND 3
      REWIND 4
      READ (4) X
C
      DO 900 N=1,NN
      REWIND 2
      REWIND 9
      MM=NUM(N)
C
      IF(ISD.EQ.2) GO TO 90
      READ (3) NPAR
      MTYPE=NPAR(1)
   90 IF(MM.EQ.0) GO TO 900
C
      DO 600 M=1,MM
      READ (3) L,KC,F
      GO TO (100,300,200) KKK
C
C     PRINT
C
  100 IF(ISD.EQ.1) GO TO 130
      WRITE (6,1000) M
      GO TO 140
  130 WRITE (6,2000) M
  140 WRITE (6,2001) (KD(1,I),KD(2,I),I=1,L)
      GO TO 300
C
C     MAXIMUMS
C
  200 IF(M.GT.1) GO TO 300
      IF(ISD.EQ.1) GO TO 230
      WRITE (6,1002)
      WRITE (6,5001)
      GO TO 300
  230 WRITE (6,2002) MTYPE
      WRITE (6,4001)
C
C     COMPUTE HISTORY
C
  300 DO 320 I=1,L
      TM(I)=0.
  320 DM(I)=0.
      TIME=0.
C
      DO 500 K=1,NDS
      TIME=TIME + DT
      DO 450 I=1,L
      DD=0.
      DO 440 J=1,NF
  440 DD = DD + F(I,J)*X(J,K)
C
      AD=ABS(DD)
      IF(AD-DM(I)) 450,450,445
  445 DM(I)=AD
      TM(I)=TIME
C
  450 D(I)=DD
      GO TO (480,450,500) KKK
C
  480 WRITE (6,1004) TIME,(D(I),I=1,L)
      GO TO 500
C
  490 WRITE (9) D
C
  500 CONTINUE
C
      GO TO (510,520,530) KKK
C
  510 WRITE (6,1005) (DM(I),I=1,L)
      WRITE (6,1006) (TM(I),I=1,L)
      GO TO 600
C
  520 WRITE (2) KC,DM,TM,L
      GO TO 600
C
  530 WRITE (6,1007) (KD(1,I),KD(2,I),DM(I),TM(I),I=1,L)
C
  600 CONTINUE
```

```
C
C     PLOT SET OF VALUES
C
      IF(KKK.NE.2) GO TO 900
      REWIND 2
      REWIND 9
      DO 800 M=1,MM
      GO TO (610,620) ISO
C
  610 WRITE (6,4000) MTYPE,M
      WRITE (6,4001)
      GO TO 630
C
  620 WRITE (6,5000) M
      WRITE (6,5001)
C
  630 CALL PLOT (2,S,NDS,ISP)
C
  800 CONTINUE
  900 CONTINUE
C
C
      RETURN
C
 1000 FORMAT (30H1TIME HISTORY FOR SELECTED DISPLACEMENT COMPONENTS ,
     . 5H..... ,I3//10X,40HNODE NUMBERS AND DISPLACEMENT COMPONENTS )
 1002 FORMAT (55H1MAXIMUM DISPLACEMENT VALUES FROM DYNAMIC RESPONSE ANAL
     .YSIS //)
 1004 FORMAT (F8.3,2X,1P8E12.3)
 1005 FORMAT ( /24H MAXIMUM ABSOLUTE VALUES //
     . 10H MAXIMUM    ,1P8E12.3)
 1006 FORMAT (10H TIME       ,1P8E12.3)
 1007 FORMAT (I6,I13,1PE18.3,E12.3,5X,2HNA)
 2000 FORMAT (  44H1TIME HISTORY FOR SELECTED STRESS COMPONENTS ,
     . 5H..... , I3//10X,37H ELEMENT AND STRESS COMPONENT NUMBERS )
 2001 FORMAT (8H    TIME , 2X, 8(I8,1H-,I2,X))
 2002 FORMAT (16H1ELEMENT TYPE... ,I2//
     . 53H MAXIMUM STRESS VALUES FROM DYNAMIC RESPONSE ANALYSIS //)
 4000 FORMAT (17H1ELEMENT TYPE....,I2//
     . 39H NORMALISED PLOT OF STRESS HISTRIES...  ,I3/)
 4001 FORMAT (58H  ELEMENT       STRESS           MAXIMUM       TIME AT      PL
     .OT / 58H  NUMBER      COMPONENT          VALUE         MAXIMUM     SYMBOL
     .)
 5000 FORMAT (46H1NORMALISED PLOT OF DISPLACEMENT HISTORIES....,I3/)
 5001 FORMAT(58H    NODE      DISPLACEMENT      MAXIMUM       TIME AT      PLO
     .T / 58H  NUMBER     COMPONENT           VALUE         MAXIMUM     SYMBOL)
C
      END



      SUBROUTINE STRSDI (NUM,SF,FI,X,NF,NSB,NDS,NEQB,NBLOCK)
C
      DIMENSION NUM(1),SF(8,NF),FI(NSB,NF),X(NF,NDS)
      COMMON / EM / NS,ND,LM(24),SA(12,24),EXRA(744)
      COMMON /ELPAR/ NPAR(14),NUMNP,MBAND,NELTYP,N1,N2,N3,N4,N5,MTOT,NEQ
      COMMON / JUNK / N,NEL,IS(12),M,I,L,KS(3,8),II,K,J,SS,JJ
     .                ,NUME,NE
C
C     ASSEMBLE MODE SHAPES IN CORE
C
      REWIND 7
      READ (7)
      NE=NSB
      NS=NE+1-NEQB
      DO 100 I=1,NBLOCK
      READ (7) ((FI(J,K),J=NS,NE),K=1,NF)
      NS=NS-NEQB
  100 NE=NE-NEQB
C
C     FORM STRESS MATRIX,MODE SHAPE TRANSFORMATION FOR
C     SELECTED STRESS COMPONENTS ONLY.
C
      REWIND 1
      REWIND 3
C
      READ (5,1000) KKK,ISP
      WRITE (6,3000)
      DO 500 N=1,NELTYP
      READ (1) NPAR
      WRITE (3) NPAR
      WRITE (6,2001) NPAR(1)
      READ (5,1000) NEL,IS
      WRITE (6,2000) NEL,IS
      NUME=NPAR(2)
      L=0
      NUM(N)=0
C
      DO 400 M=1,NUME
      READ (1) NS,ND,LM,SA
      IF(NEL.NE.M) GO TO 400
C
      DO 300 I=1,NS
      II=IS(I)
      IF(II.EQ.0) GO TO 350
      L=L+1
      KS(1,L)=NEL
      KS(2,L)=II
C
      DO 200 K=1,NF
      SS=0.
      DO 150 J=1,ND
      JJ=LM(J)
      IF (JJ.GT.NEQ) GO TO 150
      IF(JJ) 150,150,140
  140 SS = SS + SA(II,J)*FI(JJ,K)
  150 CONTINUE
  200 SF(L,K)=SS
C
      IF(L.LT.8) GO TO 300
      WRITE (3) L,KS,SF
      L=0
      NUM(N)=NUM(N) + 1
  300 CONTINUE
  350 READ  (5,1000) NEL,IS
      WRITE (6,2000) NEL,IS
  400 CONTINUE
C
      IF(L.EQ.0) GO TO 500
      WRITE (3) L,KS,SF
      NUM(N)=NUM(N) + 1
  500 CONTINUE
      WRITE (6,4000) KKK,ISP
```

```
HIST 836  C
HIST 837  C       COMPUTE AND OUTPUT HISTORY OF VALUES
HIST 838  C
HIST 839          CALL DISPLY (X,SF,NF,NDS,NUM,NELTYP,KKK,1,ISP)
HIST 840  C
HIST 841          RETURN
HIST 842  C
HIST 843   1000 FORMAT (1315)
HIST 844   2000 FORMAT (I6,4X,12I3)
HIST 845   3000 FORMAT (28H1STRESS COMPONENTS FOR WHICH /
HIST 846       . 25H TIME HISTORY IS REQUIRED )
HIST 847   3001 FORMAT (/16H ELEMENT TYPE .. ,I2//
HIST 848       . 38H ELEMENT      DESIRED STRESS COMPONENTS )
HIST 849   4000 FORMAT (/16H OUTPUT TYPE.....I1/
HIST 850       .              16H PLOT SPACING....I1)
HIST 851  C
HIST 852          END



HIST 853          SUBROUTINE PLOT (IT,JT,NDS,ISP)
HIST 854  C
HIST 855          COMMON /  EM  / PP(101),KD(3,8),XM(8),TM(8),IP(8),X(8)
HIST 856          DIMENSION SM(8)
HIST 857          DATA SM / 1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8 /
HIST 858          DATA BL,V /1H ,1HX /
HIST 859          COMMON / DYN / NT,NDT,DAMP,DT
HIST 860  C
HIST 861  C
HIST 862          READ (IT) KD,XM,TM,L
HIST 863          WRITE (6,3000) (KD(1,I),KD(2,I),XM(I),TM(I),I,I=1,L)
HIST 864  C
HIST 865          DO 100 I=1,L
HIST 866          IF(XM(I)) 50,100,50
HIST 867     50 XM(I)=50./XM(I)
HIST 868    100 CONTINUE
HIST 869          TT=0.
HIST 870          WRITE (6,999)
HIST 871          WRITE (6,1000)
HIST 872          WRITE (6,2000) TT,(V,I=1,101),TT
HIST 873  C
HIST 874          K=1
HIST 875          DO 200 I=2,100
HIST 876    200 PP(I)=BL
HIST 877  C
HIST 878          DO 500 N=1,NDS
HIST 879          READ (JT) X
HIST 880          PP(1)=V
HIST 881          PP(51)=V
HIST 882          PP(101)=V
HIST 883  C
HIST 884    220 II=ISP
HIST 885    210 IF(II.LE.0) GO TO 250
HIST 886          WRITE (6,2001) PP
HIST 887          II=II-1
HIST 888          GO TO 210
HIST 889  C
HIST 890    250 TT=TT+DT
HIST 891          DO 300 I=1,L
HIST 892          XX=XM(I)*X(I)
HIST 893          M=XX
```

```
HIST 894          M=M+51
HIST 895          IP(I)=M
HIST 896    300 PP(M)=SM(I)
HIST 897          IF(K.LT.10)  GO TO 320
HIST 898          K=1
HIST 899          WRITE (6,2000) TT,PP,TT
HIST 900          GO TO 340
HIST 901    320 WRITE (6,2001) PP
HIST 902          K=K+1
HIST 903  C
HIST 904  C       RESET PP
HIST 905  C
HIST 906    340 DO 360 I=1,L
HIST 907          M=IP(I)
HIST 908    360 PP(M)=BL
HIST 909    500 CONTINUE
HIST 910          TT=TT+DT
HIST 911          WRITE (6,2000) TT,(V,I=1,101),TT
HIST 912          WRITE (6,1000)
HIST 913  C
HIST 914          RETURN
HIST 915  C
HIST 916  C       CARRIAGE CONTROL CHARACTER Z SUPPRESSES
HIST 917  C       SKIP OVER PAGE FOLDS
HIST 918  C
HIST 919    999 FORMAT (//8X,8HORDINATE)
HIST 920   1000 FORMAT (/12HZ  TIME -1.0,21X,4H 0.5,22X,3H0.0,22X,3H0.5,22X,
HIST 921       . 3H1.0, X,4HTIME)
HIST 922   2000 FORMAT (1HZ,F7.2,2X,101A1,F7.2)
HIST 923   2001 FORMAT (1HZ,9X,101A1)
HIST 924   3000 FORMAT (I6,I12,1PE18.3,E12.3,I6)
HIST 925          END
```

```
      SUBROUTINE EMID(ID,MASS,NUMNP,NEQB,NEQ)
      DIMENSION ID(NUMNP,6),MASS(NEQB)
C
      REWIND 3
      REWIND 8
      READ (8) ID
      L=1
      DO 200 N=1,NUMNP
      DO 100 I=1,6
   50 MASS(L)=0
      II=ID(N,I)
      IF (((II.LE.0).OR.(II.GT.NEQ)) GO TO 100
      IF(L.LE.NEQB) GO TO 75
      WRITE (3) MASS
      L=1
   75 IF(I.GT.3) GO TO 90
      MASS(L)=I
   90 L=L+1
  100 CONTINUE
  200 CONTINUE
      DO 300 I=L,NEQB
  300 MASS(I)=0
      WRITE (3) MASS
C
      RETURN
C
      END
```

```
      SUBROUTINE RESPEC
C
C
      COMMON / MISC / NBLOCK,NEQB,LL,NF,LB,NSVV
      COMMON /JUNK/ XXX(4),NDYN,JUK(200)
      COMMON /ELPAR/ NPAR(14),NUMNP,MBAND,NELTYP,N1,N2,N3,N4,N5,MTOT,NEQ
      DIMENSION T(6)
      COMMON A(1)
C
C
      CALL SECOND (T(1))
C
      IF (NSVV.LT.10) NSVV=10
      CALL DYNAM (NEQ,MBAND,NBLOCK,NEQB,NF,NSVV,MTOT)
C
      CALL SECOND (T(2))
C
       N1=1
      N2=N1 + 6*NUMNP
      CALL EMID (A(N1),A(N2),NUMNP,NECB,NEQ)
      N2=N1+NECB*NF
      N3=N2+NF*3
      N4=N3+NEQP
      N5=N4+NF
      N6=N5+NECB
      N7=N6+NF
      IF(N7.GT.MTOT) CALL ERROR(N7-MTOT)
      CALL SPECTRM (A(N1), A(N2),A(N3),A(N4),A(N5),NEQB,NF,NBLOCK,A(N6))
C
C     MODE SHAPE NG IS R.M.S. DISPLACEMENT
C
      CALL SECOND (T(3))
      N2=N1+6*NUMNP
      NG=NF+1
      N3=N2+6*NG
      N4=N3+NEQB*NG
      MM=N4-MTOT
      IF (MM.GT.0) CALL ERROR (MM)
      NT=2
      CALL PRINTD(A(N1),A(N2),A(N3),NECB,NUMNP,NG,NBLOCK,NEC,NT,NF)
C
C     COMPUTE STRESSES
C
      CALL SECOND (T(4))
      N2=N1+4*LL
      N3=N2+NECB*LL
      LB=(MTOT-N3)/(NEQ +12)
      LL=NF
      NDYN=2
      CALL STRESS (A(N1),A(N2),A(N3),NEQB,LB,LL,NEQ,NBLOCK)
C
C     R M S COMBINATION OF STRESSES
C
      CALL SECOND (T(5))
      CALL RMS
      CALL SECOND (T(6))
C
      TT=0.
      DO 100 I=1,5
      T(I)=T(I+1)-T(I)
  100 TT=TT + T(I)
```

```
RESP  62          T(6)=TT
RESP  63          WRITE (6,1000) T
RESP  64    C
RESP  65          RETURN
RESP  66    C
RESP  67     1000 FORMAT (23H1....TIME LOG (SECONDS)  ///
RESP  68         .  33H  MODE SHAPES AND FREQUENCIES...  ,F8.2//
RESP  69         .  33H  MAXIMUM MODAL DISPLACEMENTS...  ,F8.2//
RESP  70         .  33H  PRINT MODE SHAPES..............  ,F8.2//
RESP  71         .  33H  COMPUT MODAL STRESSES..........  ,F8.2//
RESP  72         .  33H  R.M.S. COMBINATION OF STRESSES  ,F8.2//
RESP  73         .  33H  TOTAL FOR SPECTRUM ANALYSIS...  ,F8.2//)
RESP  74          END


RESP  75          SUBROUTINE SPECFRM (F,PX,XM,W,MASS,NEQB,NF,NBLOCK,TM)
RESP  76          DIMENSION PX(NF,3),F(NEQB,NF),XM(NEQB),W(NF),MASS(NEQB)
RESP  77          DIMENSION DIRN(3),TM(NF)
RESP  78          DATA DIRN /3H X ,3H Y ,3H Z /
RESP  79    C
RESP  80    C     COMPUTES MODAL AND R.M.S. DISPL RESPONSE TO EARTHQUAKE
RESP  81    C
RESP  82          TPI=6.2831853
RESP  83          DO 100 I=1,NF
RESP  84          TM(I)=0.
RESP  85          DO 100 J=1,3
RESP  86      100 PX(I,J)=0.
RESP  87    C
RESP  88    C      FORM MODAL PARTICIPATION FACTORS PX(I,IDRN)
RESP  89    C           IDRN=1,2,3 .... FOR X,Y,Z, DIRN EARTHQUAKE
RESP  90    C
RESP  91          REWIND 9
RESP  92          REWIND 3
RESP  93          REWIND 7
RESP  94          READ (7)
RESP  95          DO 5 I=1,NBLOCK
RESP  96        5 READ (7)
RESP  97          DO 200 N=1,NBLOCK
RESP  98          BACKSPACE 7
RESP  99          READ (7) F
RESP 100          BACKSPACE 7
RESP 101          READ (3) MASS
RESP 102          READ (9) XM
RESP 103    C
RESP 104          DO 250 I=1,NEQB
RESP 105          J=MASS(I)
RESP 106          IF (J.LE.0) GO TO 250
RESP 107          DO 240 L=1,NF
RESP 108          TM(L)=TM(L) + F(I,L)*F(I,L)*XM(I)
RESP 109      240 PX(L,J)=PX(L,J)+F(I,L)*XM(I)
RESP 110      250 CONTINUE
RESP 111      200 CONTINUE
RESP 112          CALL FRMAT (PX,NF,3,NF,2HPX)
RESP 113          CALL FRMAT (TM,NF,1,NF,2HTM)
RESP 114    C
RESP 115    C     READ FREQUENCIES W OFF TAPE 7
RESP 116    C
RESP 117          BACKSPACE 7
RESP 118          READ (7) W
RESP 119          REWIND 2
RESP 120          WRITE (2) W
RESP 121    C
RESP 122    C     COMPUTE MODAL AMPLITUDES (IN W) FROM SPECTRUM AND PX
RESP 123    C
RESP 124          READ (5,1000) IDRN
RESP 125          WRITE (6,2000) DIRN(IDRN)
RESP 126          DO 300 I=1,NF
RESP 127          WW=TPI/W(I)
RESP 128      300 W(I)=SD(WW)*PX(I,IDRN)/TM(I)
RESP 129          CALL PRMAT (W,NF,1,NF,3HAMP )
RESP 130    C
RESP 131    C     WRITE MODAL DISPLS F AND R.M.S. ON TAPE 2
RESP 132    C
RESP 133          REWIND 7
RESP 134          READ (7)
RESP 135          DO 350 N=1,NBLOCK
RESP 136          READ (7) F
RESP 137          DO 310 J=1,NF
RESP 138          AMP=W(J)
RESP 139          DO 310 I=1,NEQB
RESP 140      310 F(I,J)=F(I,J)*AMP
RESP 141    C
RESP 142          DO 320 I=1,NEQB
RESP 143          WW=0.
RESP 144          DO 330 J=1,NF
RESP 145      330 WW=WW+F(I,J)**2
RESP 146      320 XM(I)=SQRT(WW)
RESP 147      350 WRITE (2) F,XM
RESP 148    C
RESP 149          RETURN
RESP 150     1000 FORMAT (I5)
RESP 151     2000 FORMAT (28H1 RESPONSE SPECTRUM ANALYSIS  //
RESP 152         .  13H  CONSIDERING ,A3,9HDIRECTION  )
RESP 153          END


RESP 154          FUNCTION   SD(TT)
RESP 155    C
RESP 156          COMMON / JUNK / MM,L,K,NTAG,NDVN,T(90),S(90),HED(12),I,W,SS,S
RESP 157          IF (NTAG.EQ.1) GO TO 500
RESP 158          NTAG=1
RESP 159    C
RESP 160    C     READ ACCELERATION SPECTRUM
RESP 161    C
RESP 162          READ  (5,1000) HED,NPTS,SFTP,(T(I),S(I),I=1,NPTS)
RESP 163          WRITE (6,2000) HED,NPTS,SFTP,(T(I),S(I),I=1,NPTS)
RESP 164          TPI=8.*ATAN(1.)
RESP 165      500 CONTINUE
RESP 166    C
RESP 167          DO 600 I=1,NPTS
RESP 168          IF (TT.LT.T(I)) GO TO 700
RESP 169      600 CONTINUE
RESP 170      700 TI=T(I)-T(I-1)
RESP 171          SI=S(I)-S(I-1)
RESP 172          SS=S(I-1)+SI*(TT-T(I-1))/TI
RESP 173          W=TPI/TT
RESP 174          SD=SFTP*SS/W**2
RESP 175    C
RESP 176     1000 FORMAT (12A6/I5,F10.2/(2F10.2))
RESP 177     2000 FORMAT (1H0,12A6//I6,F15.3//(1P2E14.2))
```

```
RESP 178  C
RESP 179        RETURN
RESP 180        END



RESP 181        SUBROUTINE RMS
RESP 182  C
RESP 183  C     COMPUTE AND PRINT RMS STRESSES
RESP 184  C
RESP 185        COMMON / MISC / NBLOCK,NEQB,LL,NF,LB
RESP 186        COMMON /ELPAR/ NPAR(14),NUMNP,MBAND,NELTYP,N1,N2,N3,N4,N5,MTOT,NEQ
RESP 187        COMMON / JUNK / N,I,K,NTAG,NDYN,S(12),SIG(12)
RESP 188  C
RESP 189        I1=1
RESP 190        I2=2
RESP 191        REWIND 3
RESP 192  C
RESP 193        DO 700 MM=1,NELTYP
RESP 194        WRITE (6,1000)
RESP 195        REWIND I1
RESP 196        REWIND I2
RESP 197        READ (3) NPAR
RESP 198        MTYPE=NPAR(1)
RESP 199        NUMEL=NPAR(2)
RESP 200        NPAR(1)=0
RESP 201        NTAG=0
RESP 202  C
RESP 203        DO 500 L=1,NF,LB
RESP 204        DO 400 N=1,NUMEL
RESP 205        DO 20 I=1,12
RESP 206     20 S(I)=0.
RESP 207        IF (L.NE.1) READ (I2) S
RESP 208        KK=NF-L-LB
RESP 209        IF (KK.LT.0) LB=KK+1+LB
RESP 210        DO 100 K=1,LB
RESP 211        READ (3) NS,SIG
RESP 212        DO 50 I=1,NS
RESP 213     50 S(I)=S(I)+SIG(I)**2
RESP 214    100 CONTINUE
RESP 215  C
RESP 216        IF (KK.GE.0) GO TO 350
RESP 217  C
RESP 218  C     PRINT RMS STRESSES
RESP 219  C
RESP 220        DO 60 I=1,NS
RESP 221     60 S(I)=SQRT(S(I))
RESP 222        I=0
RESP 223        GO TO (1,2,3,4,5) MTYPE
RESP 224  C
RESP 225      1 CALL TRUSS
RESP 226        GO TO 400
RESP 227  C
RESP 228      2 CALL BEAM
RESP 229        GO TO 400
RESP 230  C
RESP 231      3 CALL CBEAM
RESP 232        GO TO 400
RESP 233  C
RESP 234      4 CALL BOUND
RESP 235        GO TO 400
RESP 236  C
RESP 237      5 CALL EXPJP
RESP 238        GO TO 400
RESP 239  C
RESP 240    350 WRITE (I1) S
RESP 241  C
RESP 242    400 CONTINUE
RESP 243        REWIND I1
RESP 244        REWIND I2
RESP 245        I3=I1
RESP 246        I1=I2
RESP 247    500 I2=I3
RESP 248  C
RESP 249    700 CONTINUE
RESP 250  C
RESP 251        RETURN
RESP 252   1000 FORMAT (28H1 RESPONSE SPECTRUM ANALYSIS  //
RESP 253       .  27H ROOT MEAN SQUARE STRESSES  )
RESP 254        END
```

```
      SUBROUTINE TRUSS
C
      COMMON A(1)
      COMMON /ELPAR/ NPAR(14),NUMNP,MBAND,NELTYP,N1,N2,N3,N4,N5,MTOT,NEQ
      COMMON / JUNK / MM,L,K,NTAG,NDYN,SIG(12),EXPA(188)
C
      IF(NPAR(1).EQ.0) GO TO 500
      N6=N5+NPAR(3)
      N7=N6+NPAR(3)
      N8=N7+NPAR(3)
      N9=N8+NPAR(3)
      MM=N9+NPAR(3)-MTOT
      IF(MM.GT.0) CALL ERROR(MM)
      CALL RUSS(A(N1),A(N2),A(N3),A(N4),A(N5),A(N6),A(N7),A(N8),A(N9),
     1 NUMNP)
      RETURN
C
  500 IF (NTAG.EQ.0) WRITE (6,2002)
  352 WRITE(6,3002) MM,L,SIG(1),SIG(2)
      NTAG=1
C
      RETURN
 2002 FORMAT (//23H   TRUSS MEMBER ACTIONS //
     *            46H0 MEMBER     LOAD           STRESS            FORCE  )
 3002 FORMAT (2I8,F15.5,F15.3)
      END


      SUBROUTINE RUSS(ID,X,Y,Z,E,THERM,DEN,AREA,WT,NUMNP)
C
      DIMENSION X(1),Y(1),Z(1),ID(NUMNP,1),E(1),THERM(1),DEN(1),AREA(1)
     1 ,WT(1)
      COMMON /ELPAR/ NPAR(14),NNNNN,MBAND,NELTYP,N1,N2,N3,N4,N5,MTOT,NEQ
      COMMON /EM/LM(24),ND,NS,S(24,24),P(24,4),XM(24),ST(12,24),TT(12,4)
      COMMON /JUNK/ EMUL(4,4),I,J,K,L,M,N,II,JJ,KK,MTYPE,TEMP,DX,DY,DZ,
     1 XL2,XL,XX,YY,F,FT,FX,FY,FZ,MIN,MAX,NDIF,KKK,TEM,MTYP
C
C     CONTROL INFORMATION AND MEMBER PROPERTIES
C
      NUME=NPAR(2)
      NUMMAT=NPAR(3)
      WRITE (6,2000) NUME,NUMMAT
      WRITE (6,2001)
      DO 10 I=1,NUMMAT
      READ  (5,1001) N,E(N),THERM(N),DEN(N),AREA(N),WT(N)
   10 WRITE (6,2002) N,E(N),THERM(N),DEN(N),AREA(N),WT(N)
C
C     ELEMENT LOAD MULTIPLIERS
C
      READ (5,1003) EMUL
      WRITE (6,2003) EMUL
C
C     ELEMENT INFORMATION
      WRITE (6,2005)
C
      N=1
  100 READ (5,1004)  M,II,JJ,MTYP,TEM,KK
      IF(KK.EQ.0) KK=1
  120 IF(M.NE.N) GO TO 200
      I=II
      J=JJ
      MTYPE=MTYP
      TEMP=TEM
      KKK=KK
C
C     1. FORM ELEMENT STIFFNESS AND STRESS MATRICES
C
  200 DX=X(I) X(J)
      DY=Y(I)-Y(J)
      DZ=Z(I)-Z(J)
      XL2=DX*DX+DY*DY+DZ*DZ
      XL=SQRT(XL2)
      XX=E(MTYPE)*AREA(MTYPE)*XL
      ST(1,1)=DX/XL2
      ST(1,2)=DY/XL2
      ST(1,3)=DZ/XL2
      ST(1,4)=-ST(1,1)
      ST(1,5)=-ST(1,2)
      ST(1,6)=-ST(1,3)
C
      DO 300 L=1,6
      YY=ST(1,L)*XX
      DO 250 K=L,6
      S(K,L)=ST(1,K)*YY
  250 S(L,K)=S(K,L)
      ST(1,L)=E(MTYPE)*ST(1,L)
  300 ST(2,L)=AREA(MTYPE)*ST(1,L)
C
C     2. INERTIA AND THERMAL LOADS
C
      F=WT(MTYPE)*XL/2.
      FT=TEMP*THERM(MTYPE)*E(MTYPE)*AREA(MTYPE)
      FX=DX*FT/XL
      FY=DY*FT/XL
      FZ=DZ*FT/XL
C
      DO 350 L=1,4
      TT(2,L)=EMUL(L,4)*FT
      TT(1,L)=TT(2,L)/AREA(MTYPE)
      P(1,L)=EMUL(L,1)*F-EMUL(L,4)*FX
      P(2,L)=EMUL(L,2)*F-EMUL(L,4)*FY
      P(3,L)=EMUL(L,3)*F-EMUL(L,4)*FZ
      P(4,L)=EMUL(L,1)*F+EMUL(L,4)*FX
      P(5,L)=EMUL(L,2)*F+EMUL(L,4)*FY
  350 P(6,L)=EMUL(L,3)*F+EMUL(L,4)*FZ
      F=DEN(MTYPE)*AREA(MTYPE)*XL/2.
      DO 375 L=1,6
  375 XM(L)=F
C
C     3. FORM LOCATION MATRIX AND COMPUTE BAND WIDTH
C
      DO 400 L=1,3
      LM(L)=ID(I,L)
  400 LM(L+3)=ID(J,L)
C
      ND=6
      NS=2
      CALL WRITET(MBAND,NDIF)
C
C     4. CHECK FOR MORE ELEMENTS
C
```

```
      WRITE (6,2004) N,I,J,MTYPE,TEMP,NDIF
      IF (N.EQ.NUME) RETURN
      N=N+1
      I=I+KKK
      J=J+KKK
      IF(N.GT.M) GC TO 100
      GC TC 120
C
 1001 FORMAT (I5,5F10.0)
 1003 FCRMAT (4F10.C)
 1004 FORMAT (4I5,IF10.0,I5)
 2000 FORMAT (///25H1NUMBER OF TRUSS MEMBERS= I5/
     1 25H NUMBER CF DIFF. MEMBERS= I5)
 2001 FORMAT (///1X,4HTYPE,14X,1HE,10X,5HALPHA,12X,3HDEN,11X,4HAREA
     1 11X,4HWT/L )
 2002 FORMAT (I5,5E15.7)
 2003 FORMAT(///25H ELEMENT LOAD MULTIPLIERS / 20X,1HA,14X,1HB,14X,1HC,
     1 14X,1HD,/6H X-DIR4E15.6/ 6H Y-DIR4E15.6/ 6H Z-DIR4E15.6/
     2 6H  TEMP4E15.6)
 2004 FORMAT (4I6,F10.2,I7)
 2005 FORMAT (///42H1    N     I     J  TYPE      TEMP   BAND  )
      END
```

```
      SUBROUTINE BEAM
C
      COMMON A(1)
      COMMCN /ELPAR/ NPAR(14),NUMNP,MBAND,NELTYP,N1,N2,N3,N4,N5,MTOT,NEQ
      COMMON /  JUNK / MM,L,K,NTAG,NDYN,SIG(12),EXRA(188)
C
      IF(NPAR(1).EQ.0) GO TO 500
      N6=N5+NPAR(5)
      N7=N6+NPAR(5)
      N8=N7+NPAR(5)
      N9=N8+12*NPAR(4)
      N10=N9+6*NPAR(3)
      IF(N10.GT.MTOT) CALL ERROR(N10-MTOT)
      CALL TEAM(NPAR(2),NPAR(3),NPAR(4),NPAR(5),A(N1),A(N2),A(N3),
     .          A(N4),A(N5),A(N6),A(N7),A(N8),A(N9),NUMNP,MBAND)
      RETURN
C
  500 IF(NTAG.EQ.0) WRITE (6,2002)
      WRITE(6,3002) MM,L,(SIG(I),I=1,12)
      NTAG=1
      RETURN
 2002 FORMAT(/29H0.....BEAM FORCES AND MOMENTS//
     . 10HOBEAM LOAD 5X 5HAXIAL  2(7X,5HSHEAR),5X 7HTORSION
     .  2(5X,7HBENDING)/ 10H  NO.  NO.  8X  2HR1  10X  2HR2  10X
     .  2HR3 10X 2HM1 10X 2HM2 10X 2HM3)
 3002 FORMAT (I5,I4,1PE11.3,5E12.3/8X,6E12.3/)
      END
```

```
      SUBROUTINE TEAM(NBEAM,NUMETP,NUMFIX,NUMMAT,ID,X,Y,Z,E,G,RC,
     .SFT, COPROP,NUMNP,MBAND)
C
C     FORMS 3-D BEAM STIFFNESS AND STRESS ARRAYS
C
      COMMON/EM/LM(24),ND,NS,ASA(24,24),RF(24,4),XM(24),SA(12,24),
     .    SF(12,4)
      DIMENSION X(1),Y(1),Z(1),ID(NUMNP,1),E(1),G(1),SFT(NUMFIX,1)
     .  ,COPROP(NUMETP,1),RO(1),EMUL(3,4)
      COMMON /NEWB/ LC(4),T(3,3),JK(6),MELTYP,DL,MATTYP
      DIMENSION ILC(4),TI(3,3),TJ(3,3),STIF(722),TS(2,2),LS(4)
      EQUIVALENCE (STIF,LM)
C
C     INITIALIZATION
C
      WRITE (6,2005) NBEAM,NUMETP,NUMFIX,NUMMAT
      N=0
      DO 5 I=1,1056
    5 STIF(I)=0.
C
C     READ AND PRINT MATERIAL PROPERTY DATA
C
      WRITE (6,2001)
      DO 10 I=1,NUMMAT
      READ  (5,1001) N,E(N),G(N),RO(N)
      WRITE (6,2002) N,E(N),G(N),RO(N)
   10 G(N)=0.5*E(N)/(1.+G(N))
C
C     READ AND PRINT GEOMETRIC PROPERTIES OF COMMON ELEMENTS.
C
```

```
      WRITE (6,2003)
      DO 30 I=1,NUMETP
      READ  (5,1002) N,(COPROP(N,J),J=1,6)
      IF((COPROP(N,1).NE.0.0).AND.(COPROP(N,4).NE.0.0).AND.
     1   (COPROP(N,5).NE.0.0).AND.(COPROP(N,6).NE.0.0))  GO TO 20
      WRITE (6,2013)
      CALL EXIT
   20 WRITE (6,2004) N,(COPROP(N,J),J=1,6)
   30 CONTINUE
C
C     ELEMENT LOAD MULTIPLIERS
C
      READ  (5,1006) ((EMUL(I,J),J=1,4),I=1,3)
      WRITE (6,2006) ((EMUL(I,J),J=1,4),I=1,3)
C
C     READ AND PRINT FIXED END FORCES IN LOCAL COORDINATES
C
      IF(NUMFIX .EQ. 0) GO TO 56
      WRITE (6,2010)
      DO 55 I=1,NUMFIX
      READ  (5,1005) N,(SET(N,J),J=1,12)
   55 WRITE (6,2011) N,(SET(N,J),J=1,12)
   56 CONTINUE
C
C     READ AND PRINT ELEMENT DATA. GENERATE MISSING INPUT.
C
      WRITE (6,4000)
      L=0
   60 KKK=0
      READ (5,3000) INEL,INI,INJ,INK,IMAT,IMEL,ILC,INELKI,INELKJ,INC
      IF (INEL.NE.1) GO TO 15
      NI=INI
      NJ=INJ
      NK=INK
   15 IF (INC.LE.0) INC=1
   65 L=L+1
      KKK=KKK+1
      ML=INEL-L
      IF (ML)  66,67,68
   66 WRITE (6,4003) INEL
      CALL EXIT
   67 NEL=INEL
      NI=INI
      NJ=INJ
      NK=INK
      MATTYP=IMAT
      MELTYP=IMEL
      DO 90 I=1,4
   90 LC(I)=ILC(I)
      NEKODI=INELKI
      NEKODJ=INELKJ
      DO 91 I=1,3
   91 T(2,I)=TI(2,I)
      GO TO 69
   68 NEL=INEL--ML
      NI =IN+KKK*INCK
      NJ =JN+KKK*INCP
   69 CONTINUE
      WRITE (6,4001) NEL,NI,NJ,NK,MATTYP,MELTYP,LC,NEKODI,NEKODJ
C
   74 DX=X(NJ)-X(NI)
      DY=Y(NJ)-Y(NI)
      DZ=Z(NJ)-Z(NI)
      DL=SQRT(DX*DX+DY*DY+DZ*DZ)
      IF(DL) 75,75,76
   75 WRITE (6,4005) NEL
      CALL EXIT
C
C     FORM GLOBAL TO LOCAL COORDINATE TRANSFORMATION.
C
   76 T(1,1)=DX/DL
      T(1,2)=DY/DL
      T(1,3)=DZ/DL
C
C     COMPUTE DIRECTION COSINES OF LOCAL Y-AXIS
C
      A1=X(NJ)-X(NI)
      A2=Y(NJ) Y(NI)
      A3=Z(NJ)-Z(NI)
      B1=X(NK)-X(NI)
      B2=Y(NK)-Y(NI)
      B3=Z(NK)-Z(NI)
      AA=A1*A1+A2*A2+A3*A3
      AB=A1*B1+A2*B2+A3*B3
      U1=AA*B1-AB*A1
      U2=AA*B2 AB*A2
      U3=AA*B3-AB*A3
      UU=U1*U1+U2*U2+U3*U3
      UU=SQRT(UU)
      IF (UU.GT.0.) GO TO 40
      WRITE (6,4002) INEL
      STOP
   40 T(2,1)=U1/UU
      T(2,2)=U2/UU
      T(2,3)=U3/UU
      T(3,1)=T(1,2)*T(2,3)-T(1,3)*T(2,2)
      T(3,2)=T(1,3)*T(2,1)-T(1,1)*T(2,3)
      T(3,3)=T(1,1)*T(2,2)-T(1,2)*T(2,1)
C
C     CHECK IF NEW STIFFNESS NEEDED
C
      IF (NEL.GE.1) GO TO 80
      IF  (ABS(DS-DL).GT.DL/100.)  GO TO 80
      IF ((MT.NE.MATTYP).OR.(ME.NE.MELTYP)) GO TO 80
      IF ((JK(1).NE.NEKODI).OR.(JK(2).NE.NEKODJ)) GO TO 80
      DO 81 I=1,4
      IF (LS(I).NE.LC(I)) GO TO 80
   81 CONTINUE
      DO 82 I=1,2
      DO 82 J=1,2
      IF (ABS(TS(I,J)-T(I,J)).GT.ABS(T(I,J)/100.)) GO TO 80
   82 CONTINUE
      GO TO 185
C
   80 DS=DL
      MT=MATTYP
      ME=MELTYP
      DO 77 I=1,2
      DO 77 J=1,2
   77 TS(I,J)=T(I,J)
      DO 78 I=1,4
   78 LS(I)=LC(I)
      JK(1)=NEKODI
      JK(2)=NEKODJ
```

```
C
C     FORM NEW STIFFNESS
C
      CALL NEWBM (E,G,RO,COPROP,SFT,NUMFIX,NUMETP)
C
C     ADD GRAVITY LOADING ... POINT LOADS ONLY COMPUTED
C
      DO 180 I=1,3
      DO 180 J=1,4
      RF(I,J)=RF(I,J)+EMUL(I,J)*XM(I)
  180 RF(I+6,J)=RF(I+6,J)+EMUL(I,J)*XP(I+6)
C
C     FORM ELEMENT LOCATION MATRIX
C
  185 CONTINUE
      DO 170 M=1,6
      LM(M)=ID(NI,M)
      LM(M+12)=0
      LM(M+18)=0
  170 LM(M+6)=ID(NJ,M)
C
      NS=12
      ND=12
C
C     TRANSFORM TO MASTER DEGREES OF FREEDOM
C
      CALL SLAVE (X,Y,Z,ID,NUMNP,NI,NJ)
C
C     WRITE ELEMENT INFORMATION ON TAPE
C
      CALL WRITET (MBAND,NDIF)
C
C     CHECK FOR LAST ELEMENT
C
      IF (NBEAM-NEL) 66,500,260
  260 IF (ML.GT.0)  GO TO 65
      IN=INI
      JN=INJ
      INCR=INC
      GO TO 60
  500 RETURN
C
 1001 FORMAT(I5,3F10.0)
 1002 FORMAT(I5,6F10.0)
 1005 FORMAT(I5,6F10.0/F15.0,5F10.0)
 1006 FORMAT (4F10.0)
 2001 FORMAT(55H1MATERIAL   YOUNG S     POISSON S            MASS
     1     /55H                 MODULUS       RATIO              DENSITY      )
 2002 FORMAT(1H ,I5,3X,F12.0,F14.5,F14.5)
 2003 FORMAT(1H1/
     .  26H BEAM GEOMETRIC PROPERTIES//
     1   48H ELEMENT        AREA          AREA         AREA
     2  60H INERTIA      INERTIA      INERTIA
     3 / 48H  TYPE              X             Y            Z
     4                 30H      X             Y            Z              )
 2004 FORMAT(1H ,I5,2X,6F12.3)
 2005 FORMAT(39H1.......THREE DIMENSIONAL BEAM ELEMENTS//
     .  36H  NUMBER OF BEAMS                    =,I5/
     .  36H  NUMBER OF GEOMETRIC PROPERTY SETS=,I5/
     .  36H  NUMBER OF FIXED END FORCE SETS    =,I5/
     .  36H  NUMBER OF MATERIALS                =,I5)
 2006 FORMAT(///25H ELEMENT LOAD MULTIPLIERS / 20X,1HA,14X,1HB,14X,1HC,
     1 14X,1HD,/6H X-DIR4E15.6/ 6H Y-DIR4E15.6/ 6H Z-DIR4E15.6/ )
 2010 FORMAT(1H1,)  ,
     1  30X40H FIXED END FORCES IN LOCAL COORDINATES
     2//53H TYPE   NODE          FORCE X      FORCE Y      FORCE Z
     3                       35H MOMENT X    MOMENT Y    MOMENT Z           )
 2011 FORMAT(1H ,I3,6X,1HI,3X,6F12.3/1H ,9X,1HJ,3X,6F12.3/)
 2013 FORMAT(1H0/
     1  60H SECTION PROPERTIES OTHER THAN SHEAR AREAS MAY NOT BE SPECIF
     2  34HIED AS ZERO. EXECUTION TERMINATED.)
 3000 FORMAT (1015,2I6,I8)
 4000 FORMAT(1H1/
    .. 5H0BEAM 5X 5HNODES 5X 5H MATL 5H GEOM 5X 10HELEM LOADS 4X   10X
     . 12H  END CODES / 5H  NO      4X 1HI 4X 1HJ 4X 1HK 5H  NO  5H  NO
     .  4X 1HA 4X 1HB 4X 1HC 4X 1HD 9X 1HI 9X 1HJ)
 4001 FORMAT (1015,2I10)
 4002 FORMAT (9H0BEAM NO ,I5, 26H     K NODE ON BEAM X-AXIS     ,
     .  26H......EXECUTION TERMINATED )
 4003 FORMAT(36H0ELEMENT CARD ERROR, ELEMENT NUMBER= I6)
 4004 FORMAT(1H ,31HNODAL POINT NUMBERS FOR ELEMENT,I5,36HARE IDENTICAL.
     1 EXECUTION TERMINATED.)
 4005 FORMAT(8H0ELEMENT,I5,39H HAS ZERO LENGTH. EXECUTION TERMINATED.)
      END


      SUBROUTINE NEWBM(E,G,RO,COPROP,SFT,NUMFIX,NUMETP)
C
C     FORM NEW BEAM STIFFNESS
C
      DIMENSION E(1),G(1),RO(1),COPROP(NUMETP,1),SFT(NUMFIX,1)
      COMMON/EM/LM(24),ND,NS,ASA(24,24),RF(24,4),XP(24),SA(12,24),
     .     SF(12,4)
      COMMON /NEWB/ LC(4),T(3,3),JK(6),MELTYP,DL,MATTYP
      DIMENSION R(12),S(12,12),C(12)
C
      DO 5 I=1,144
    5 S(I)=0.
      AX =COPROP(MELTYP,1)
      AY =COPROP(MELTYP,2)
      AZ =COPROP(MELTYP,3)
      AAX=COPROP(MELTYP,4)
      AAY=COPROP(MELTYP,5)
      AAZ=COPROP(MELTYP,6)
      SHFY=0.0
      SHFZ=0.0
      ZY=E(MATTYP)/(DL*DL)
      EIY=ZY*AAY
      EIZ=ZY*AAZ
      IF(AY.NE.0.0) SHFY=6.*EIZ/(G(MATTYP)*AY)
      IF(AZ.NE.0.0) SHFZ=6.*EIY/(G(MATTYP)*AZ)
      COMMY=EIY/(1.+2.*SHFZ)
      COMMZ=EIZ/(1.+2.*SHFY)
C
C     FIXED END FORCES IN LOCAL COORDS
C
      DO 73 N=1,4
      M=LC(N)
      IF (M.GT.0)  GO TO 71
      DO 70 I=1,12
   70 SF(I,N)=0.
      GO TO 73
```

```
   71 DO 72 I=1,12
   72 SF(I,N)=SFT(M,I)
   73 CONTINUE
C
C     FORM ELEMENT STIFFNESS IN LOCAL COORDINATES
C
      S(1,1)= E(MATTYP)* AX/UL
      S(4,4)= G(MATTYP)*AAX/UL
      S(2,2)= CCMMZ*12./DL
      S(3,3)= CCMMY*12./DL
      S(5,5)= CCMMY* 4.*DL*(1.+0.5*SHFZ)
      S(6,6)= CCMMZ* 4.*UL*(1.+0.5*SHFY)
      S(2,6)= CCMMZ* 6.
      S(3,5)=-CCMMY* 6.
      DO 102 I=1,6
      J=I+6
  102 S(J,J)=S(I,I)
      DO 104 I=1,4
      J=I+6
  104 S(I,J)= S(I,I)
      S(6,12)= S(6,6)*(1.-SHFY)/(2.+SHFY)
      S(5,11)= S(5,5)*(1.-SHFZ)/(2.+SHFZ)
      S(2,12)= S(2,6)
      S(6, 8)=-S(2,6)
      S(8,12)= S(2,6)
      S(3,11)= S(3,5)
      S(5, 9)= S(3,5)
      S(9,11)= S(3,5)
      DO 106 I=2,12
      K=I-1
      DO 106 J=1,K
  106 S(I,J)=S(J,I)
C
C     MODIFY ELEMENT STIFFNESS AND FORCE FOR ZERO END FORCE CONDITION
C
      IF ((JK(1)+JK(2)).EQ.0) GO TO 145
      DO 140 K=1,2
      KK=JK(K)
      KD=100000
      I1=6*(K-1)+1
      I2=I1+5
      DO 140 I=I1,I2
      IF (KK.LT.KD)  GO TO 140
      SII=S(I,I)
      DO 125 N=1,12
  125 R(N)=S(I,N)
      DO 130 M=1,12
      C(M)=S(M,I)/SII
      DO 130 N=1,12
  130 S(M,N)=S(M,N)-C(M)*R(N)
      DO 135 N=1,4
      SFI=SF(I,N)
      DO 135 M=1,12
  135 SF(M,N)=SF(M,N)-C(M)*SFI
  136 KK=KK-KD
  140 KD=KD/10
  145 CONTINUE
C
C     FORM LOCAL FORCES TO GLOBAL DISPLACEMENTS ARRAY
C
      DO 31 I=1,288
   31 SA(I)=0.
```

```
      DO 150 LA=1,10,3
      LB=LA+2
      DO 150 MA=1,10,3
      MB=MA-1
      DO 150 I=LA,LB
      DO 150 JM=1,3
      J=JM+MB
      XX=0.
      DO 151 K=1,3
  151 XX=XX+S(I,K+MB)*T(K,JM)
  150 SA(I,J)=XX
C
C     ELEMENT STIFFNESS AND FORCE IN GOLBAL COORDS
C
      DO 32 I=1,576
   32 ASA(I)=0.
      DO 160 LA=1,10,3
      LB=LA-1
      DO 160 MA=1,10,3
      MB=MA+2
      DO 160 IL=1,3
      I=IL+LB
      DO 160 J=MA,MB
      XX=0.
      DO 161 K=1,3
  161 XX=XX+T(K,IL)*SA(K+LB,J)
  160 ASA(I,J)=XX
C
      DO 165 LA=1,10,3
      LB=LA-1
      DO 165 IL=1,3
      I=IL+LB
      DO 165 N=1,4
      XX=0.
      DO 162 K=1,3
  162 XX=XX-T(K,IL)*SF(K+LB,N)
  165 PF(I,N)=XX
C
C     FORM MASS MATRIX
C
      XXM=PC(MATTYP)*AX*DL/2.
      DO 180 M=1,3
      XM(M)=XXM
      XM(M+3)=0.
      XM(M+9)=0.
  180 XM(M+6)=XXM
      RETURN
      END
```

```
      SUBROUTINE SLAVE (X,Y,Z,ID,NUMNP,NI,NJ)
C
C     PERFORMS SLAVE...MASTER DISPLACEMENT TRANSFORMATION
C
      DIMENSION X(1),Y(1),Z(1),ID(NUMNP,1)
      COMMON /EM/ LM(24),ND,NS,S(24,24),R(24,4),XM(24),SA(12,24),T(12,4)
C
C     DETERMINE REQUIRED TRANSLATION DEGREES OF FREEDOM
C
      DO 54 NF=1,12,6
      NOD=NI
      IF (NF.EQ.7) NOD=NJ
      DO 30 K=1,3
      I=K+NF-1
      IF (LM(I).GE.0) GO TO 30
      M=-LM(I)
      LM(I)=ID(M,K)
      IF(K-2) 35,45,55
   35 D1=-(Y(NOD)-Y(M))
      D2=  Z(NOD)-Z(M)
      LM(ND+1)=ID(M,6)
      LM(ND+2)=ID(M,5)
      GO TO 50
   45 D1=-(Z(NOD)-Z(M))
      D2=  X(NOD)-X(M)
      LM(ND+1)=ID(M,4)
      LM(ND+2)=ID(M,6)
      GO TO 50
   55 D1=-(X(NOD)-X(M))
      D2=  Y(NOD)-Y(M)
      LM(ND+1)=ID(M,5)
      LM(ND+2)=ID(M,4)
   50 CONTINUE
C
C     TRANSFORMATION...ARRAYS INCREASE IN SIZE
C
      DO 60 II=1,NC
      S(ND+1,II)=S(I,II)*D1
      S(ND+2,II)=S(I,II)*D2
      XM(ND+1)=XM(I)*D1*D1
      XM(ND+2)=XM(I)*D2*D2
      S(II,ND+1)=S(II,I)*D1
      S(II,ND+2)=S(II,I)*D2
      DO 56 J=1,4
      R(ND+1,J)=R(I,J)*D1
      R(ND+2,J)=R(I,J)*D2
   56 CONTINUE
   60 CONTINUE
C
      DO 70 II=1,NS
      SA(II,ND+1)=SA(II,I)*D1
   70 SA(II,ND+2)=SA(II,I)*D2
C
      S(ND+1,ND+1)=S(I,I)*D1**2
      S(ND+2,ND+2)=S(I,I)*D2**2
      S(ND+1,ND+2)=S(I,I)*D1*D2
      S(ND+2,ND+1)=S(ND+1,ND+2)
      ND=ND+2
   30 CONTINUE
C
C     SET ROTATIONS
C
      DO 54 J=1,3
      K=NF+J+2
      IF(LM(K).GE.0)  GO TO 54
      M=-LM(K)
      LM(K)=ID(M,J+3)
   54 CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE CBEAM
C
      COMMON A(1)
      COMMON /ELPAR/ NPAR(14),NUMNP,MBAND,NELTYP,N1,N2,N3,N4,N5,MTOT,NEQ
      COMMON / JUNK / MM,L,K,NTAG,NDYN,SIG(12),EXRA(188)
C
      IF(NPAR(1).EQ.0) GO TO 500
      N6=N5+NPAR(5)
      N7=N6+NPAR(5)
      N8=N7+NPAR(5)
      N9=N8+12*NPAR(4)
      N10=N9+6*NPAR(3)
      N11=N10+NPAR(6)
      IF (N11.GT.MTOT) CALL ERROR (N11-MTOT)
C
      CALL CTEAM (NPAR(2),NPAR(3),NPAR(4),NPAR(5),NPAR(6),
     .              A(N1),A(N2),A(N3),A(N4),A(N5),A(N6),A(N7),A(N8),
     .              A(N9),A(N10),NUMNP,MBAND)
      RETURN
C
  500 IF(NTAG.EQ.0) WRITE (6,2002)
      WRITE(6,3002) MM,L,(SIG(I),I=1,12)
      NTAG=1
      RETURN
C
 2002 FORMAT(/36H0      CURVED BEAM FORCES AND MOMENTS//
     . 10HOBEAM LOAD 5X 5HAXIAL  2(7X,5HSHEAR),5X 7HTORSION
     .  2(5X,7HBENDING)/ 10H  NO.  NO.  8X  2HR1  10X  2HR2  10X
     .  2HR3 10X 2HM) 10X 2HM2 10X 2HM3)
 3002 FORMAT (I5,I4,1PE11.3,5E12.3/8X,6E12.3/)
      END



      SUBROUTINE CTEAM (NCBEAM,NUMETP,NUMFIX,NUMMAT,NUMRAD,ID,X,Y,Z,
     .                  E,G,RO,SFT,COPROP,RAD,NUMNP,MBAND)
C
      COMMON/EM/LM(24),ND,NS,ASA(24,24),RF(24,4),XM(24),SA(12,24),
     .    SF(12,4)
      DIMENSION X(1),Y(1),Z(1),ID(NUMNP,1),E(1),G(1),SFT(NUMFIX,1)
     . ,COPROP(NUMETP,1),RO(1),EMUL(3,4),RAD(1)
      COMMON/MEMBCB/LC(4),TI(3,3),TJ(3,3),JK(6),MELTYP,DL,MATTYP,S(12,12)
     .             ,BETA,B,DP,D,RA,AREA
      DIMENSION ILC(4),STIF(722),LS(4)
      EQUIVALENCE (STIF,LM)
C
C     FORM 3-D CURVED BEAM STIFFNESS AND STRESS-DISP. ARRAYS
C
      WRITE (6,2005) NCBEAM,NUMETP,NUMFIX,NUMMAT,NUMRAD
      N=0
      DO 5 I=1,1058
    5 STIF(I)=0.
C
C     READ AND PRINT RADIUS OF CURVATURE DATA
C
      WRITE (6,2000)
      DO 8 I=1,NUMRAD
      READ (5,1000) N,RAD(N)
      IF (RAD(N).NE.0.0) GO TO 7
      WRITE (6,2014)
      CALL EXIT
    7 WRITE (6,2015) N,RAD(N)
    8 CONTINUE
C
C     READ AND PRINT MATERIAL PROPERTY DATA
C
      WRITE (6,2001)
      DO 10 I=1,NUMMAT
      READ  (5,1001) N,E(N),G(N),RO(N)
      WRITE (6,2002) N,E(N),G(N),RO(N)
   10 G(N)=0.5*E(N)/(1.+G(N))
C
C     READ AND PRINT CROSS SECTIONAL PROPERTIES
C
      WRITE (6,2003)
      DO 30 I=1,NUMETP
      READ  (5,1002) N,(COPROP(N,J),J=1,6)
      IF((COPROP(N,1).NE.0.0).AND.(COPROP(N,4).NE.0.0).AND.
     1   (COPROP(N,5).NE.0.0).AND.(COPROP(N,6).NE.0.0))  GO TO 20
      WRITE (6,2013)
      CALL EXIT
   20 WRITE (6,2004) N,(COPROP(N,J),J=1,6)
   30 CONTINUE
C
C     ELEMENT LOAD MULTIPLIERS
C
      READ  (5,1006) ((EMUL(I,J),J=1,4),I=1,3)
      WRITE (6,2006) ((EMUL(I,J),J=1,4),I=1,3)
C
C     READ AND PRINT FIXED END FORCES IN LOCAL COORDINATES
C
      IF(NUMFIX .EQ. 0) GO TO 56
      WRITE (6,2010)
      DO 55 I=1,NUMFIX
      READ  (5,1005) N,(SFT(N,J),J=1,12)
   55 WRITE (6,2011) N,(SFT(N,J),J=1,12)
   56 CONTINUE
C
C     READ AND PRINT ELEMENT DATA...NEED BE SUPPLIED FOR EACH ELEMENT
C
      WRITE (6,4000)
      L=0
   60 READ (5,3000) INEL,INI,INJ,INK,IMAT,IMEL,ILC,INELKI,INELKJ,IRAD
      L=L+1
      ML=INEL-L
      IF (ML)  66,67,66
   66 WRITE (6,4003) INEL
      CALL EXIT
   67 NEL=INEL
      NI=INI
      NJ=INJ
      NK=INK
      MATTYP=IMAT
      MELTYP=IMEL
      DO 90 I=1,4
   90 LC(I)=ILC(I)
      NEKODI=INELKI
      NEKODJ=INELKJ
      NR=IRAD
      WRITE (6,4001) NEL,NI,NJ,NK,MATTYP,MELTYP,LC,NEKODI,NEKODJ,NR
C
   74 DX=X(NJ)-X(NI)
```

```
CBEA 120          DY=Y(NJ)-Y(NI)
CBEA 121          DZ=Z(NJ)-Z(NI)
CBEA 122          DL=SQRT(DX*DX+DY*DY+DZ*DZ)
CBEA 123          IF(DL) 75,75,76
CBEA 124       75 WRITE (6,4005) NEL
CBEA 125          CALL EXIT
CBEA 126    C
CBEA 127    C     FORM GLOBAL TO LOCAL COORDINATE TRANSFORMATION.
CBEA 128    C
CBEA 129       76 RA=RAD(NR)
CBEA 130          ARGU=DL/(2.0*RA)
CBEA 131          BETA=2.0*ASIN(ARGU)
CBEA 132          B =SIN(BETA)
CBEA 133          DP=COS(BETA)
CBEA 134          D =1.0-DP
CBEA 135          BX=X(NI)-X(NK)
CBEA 136          BY=Y(NI)-Y(NK)
CBEA 137          BZ=Z(NI)-Z(NK)
CBEA 138          BB=BX*BX+BY*BY+BZ*BZ
CBEA 139          BB=SQRT(BB)
CBEA 140          IF (BB.GT.0.0) GO TO 40
CBEA 141          WRITE (6,4002) INEL
CBEA 142          STOP
CBEA 143    C
CBEA 144       40 CX=DY*BZ-DZ*BY
CBEA 145          CY=DZ*BX-DX*BZ
CBEA 146          CZ=DX*BY-DY*BX
CBEA 147          CC=CX*CX+CY*CY+CZ*CZ
CBEA 148          CC=SQRT(CC)
CBEA 149    C
CBEA 150          BX=BX/BB
CBEA 151          BY=BY/BB
CBEA 152          BZ=BZ/BB
CBEA 153          CX=CX/CC
CBEA 154          CY=CY/CC
CBEA 155          CZ=CZ/CC
CBEA 156          AX=BY*CZ-BZ*CY
CBEA 157          AY=BZ*CX-BX*CZ
CBEA 158          AZ=BX*CY-BY*CX
CBEA 159    C
CBEA 160          TI(1,1)=AX
CBEA 161          TI(1,2)=AY
CBEA 162          TI(1,3)=AZ
CBEA 163          TI(2,1)=BX
CBEA 164          TI(2,2)=BY
CBEA 165          TI(2,3)=BZ
CBEA 166          TI(3,1)=CX
CBEA 167          TI(3,2)=CY
CBEA 168          TI(3,3)=CZ
CBEA 169    C
CBEA 170          TJ(1,1)=AX*DP-BX*B
CBEA 171          TJ(1,2)=AY*DP-BY*B
CBEA 172          TJ(1,3)=AZ*DP-BZ*B
CBEA 173          TJ(2,1)=AX*B+BX*DP
CBEA 174          TJ(2,2)=AY*B+BY*DP
CBEA 175          TJ(2.3)=AZ*B+BZ*DP
CBEA 176          TJ(3,1)=CX
CBEA 177          TJ(3,2)=CY
CBEA 178          TJ(3,3)=CZ
CBEA 179    C
CBEA 180    C     CHECK IF NEW STIFFNESS NEEDED
CBEA 181    C
```

```
CBEA 182          IF (NEL.EQ.1) GO TO 80
CBEA 183          IF (MR.NE.NR)  GO TO 80
CBEA 184          IF  (ABS(DS-DL).GT.DL/100.)  GO TO 80
CBEA 185          IF ((MT.NE.MATTYP).OR.(ME.NE.MELTYP)) GO TO 80
CBEA 186          IF ((JK(1).NE.NEKODI).OR.(JK(2).NE.NEKODJ)) GO TO 60
CBEA 187          DO 81 I=1,4
CBEA 188          IF (LS(I).NE.LC(I)) GO TO 80
CBEA 189       81 CONTINUE
CBEA 190          GO TO 100
CBEA 191    C
CBEA 192       80 DS=DL
CBEA 193          MT=MATTYP
CBEA 194          ME=MELTYP
CBEA 195          MR=NR
CBEA 196          DO 78 I=1,4
CBEA 197       78 LS(I)=LC(I)
CBEA 198          JK(1)=NEKODI
CBEA 199          JK(2)=NEKODJ
CBEA 200    C
CBEA 201    C     FORM NEW CURVED BEAM STIFFNESS
CBEA 202    C
CBEA 203          CALL NEWCBM (E,G,RO,COPROP,SFT,RAD,NUMFIX,NUMETP)
CBEA 204    C
CBEA 205      100 CALL TRANSFM (PJ)
CBEA 206    C
CBEA 207    C     ADD GRAVITY LOADING....POINT LOADS ONLY COMPUTED
CBEA 208    C
CBEA 209          DO 180 I=1,3
CBEA 210          DO 180 J=1,4
CBEA 211          RF(I,J)=RF(I,J)+EMUL(I,J)*XM(I)
CBEA 212      180 RF(I+6,J)=RF(I+6,J)+EMUL(I,J)*XM(I+6)
CBEA 213    C
CBEA 214    C     FORM ELEMENT LOCATION MATRIX
CBEA 215    C
CBEA 216          DO 170 M=1,6
CBEA 217          LM(M)=ID(NI,M)
CBEA 218          LM(M+12)=0
CBEA 219          LM(M+18)=0
CBEA 220      170 LM(M+6)=ID(NJ,M)
CBEA 221    C
CBEA 222          NS=12
CBEA 223          ND=12
CBEA 224    C
CBEA 225    C     TRANSFORM TO MASTER DEGREES OF FREEDOM
CBEA 226    C
CBEA 227          CALL SLAVE (X,Y,Z,ID,NUMNP,NI,NJ)
CBEA 228    C
CBEA 229    C     WRITE ELEMENT INFORMATION ON TAPE
CBEA 230    C
CBEA 231          CALL WRITET (MBAND,NDIF)
CBEA 232    C
CBEA 233    C     CHECK FOR LAST ELEMENT
CBEA 234    C
CBEA 235          IF (NCREAM-NEL) 66,500,60
CBEA 236      500 RETURN
CBEA 237    C
CBEA 238     1000 FORMAT(I5,F10.0)
CBEA 239     1001 FORMAT(I5,3F10.0)
CBEA 240     1002 FORMAT(I5,6F10.0)
CBEA 241     1005 FORMAT(I5,6F10.0/F15.0,5F10.0)
CBEA 242     1006 FORMAT (4F10.0)
CBEA 243     2000 FORMAT (////
```

```
     .         27H RADIUS        RADIUS OF      /
     .        27H   NO           CURVATURE    //)
2001 FORMAT(55H1MATERIAL   YOUNG S     POISSON S           MASS
     .      /55H           MODULUS       RATIO            DENSITY      /
     .      /)
2002 FORMAT(1H ,I5,3X,F12.0,F14.5,F14.5)
2003 FORMAT(1H1/
    0  26H BEAM SECTIONAL PROPERTIES//
    1   48H ELEMENT      AREA         AREA        AREA
    2  60H INERTIA     INERTIA     INERTIA
    3 / 48H  TYPE           X     .     Y           Z
    4                 30H    X          Y           Z                  //)
2004 FORMAT(1H ,I5,2X,6F12.3)
2005 FORMAT(39H1.......THREE DIM. CURVED BEAM ELEMENTS//
     .  36H  NUMBER OF CURVED BEAMS              =,I5/
     .  36H  NUMBER OF SECTIONAL PROPERTY SETS=,I5/
     .  36H  NUMBER OF FIXED END FORCE SETS   =,I5/
     .  36H  NUMBER OF MATERIALS              =,I5/
     .  36H  NUMBER OF RADII                  =,I5)
2006 FORMAT(///25H ELEMENT LOAD MULTIPLIERS / 10X,1HA,14X,1HB,14X,1HC,
     .14X,1HD,//6H X-DIR4E15.6/ 6H Y-DIR4E15.6/ 6H Z-DIR4E15.6/ )
2010 FORMAT(1H1,1H ,
    1  30X40H FIXED END FORCES IN LOCAL COORDINATES
    2//53H TYPE   NODE        FORCE X     FORCE Y     FORCE Z
    3                     35H MOMENT X    MOMENT Y    MOMENT Z          //)
2011 FORMAT(1H ,I3,6X,1HI,3X,6F12.3/1H ,9X,1HJ,3X,6F12.3/)
2013 FORMAT(1H0/
    1  60H SECTION PROPERTIES OTHER THAN SHEAR AREAS MAY NOT BE SPECIF
    2  34HIED AS ZERO. EXECUTION TERMINATED.)
2014 FORMAT(1H0/
    1  60H RADIUS OF CURVATURE MAY NOT BE SPECIFIED
    2  34H    AS ZERO. EXECUTION TERMINATED.)
2015 FORMAT (I5,F20.5)
3000 FORMAT (10I5,2I6,I8)
4000 FORMAT(1H1/
     . 5H0BEAM 5X 5HNODES 5X 5H MATL 5H GEOM 5X 10HELEM LOADS 4X   10X
     . 12H  END CODES ,10X,6HRADIUS/
     .                        5H  NO      4X 1HI 4X 1HJ 4X 1HK 5H  ND  5H  NO
     .  4X 1HA 4X 1HB 4X 1HC 4X 1HD 9X 1HI 9X 1HJ ,12X,2HRO//)
4001 FORMAT (10I5,2I10,I14)
4002 FORMAT (9H0BEAM NO ,I5, 26H      K NODE ON BEAM X-AXIS      ,
     .  26H......EXECUTION TERMINATED )
4003 FORMAT(36H0ELEMENT CARD ERROR, ELEMENT NUMBER= I6)
4004 FORMAT(1H ,31HNODAL POINT NUMBERS FOR ELEMENT,I5,36HARE IDENTICAL.
     . EXECUTION TERMINATED.)
4005 FORMAT(8HOELEMENT,I5,39H HAS ZERO LENGTH. EXECUTION TERMINATED.)
      END



      SUBROUTINE NEWCBM (E,G,RO,COPROP,SFT,RAD,NUMFIX,NUMETP)
C
C     FORM NEW CURVED BEAM STIFFNESS
C
      DIMENSION E(1),G(1),RO(1),COPROP(NUMETP,1),SFT(NUMFIX,1),RAD(1)
      COMMON/FM/LM(24),ND,NS,ASA(24,24),RF(24,4),XM(24),SA(12,24),
     .    SF(12,4)
      COMMON/NEWCB/LC(4),TI(3,3),TJ(3,3),JK(6),MELTYP,DL,MATTYP,S(12,12)
     .             ,BETA,B,DP,D,RA,AX
      DIMENSION R(12),C(12)
C
      DO 5 I=1,144
    5 S(I)=0.
       AX=CCPROP(MELTYP,1)
       AY=CCPROP(MELTYP,2)
       AZ=CCPROP(MELTYP,3)
      AAX=CCPROF(MELTYP,4)
      AAY=CCPROP(MELTYP,5)
      AAZ=CCPROP(MELTYP,6)
C
      RA2=RA*RA
      RA3=RA2*RA
      EIY=E(MATTYP)*AAY
      EIZ=E(MATTYP)*AAZ
      PHIZ=AAZ/(AX*RA2)
      PHIY=EIY/(G(MATTYP)*AAX)
C
      IF (AY.LE.0.) GO TO 20
      ZETAZ=EIZ/(AY*G(MATTYP)*RA2)
      GO TO 21
   20 ZETAZ=0.
   21 IF (AZ.LE.0.) GO TO 30
      ZETAY=EIY/(AZ*G(MATTYP)*RA2)
      GO TO 31
   30 ZETAY=0.
   31 CONTINUE
      COMMY=RA3/EIY
      COMMZ=RA3/EIZ
C
C     FIXED END FORCES IN LOCAL COORDINATES
C
      DO 73 N=1,4
      M=LC(N)
      IF (M.GT.0)  GO TO 71
      DO 70 I=1,12
   70 SF(I,N)=0.
      GO TO 73
   71 DO 72 I=1,12
   72 SF(I,N)=SFT(M,I)
   73 CONTINUE
C
C     FORM ELEMENT STIFFNESS IN LOCAL COORDINATES
C
      AA=BETA*0.5 C.5*B*DP
      CC=BETA*C.5+C.5*B*DP
      EE=0.5*B*B
C
      F11=COMMZ*(BETA+CC-2.0*B+PHIZ*CC+ZETAZ*AA)
      F12=COMMZ*(D-EE-PHIZ*EE+ZETAZ*EE)
      F16=(COMMZ/RA)*(B-BETA)
      F22=COMMZ*(AA+PHIZ*AA+ZETAZ*CC)
      F26=(COMMZ/RA)*(-D)
C
      F33=COMMY*(AA+PHIY*(BETA+CC-2.0*B)+ZETAY*BETA)
      F34=(COMMY/RA)*(AA-PHIY*(B-CC))
      F35=(COMMY/RA)*(EE+PHIY*(D EE))
      F44=(COMMY/RA2)*(AA+PHIY*CC)
      F45=(COMMY/RA2)*(EE-PHIY*EE)
      F55=(COMMY/RA2)*(CC+PHIY*AA)
      F66=(COMMZ/RA2)*BETA
C
      U=F11*F22*F66+2.0*F12*F16*F26-F22*F16*F16-F11*F26*F26-F66*F12*F12
      W=F33*F44*F55+2.0*F34*F45*F35-F44*F35*F35-F33*F45*F45-F55*F34*F34
```

```
C
      S(1,1)= (F22*F66-F26*F26)/U
      S(1,2)=-(F12*F66-F16*F26)/U
      S(1,6)= (F12*F26-F22*F16)/U
      S(2,2)= (F11*F66-F16*F16)/U
      S(2,6)=-(F11*F26-F12*F16)/U
      S(6,6)= (F11*F22-F12*F12)/U
C
      S(3,3)= (F44*F55-F45*F45)/W
      S(3,4)=-(F34*F55-F35*F45)/W
      S(3,5)= (F34*F45-F44*F35)/W
      S(4,4)= (F33*F55-F35*F35)/W
      S(4,5)=-(F33*F45-F35*F34)/W
      S(5,5)= (F33*F44-F34*F34)/W
C
      S(1, 7)=-S(1,1)*OP+S(1,2)*B
      S(1, 8)=-S(1,1)*B-S(1,2)*OP
      S(1,12)= S(1,1)*RA*D+S(1,2)*RA*B-S(1,6)
      S(2, 7)=-S(1,2)*OP+S(2,2)*B
      S(2, 8)= S(1,2)*B-S(2,2)*OP
      S(2,12)= S(1,2)*RA*D+S(2,2)*RA*B-S(2,6)
      S(3, 9)= S(3,3)
      S(3,10)= S(3,3)*RA*D-S(3,4)*OP+S(3,5)*B
      S(3,11)=-S(3,3)*RA*B-S(3,4)*B-S(3,5)*OP
      S(4, 9)= S(3,4)
      S(4,10)= S(3,4)*RA*D-S(4,4)*OP+S(4,5)*B
      S(4,11)=-S(3,4)*RA*B-S(4,4)*B - S(4,5)*OP
      S(5, 9)= S(3,5)
      S(5,10)= S(3,5)*RA*D-S(4,5)*OP+S(5,5)*B
      S(5,11)=-S(3,5)*RA*B-S(4,5)*B-S(5,5)*OP
      S(6, 7)= S(1,6)*OP+S(2,6)*B
      S(6, 8)=-S(1,6)*B-S(2,6)*OP
      S(6,12)= S(1,6)*RA*D+S(2,6)*RA*B-S(6,6)
C
      S( 7, 7)= S(1,1)
      S( 7, 8)=-S(1,2)
      S( 7,12)= S(1,6)
      S( 8, 8)= S(2,2)
      S( 8,12)=-S(2,6)
      S( 9, 9)= S(3,3)
      S( 9,10)= S(3,4)
      S( 9,11)=-S(3,5)
      S(10,10)= S(4,4)
      S(10,11)=-S(4,5)
      S(11,11)= S(5,5)
      S(12,12)= S(6,6)
C
      DO 106 I=2,12
      K=I-1
      DO 106 J=1,K
  106 S(I,J)=S(J,I)
C
C     MODIFY ELEMENT STIFFNESS AND FORCES FOR ZERO END FORCE CONDITION
C
      IF ((JK(1)+JK(2)).EQ.0) GO TO 145
      DO 140 K=1,2
      KK=JK(K)
      KD=100000
      I1=6*(K-1)+1
      I2=I1+5
      DO 140 I=I1,I2
      IF (KK.LT.KD)  GO TO 140
      SII=S(I,I)
      DO 125 N=1,12
  125 R(N)=S(I,N)
      DO 130 M=1,12
      C(M)=S(M,I)/SII
      DO 130 N=1,12
  130 S(M,N)=S(M,N)-C(M)*R(N)
      DO 135 N=1,4
      SFI=SF(I,N)
      DO 135 M=1,12
  135 SF(M,N)=SF(M,N)-C(M)*SFI
  136 KK=KK-KD
  140 KD=KD/10
  145 CONTINUE
      RETURN
      END


      SUBROUTINE TRANSFM (RO)
C
      DIMENSION RO(1)
      COMMON/EM/LM(24),ND,NS,ASA(24,24),RF(24,4),XP(24),SA(12,24),
     .    SF(12,4)
      COMMON/NEWCB/LC(4),TI(3,3),TJ(3,3),JK(6),MELTYP,DL,MATTYP,S(12,12)
     .             ,BETA,B,OP,D,RA,AX
C
C     FORM LOCAL FORCE TO GLOBAL DISPLACEMENT ARRAY
C
      DO 31 I=1,288
   31 SA(I)=0.
C
      DO 150 LA=1,10,3
      LB=LA+2
      DO 150 MA=1,10,3
      MB=MA-1
      DO 150 I=LA,LB
      DO 150 JM=1,3
      J=JM+MB
      XX=0.
      IF (J.GT.6) GO TO 152
      DO 151 K=1,3
  151 XX=XX+S(I,K+MB)*TI(K,JM)
      GO TO 150
  152 DO 153 K=1,3
  153 XX=XX+S(I,K+MB)*TJ(K,JM)
  150 SA(I,J)=XX
C
C     ELEMENT STIFFNESS AND FORCES IN GLOBAL COORDS
C
      DO 32 I=1,576
   32 ASA(I)=0.
C
      DO 160 LA=1,10,3
      LB=LA-1
      DO 160 MA=1,10,3
      MB=MA+2
      DO 160 IL=1,3
      I=IL+LB
      DO 160 J=MA,MB
      XX=0.
```

```
      IF (I.GT.6) GO TO 162
      DO 161 K=1,3
  161 XX=XX+TI(K,IL)*SA(K+LB,J)
      GO TO 160
  162 DO 163 K=1,3
  163 XX=XX+TJ(K,IL)*SA(K+LB,J)
  160 ASA(I,J)=XX
C
      DO 170 LA=1,IC,3
      LB=LA-1
      DO 170 IL=1,3
      I=IL+LB
      DO 170 N=1,4
      XX=0.
      IF (I.GT.6) GO TO 172
      DO 171 K=1,3
  171 XX=XX-TI(K,IL)*SF(K+LB,N)
      GO TO 170
  172 DO 173 K=1,3
  173 XX=XX-TJ(K,IL)*SF(K+LB,N)
  170 PF(I,N)=XX
C
C     FORM MASS MATRIX
C
      XXM=RO(MATTYP)*AX*RA*BETA/2.0
      DO 180 M=1,3
      XM(M)=XXM
      XM(M+3)=0.
      XM(M+9)=0.
  180 XM(M+6)=XXM
      RETURN
      END
```

```
      SUBROUTINE BOUND
C
      COMMON/ELPAR/ NPAR(14),NUMNP,MBAND,NELTYP,N1,N2,N3,N4,N5,MTOT,NEG
      COMMON/JUNK/  MM,L,K,NTAG,NDYN,SIG(12),EXRA(188)
      COMMON A(1)
C
      IF (NPAR(1).EQ.0) GO TO 500
      N6=N5+NPAR(3)*6
      IF (N6.GT.MTOT) CALL ERROR (N6-MTOT)
      CALL BIND (NPAR(2),NPAR(3),A(N1),A(N2),A(N3),A(N4),A(N5),
     .              NUMNP,MBAND)
      RETURN
C
  500 IF (NTAG.EQ.0) WRITE (6,2002)
      WRITE(6,3002) MM,L,(SIG(I),I=1,6 )
      NTAG=1
      RETURN
C
 2002 FORMAT (/33H0 ....BOUNDARY FORCES AND MOMENTS  //
     .  10H0 BD. LOAD 5X 5HAXIAL 2(7X 5HSHEAR) 5X 7HTORSION
     .  2(5X 7HBENDING) /10H  NO    NO  8X 2HR1 10X 2HR2 10X 2HR3 10X
     .  2HM1 10X 2HM2 10X 2HM3 )
 3002 FORMAT (I5,I4,1PE11.3,5E12.3)
C
      END
```

```
      SUBROUTINE BIND (NBOUND,NSTIF,ID,X,Y,Z,STIF,NUMNP,MBAND)
C
      DIMENSION ID(NUMNP,1),X(1),Y(1),Z(1),STIF(NSTIF,1),T(3,3),
     .              SPRING(6)
      COMMON/EM/LM(24),ND,NS,ASA(24,24),P(24,4),XM(24),SA(12,24),
     .              SF(12,4)
      COMMON/JUNK/  DATA(17),S(12,12)
C
C     FORM ELEMENT STIFFNESS OF BOUNDARY SPRING ELEMENTS
C
      DO 5 I=1,1058
    5 LM(I)=0.
      ND=6
      NS=6
      NE=0
      WRITE(6,2000) NBOUND,NSTIF
C
C     READ AND PRINT SPRING STIFNESS SETS
C
      READ (5,1010) (N,(STIF(N,J),J=1,6),N=1,NSTIF)
      WRITE(6,2001)
      WRITE(6,2010) (N,(STIF(N,J),J=1,6),N=1,NSTIF)
C
C     READ AND PRINT ELEMENT DATA---DATA NEED TO BE SUPPLIED FOR EACH
C
      WRITE(6,2004)
   10 NE=NE+1
      READ (5,1030) INE,NI,NJ,NK,NS
      IF (INE.EQ.NE) GO TO 20
      WRITE(6,3000) INE
      STOP
   20 WRITE(6,2040) NE,NI,NJ,NK,NS
```

```
C
C     FORM LOCAL TO GLOBAL TRANSFORMATION
C
      DX=X(NJ)-X(NI)
      DY=Y(NJ)-Y(NI)
      DZ=Z(NJ)-Z(NI)
      DD=SQRT(DX*DX+DY*DY+DZ*DZ)
      IF (DD.GT.0.) GO TO 30
   25 WRITE(6,3001) NE
      STOP
   30 AX=X(NK)-X(NI)
      AY=Y(NK)-Y(NI)
      AZ=Z(NK)-Z(NI)
      BX=DY*AZ-DZ*AY
      BY=DZ*AX-DX*AZ
      BZ=DX*AY-DY*AX
      BB=SQRT(BX*BX+BY*BY+BZ*BZ)
      IF (BB.LE.0.) GO TO 25
C
      T(3,1)=BX/BB
      T(3,2)=BY/BB
      T(3,3)=BZ/BB
      T(1,1)=DX/DD
      T(1,2)=DY/DD
      T(1,3)=DZ/DD
      T(2,1)=T(3,2)*T(1,3)-T(3,3)*T(1,2)
      T(2,2)=T(3,3)*T(1,1)-T(3,1)*T(1,3)
      T(2,3)=T(3,1)*T(1,2)-T(3,2)*T(1,1)
C
C     FORM BOUNDARY STIFFNESSES IN LOCAL COORDINATE SYSTEM
C
      DO 100 I=1,144
  100 S(I)=0.
      DO 110 I=1,6
      SPRING(I)=STIF(NS,I)
      S(I,I)=SPRING(I)
  110 CONTINUE
C
C     FORM LOCAL TO GLOBAL STRESS ARRAY
C
      DO 120 I=1,288
  120 SA(I)=0.
      DO 150 LA=1,6,3
      LB=LA+2
      DO 150 MA=1,6,3
      MB=MA-1
      DO 150 I=LA,LB
      DO 150 JM=1,3
      J=JM+MB
      XX=0.
      DO 151 K=1,3
      YY=S(I,K+MB)
      IF (YY.EQ.0.) GO TO 151
      XX=XX+YY*T(K,JM)
  151 CONTINUE
  150 SA(I,J)=XX
C
C     FORM ELEMENT STIFFNESSES IN GLOBAL COORDINATE SYSTEM
C
      DO 155 I=1,576
  155 ASA(I)=0.
      DO 160 LA=1,6,3
      LB=LA-1
      DO 160 MA=1,6,3
      MB=MA+2
      DO 160 IL=1,3
      I=IL+LB
      DO 160 J=MA,MB
      XX=0.
      DO 161 K=1,3
      YY=SA(K+LB,J)
      IF (YY.EQ.0.) GO TO 161
      XX=XX+T(K,IL)*YY
  161 CONTINUE
  160 ASA(I,J)=XX
C
C     FORM ELEMENT LOCATION MATRIX
C
      DO 170 I=1,6
      LM(I)=ID(NI,I)
      LM(I+6)=0
      LM(I+12)=0
  170 LM(I+18)=0
C
C     WRITE ELEMENT INFORMATION ON TAPE
C
      CALL WRITET (MBAND,NDIF)
C
  250 IF (NE.LT.NBOUND) GO TO 10
      RETURN
C
 1010 FORMAT (I5,6F10.0)
 1030 FORMAT (6I5)
 2000 FORMAT (24H1......BOUNDARY ELEMENTS   ///
     .  28H NUMBER OF ELEMENTS           =,I5//
     =  28H NUMBER OF STIFFNESS SETS  =,I5)
 2001 FORMAT (4H1SET 20X 31HSTIFFNESSES OF BOUNDARY SPRINGS /
     .  4H NO  4X 2HR1 10X 2HR2 10X 2HR3 10X 2HM1 10X 2HM2 10X 2HM3 //)
 2004 FORMAT (5H1ELE. 5X 5HNODES 5X 5H MATL     /
     .  5H  NO  4X 1HI 4X 2HJ 4X 1HK 5H  NO      //)
 2010 FORMAT (I4,6E12.3)
 2040 FORMAT (5I5)
 3000 FORMAT (32H1*****BOUNDARY ELEMENT DATA NO. ,I3,15H IN WRONG ORDER /
     =           21H .....XEQ TERMINATED  )
 3001 FORMAT (35H1*****J/K NODE FOR BOUNDARY ELEMENT NO. ,I3,
     .           21HSPECIFIED INCORRECTLY  /21H .....XEQ TERMINATED  )
C
      END
```

```
      SUBROUTINE EXPJ8
C
      COMMON A(1)
      COMMON/ELPAR/ NPAR(14),NUMNP,MBAND,NELTYP,N1,N2,N3,N4,N5,MTOT,NEQ
      COMMON/JUNK/ MM,L,K,NTAG,NDYN,SIG(12),EXPA(188)
C
      IF (NPAR(1).EQ.0) GO TO 500
      CALL XPJ8 (NPAR(2),A(N1),A(N2),A(N3),A(N4),NUMNP,MBAND)
C
      RETURN
C
  500 IF (NTAG.EQ.0) WRITE(6,2002)
      WRITE(6,3002) MM,L,(SIG(I),I=1,12)
      NTAG=1
      RETURN
C
 2002 FORMAT (/29H0....JOINT FORCES AND MOMENTS  //
     .  10H0 JT. LOAD,5X,5HAXIAL,2(7X,5HSHEAR),5X,7HTORSION,
     .  2(5X,7HBENDING)/10H NO.  NO. ,8X,2HR1,10X,2HR2,10X,2HR3,
     .  10X,2HM1,10X,2HM2,10X,2HM3)
 3002 FORMAT (I5,I4,1PE11.3,5E12.3/8X,6E12.3/)
C
      END
```

```
      SUBROUTINE XPJ8 (NUMEL,ID,X,Y,Z,NUMNP,MBAND)
C
      COMMON/EP/ LM(24),ND,NS,ASA(24,24),P(24,4),XM(24),SA(12,24),
     .             SF(12,4)
      COMMON/JUNK/ MPM(5),SIG(12),S(12,12),T(3,3),KD(6),XXX(29)
      DIMENSION X(1),Y(1),Z(1),ID(NUMNP,1)
C
C     FORM 3-D STIFFNESSES FOR EXPANSION JOINT ELEMENTS
C
      DO 5 I=1,1058
    5 LM(I)=0.
      ND=12
      NS=12
      NE=0
C
C     INPUT EXPANSION JOINT DATA
C
      WRITE(6,2000) NUMEL
      WRITE(6,2010)
   10 NE=NE+1
      READ (5,1030) INE,NI,NJ,NK,NL,KD,TRACE
      IF (INE.EQ.NE) GO TO 15
      WRITE (6,4002)
      CALL EXIT
   15 IF (TRACE.EQ.0.) TRACE=1.0E+10
      IF (NK.EQ.0) READ (5,1040) XNK,YNK,ZNK
      IF (NL.EQ.0) READ (5,1060) XNL,YNL,ZNL
C
      WRITE(6,2020) NE,NI,NJ,NK,NL,KD
      IF (NK.EQ.0) GO TO 30
      XNK=X(NK)
      YNK=Y(NK)
      ZNK=Z(NK)
      GO TO 35
   30 WRITE(6,2060) XNK,YNK,ZNK
   35 IF (NL.EQ.0) GO TO 40
      XNL=X(NL)
      YNL=Y(NL)
      ZNL=Z(NL)
      GO TO 50
   40 WRITE(6,2070) XNL,YNL,ZNL
C
C     FORM LOCAL TO GLOBAL TRANSFORMATION
C
   50 DX=XNL-X(NI)
      DY=YNL-Y(NI)
      DZ=ZNL-Z(NI)
      DL=SQRT(DX*DX+DY*DY+DZ*DZ)
      IF (DL) 55,55,60
   55 WRITE(6,4000) NE
      CALL EXIT
C
   60 AX=XNK-X(NI)
      AY=YNK-Y(NI)
      AZ=ZNK-Z(NI)
C
      BX=AY*DZ-AZ*DY
      BY=AZ*DX-AX*DZ
      BZ=AX*DY-AY*DX
      BB=SQRT(BX*BX+BY*BY+BZ*BZ)
      IF (BB) 65,65,70
   65 WRITE(6,4001) NE
      CALL EXIT
C
   70 T(3,1)=BX/BB
      T(3,2)=BY/BB
      T(3,3)=BZ/BB
C
      AA=SQRT(AX*AX+AY*AY+AZ*AZ)
      IF (AA) 65,65,80
   80 T(2,1)=-AX/AA
      T(2,2)=-AY/AA
      T(2,3)=-AZ/AA
C
      T(1,1)=T(2,2)*T(3,3)-T(2,3)*T(3,2)
      T(1,2)=T(2,3)*T(3,1)-T(2,1)*T(3,3)
      T(1,3)=T(2,1)*T(3,2)-T(2,2)*T(3,1)
C
C     FORM JOINT STIFFNESS IN LOCAL COORDINATE SYSTEM
C
      DO 100 I=1,144
  100 S(I)=0.
      DO 110 I=1,6
      IF (KD(I).EQ.0) GO TO 110
      S(I,I)=TRACE
      S(I,I+6)=-TRACE
      S(I+6,I)=-TRACE
      S(I+6,I+6)=TRACE
  110 CONTINUE
C
C     FORM LOCAL-GLOBAL STRESS ARRAY
C
      DO 120 I=1,288
  120 SA(I)=0.
      DO 150 LA=1,10,3
      LB=LA+2
```

```
      DO 150 MA=1,10,3
      MB=MA-1
      DO 150 I=LA,LB
      DO 150 JM=1,3
      J=JM+MB
      XX=0.
      DO 151 K=1,3
      YY=S(I,K+MB)
      IF (YY.EQ.0.) GO TO 151
      XX=XX+YY*T(K,JM)
  151 CONTINUE
  150 SA(I,J)=XX
C
C     FORM JOINT STIFFNESS IN GLOBAL COORDINATE SYSTEM
C
      DO 155 I=1,576
  155 ASA(I)=0.
      DO 160 LA=1,10,3
      LB=LA-1
      DO 160 MA=1,10,3
      MB=MA+2
      DO 160 IL=1,3
      I=IL+LB
      DO 160 J=MA,MB
      XX=0.
      DO 161 K=1,3
      YY=SA(K+LB,J)
      IF (YY.EQ.0.) GO TO 161
      XX=XX+T(K,IL)*YY
  161 CONTINUE
  160 ASA(I,J)=XX
C
C     FORM ELEMENT LOCATION MATRIX
C
      DO 170 I=1,6
      LM(I)=ID(NI,I)
      LM(I+6)=ID(NJ,I)
      LM(I+12)=0
  170 LM(I+18)=0
C
C     WRITE ELEMENT INFORMATION ON TAPE
C
      CALL WRITET (MBAND,NDIF)
C
      IF (NE.LT.NUMEL) GO TO 10
C
      RETURN
 1030 FORMAT (5I5,6I1,F19.0)
 1040 FORMAT (3F10.0)
C
 2000 FORMAT (31H1....EXPANSION JOINT ELEMENT...  ///
     .  29H TOTAL NUMBER OF ELEMENTS...=  ,I5)
 2010 FORMAT (//4H  NE,5H   N1,5H   NJ,9H    NK   ,9H    NL   ,3X,
     .  6HJ.CODE  /)
 2020 FORMAT (I4,2I5,I6,3X,I6,6X,6I1   )
 2060 FORMAT (14X,F9.3/14X,F9.3/14X,F9.3)
 2070 FORMAT (23X,F9.3/23X,F9.3/23X,F9.3)
 4000 FORMAT (10H0  ELEMENT,I5,39H HAS ZERO LENGTH. EXECUTION TERMINATED
     ..  )
 4001 FORMAT (10H0  ELEMENT,I5,40H HAS WRONG K NODE. EXECUTION TERMINATE
     .D.  )
 4002 FORMAT (40H0...E.J. ELEMENT DATA IN WRONG ORDER... )
C
      END
```

APPENDIX III

FORTRAN IV LISTING OF PROGRAM NEABS

```
      PROGRAM NEABS (INPUT,OUTPUT,TAPE5=INPUT,TAPE6=OUTPUT,TAPE1,TAPE2,
     .               TAPE3=2100,TAPE4,TAPE7,TAPE8,TAPE9,TAPE10,TAPE11,
     .               TAPE12,TAPE13,TAPE14,TAPE15,TAPE30)
C
C
C     PROGRAM FCR NCNLINEAR EARTHQUAKE ANALYSIS OF BRIDGE STRUCTURES
C
C
      COMMON/ELPAR/NPAR(14),NUMNP,NELTYP,NUMEL,NUMNEL,NEQ,NEQG,MBAND,
     .             NEQB,NBLOCK,MTOT,N1,N2,N3,N4,N5,N6,N7
      COMMCN/EM/ GGCQ(2043)
      COMMCN/MISC/ LG,LL,NFNR,NFN,NGM,NAT,NT,NOT,MSTEP
      COMMCN/JUNK/ HED(12),JUK(193)
      DIMENSION T(1C)
C
C     PROGRAM CAPACITY CONTROLLED BY THE FOLLOWING TWO STATEMENTS...
C
      COMMON A(200CC)
      MTOT=20000
C
C     PROGRAM CCNTRCL OATA
C
    5 CALL SECPNO (T(1))
      READ (5,1C0) HEO,NUMNP,NELTYP,NUMEL.NFN,NGM,NAT,NT,NOT,MSTEP
      IF (NUMNP.EQ.C) STOP
      WRITE(6,200) HED,NUMNP,NELTYP,NUMEL,NFN,NGM,NAT,NT,NOT,MSTEP
      IF (NAT.EQ.0) NAT=1
C
C     SETUP STRLCTURAL SYSTEM INPUTS
C
      CALL SETUP
      CALL SECUND (T(2))
C
C     STATIC ANALYSIS
C
      CALL STATIC
      CALL SECOND (T(3))
C
C     SETUP OYNAMIC LOAD INPUTS
C
      CALL LOADS
      CALL SECOND (T(4))
C
C     STEP-BY-STEP CYNAMIC ANALYSIS
C
      IF (MSTEP.EQ.1) CALL STEPC
      IF (MSTEP.EQ.2) CALL STEPB
      CALL SECOND (T(5))
C
C     OUTPUT RESULTS OF STEP-BY-STEP OYNAMIC ANALYSIS
C
      CALL CUTPUT
      CALL SECOND (T(6))
C
C     OUTPUT CP EXECUTION TIMES
C
      TT=0.
      DO 90 I=1,5
      II=I+1
      T(I)=T(II)-T(I)
   90 TT=TT+T(I)
      T(6)=TT
      WRITE(6.205) (T(I),I=1,6)
C
      STOP
C
  100 FORMAT (12A6/9I5)
  200 FORMAT (1H1,12A6 ///
     .            30H  NUMBER OF NODAL POINTS       =  I5//
     .            30H  NUMBER OF ELEMENT TYPES      =  I5//
     .            30H  NUMBER OF TOTAL ELEMENTS     =  I5//
     .            30H  NUMBER OF TIME FUNCTIONS     =  I5//
     .            3CH  GROUND MOTION INPUT CODE     =  I5//
     .            30H  NUMBER OF DELAY TIMES        =  I5//
     .            3CH  NUMBER OF TIME STEPS (DT)    =  I5//
     .            30H  DT MULTIPLIER FOR OUTPUT     =  I5//
     .            3CH  STEP-BY-STEP SOLUTION CODE   =  I5  )
  205 FORMAT (21H1OVERALL TIME LOG....  ///
     .            33H  STRUCTURAL SYSTEM SETUP  ...... ,F8.2//
     .            33H  STATIC ANALYSIS  .............. ,F8.2//
     .            33H  SET UP DYNAMIC LOAD INPUTS  ... ,F8.2//
     .            33H  STEP-BY-STEP DYNAMIC ANALYSIS . ,F8.2//
     .            33H  CUTPUT DYNAMIC RESPONSES ...... ,F8.2///
     .            33H  TOTAL SOLUTION TIME  .......... ,F8.2  )
C
      END


      SUBROUTINE ERROR (MODE,N,PM)
C
C     PRINT APPROPRIATE ERROR MESSAGE...
C
      GO TO (1,2,3,4,5,6,7,8,9)  MODE
C
C     ERROR MOOE 1
C
    1 WRITE(6,2001) N,PM
      GO TO 100
C
C     ERROR MOOE 2
C
    2 WRITE(6,2002) N,PM
      GO TO 100
C
C     ERROR MODE 3
C
    3 WRITE(6,2003) N,PM
      GO TO 100
C
C     ERROR MODE 4
C
    4 WRITE(6,2004) N,PM
      GO TO 100
C     ERROR MODE 5
C
    5 WRITE(6,2005) N,PM
      GO TO 100
C
C     ERROR MODE 6
C
    6 WRITE(6,2006) N,PM
```

```
      GO TO 100
C
C     ERROR MODE 7
C
    7 WRITE(6,2007) N,PH
      GO TO 100
C
C     ERROR MODE 8
C
    8 WRITE(6,2008) N,PH
      GU TO 100
C
C     ERROR MODE 9
C
    9 WRITE(6,2009) N,PH
  100 STOP
C
 2001 FORMAT (1H1,44H*****DIMENSION IN BLANK COMMON A EXCEEDED BY,I6,
     .  20H    AT SLBRUUTINE    ,A7/27H .....EXECUTION TERMINATED.)
 2002 FORMAT (1H1,42H*****WRONG INPUT FOR TOTAL NO. OF ELEMENTS ,I6.
     .  20H    AT SLBROUTINE    ,A7/27H .....EXECUTION TERMINATED.)
 2003 FORMAT (1H1,48H*****ELEMENT NONLINEAR PARAMETER NO. INPUT WRONG,
     .  10H   FCR NO. ,I6,17H     ELEMENT OF     ,A7/
     .  27H .....EXECUTION TERHINATED.)
 2004 FORMAT (1H1,50H*****ELEMENT DATA INPUT IN WRONG ORDER DETECTED AT,
     .  3HNO.,I6,17H    ELLMENT OF    ,A7/27H .....EXECUTION TERMINATED.)
 2005 FORMAT (1H1,16H*****ELEMENT NO. ,I6,9H    OF    ,A7,12HHAS 0 LENGTH
     .  27H .....EXECUTION TERMINATED.)
 2006 FORMAT (1H1,16H*****ELEMENT NO. ,I6,9H    OF    ,A7,17HHAS WRONG K
     .NODE. /27H .....EXECUTION TERMINATED.)
 2007 FORMAT (1H1,37H*****DATA INPUT NOT IN ORDER FOR NODE,I6,
     .  15H    DETECTED AT ,A7/27H .....EXECUTION TERMINATED.)
 2008 FORMAT (1H1,41H*****MULTIPLE GROUND MOTION INPUT AT NODE,I6/
     .  50H      NCT COMPATIBLE WITH THAT OF JOINT INPUT CODE ,
     .  13H  DETECTED AT,A7/27H .....EXECUTION TERMINATED.)
 2009 FORMAT (1H1,25H*****NONLINEAR ELEMENT NO ,I6,16H     IS SINGULAR.
     .  15H    DETECTED AT ,A7/27H .....EXECUTION TERMINATED. )
      END
```

```
      SUBRCUTINE SETUP
C
      COMMON/ELPAR/NPAR(14),NUMNP,NELTYP,NUMEL,NUMNEL,NEC,NEQG,MBAND.
     .            NEQB,NBLOCK,MTOT,N1,N2,N3,N4,N5,N6,N7
      COMMCN/EM/ QQCQ(2043)
      COMMCN/MISC/ LG,LL,NFNR,NFA,NGM,NAT,NT,NOT,MSTEP
      CCMMON/JUNK/ JUK(205)
      COMMON A(1)
C
C     INPUT JOINT DATA----JOINT ID CODE ARRAY STORED ON TAPE 8
C
      N1=1
      N2=N1+NUMNP*6
      N3=N2+NUMNP
      N4=N3+NUMNP
      N5=N4+NUMNP
      IF (N5.GT.MTOT) CALL ERROR (1,N5-MTUT,7HINPUTM  )
      CALL INPUTM (A(N1),A(N2),A(N3),A(N4),NUMNP,NEC,NEGG,NGM)
C
C     INPUT ELEMENT DATA AND FORM ELEMENT STIFFNESSES--
C     STIFFNESS ON TAPE 2, STRESS ON TAPE 1, NONLINEAR DATA ON TAPE 3.
C
      N6=N5+NUMEL
      IF (N6.GT.HTOT) CALL ERROR (1,N6-MTOT,7HELSTF   )
      CALL ELSTF (A(N5),NUMEL)
C
C     CHECK AVAILABLE STORAGE AND CALCULATE SYSTEM EQUATION BLOCK SIZE
C
      LG=NECG-NEQ
      CALL BLOCEC (NEQB)
C
      NBLOCK=(NEQ-1)/NEQB+1
      IF (NEQB.GT.NEQ) NEQB=NEQ
      WRITE(6,2000) NEQ,MBAND,NEQB,NBLOCK
C
C     INPUT NODAL LCADS AND MASSES----ON TAPE 10
C
      N3=N2+NUMNP
      N4=N3+NUMNP*6
      N5=N4+NUMNP*6
      N6=N5+NECB
      N7=N6+NEQE
      IF (N7.GT.MTOT) CALL ERROR (1,N7-MTOT,7HINLM     )
      CALL INLM (A(N1),A(N2),A(N3),A(N4),A(N5),A(N6),NUMNP,NEQB,NEQ)
C
C     ASSEMBLE TOTAL STIFFNESS,LOAD---ON TAPE 4,  MASS---CN TAPE 9
C
      NE2B=2*NECB
      N2=N1+NE2B*MBAND
      N3=N2+NE2B
      N4=N3+NE2B
      N5=N4+NE2B*LG
      IF (N5.GT.MTOT) CALL ERROR (1,N5-MTOT,7HADDSTF   )
      CALL ADDSTF (A(N1),A(N2),A(N3),A(N4).NE2B,MBAND,LG,NUMEL,NEQ,
     .            NBLOCK)
      RETURN
C
 2000 FORMAT (/////28H SYSTEM SETUP INFORMATION...  ///
     .          30H TCTAL NUMBER OF EQUATIONS    =  I5//
     .          30H BAND WIDTH                   =  I5//
     .          30H NC OF EQUATIONS IN ONE BLCCK=  I5//
```

```
     .          30H TOTAL NUMBER OF BLOCKS       =  I5  )
      END



      SUBROUTINE INPUTM (ID,X,Y,Z,NUMNP,NEQ,NEQG,NGM)
C
      DIMENSION ID(NUMNP,6),X(1),Y(1),Z(1),NID(6)
C
C     NODAL POINT INPUT AND GENERATION
C
      NO=6
      WRITE(6,200)
      WRITE(6,204)
      KO=1
      DO 10 I=1,6
   10 NID(I)=0
   11 READ (5,100) N,(ID(N,I),I=1,6),X(N),Y(N),Z(N),KN
      WRITE(6,203) N,(ID(N,I),I=1,6),X(N),Y(N),Z(N),KN
      DO 30 I=1,6
      IF (ID(N,I)) 21,22,23
   21 IF (ID(N,I).EQ.-2) GO TO 25
      NID(I)=-1
      ID(N,I)=1
      GO TO 30
   22 IF (NID(I).EQ.-1) ID(N,I)=1
      GO TO 30
   23 NID(I)=0
      GO TO 30
   25 ID(N,I)=-1
   30 CONTINUE
C
C     CHECK IF GENERATION NEEDED THEN GENERATE NEW NODES
C
      IF (KO.EQ.1) GO TO 32
      IF (KN) 33,33,34
   32 KO=0
   33 CONTINUE
      NUMINT=1
      GO TO 45
   34 NUMINT=(N-NI)/KN
      DX=(X(N)- X(NI))/NUMINT
      DY=(Y(N)-Y(NI))/NUMINT
      DZ=(Z(N)-Z(NI))/NUMINT
      DO 41 J=1,NUMINT
      NN=NI+J*KN
      X(NN)=X(NN-KN)+DX
      Y(NN)=Y(NN-KN)+DY
      Z(NN)=Z(NN-KN)+DZ
C
C     SET DOF CODES SAME AS FIRST JOINT IN THE SERIES
C
      DO 40 JJ=1,6
      IF (ID(NI,JJ)-1) 37,36,35
   35 ID(NN,JJ)=ID(NI,JJ)+J*KN
      GO TO 40
   36 ID(NN,JJ)=ID(NI,JJ)
      GO TO 40
   37 IF (ID(NI,JJ).NE.-2) GO TO 38
      ID(NN,JJ)=ID(NI,JJ)
      GO TO 40
   38 ID(NN,JJ)=0
   40 CONTINUE
   41 CONTINUE
   45 NI=N
C
C     CHECK FOR LAST NODAL POINT
C
      IF (NUMNP-NI) 48,48,11
   48 CONTINUE
      WRITE(6,202)
      WRITE(6,204)
      DO 50 N=1,NUMNP
   50 WRITE(6,203) N,(ID(N,I),I=1,6),X(N),Y(N),Z(N)
C
C     COUNT AND NUMBER UNKNOWNS ---SET SLAVE NODE NEGATIVE OF MASTER
C
      NEQ=0
      DO 60 I=1,NUMNP
      DO 60 J=1,NO
      IF (ID(I,J).LT.0) GO TO 60
      IF (ID(I,J)-1) 57,58,59
   57 NEQ=NEQ+1
      ID(I,J)=NEQ
      GO TO 60
   58 ID(I,J)=0
      GO TO 60
   59 ID(I,J)=-ID(I,J)
   60 CONTINUE
C
C     ADD GROUND  BOUNDARY DOF
C
      NEQG=NEQ
      IF (NGM.NE.2) GO TO 90
      DO 80 I=1,NUMNP
      DO 80 J=1,NO
      IF (ID(I,J).NE.-1) GO TO 80
      NEQG=NEQG+1
      ID(I,J)=NEQG
   80 CONTINUE
C
C     PRINT INTERNAL EQUATION NUMBER
C
   90 WRITE(6,205)
      WRITE(6,206)
      WRITE(6,207) (N,(ID(N,I),I=1,6),N=1,NUMNP)
      REWIND 8
      WRITE (8) ID
      RETURN
C
  100 FORMAT (7I5,3F10.0,I5)
  200 FORMAT (31H1NODAL POINT DATA AS INPUT.....  //)
  202 FORMAT (31H1COMPLETE NODAL POINT DATA.....  //)
  203 FORMAT (I5,6I5,3F13.3,I5)
  204 FORMAT (5HONODE 6X 24HBOUNDARY CONDITION CODES 13X
     .  23HNODAL POINT COORDINATES /
     .  7H NUMBER 2X 1HX 4X 1HY 4X 1HZ 3X 2HXX 3X 2HYY 3X 2HZZ 12X 1HX
     .  12X 1HY 12X 1HZ  //)
  205 FORMAT (48H1EQUATION NUMBER OF NODAL DEGREES OF FREEDOM....  //)
  206 FORMAT (5HONODE 10X 15HEQUATION NUMBER 12X
     .          5H NODE 10X 15HEQUATION NUMBER  /
     .  7H NUMBER 2X 1HX 4X 1HY 4X 1HZ 3X 2HXX 3X 2HYY 3X 2HZZ  6X
     .  7H NUMBER 2X 1HX 4X 1HY 4X 1HZ 3X 2HXX 3X 2HYY 3X 2HZZ  //)
```

```
  207 FORMAT (7I5,6X,7I5)
      END



      SUBROUTINE ELSTF (IND,NEL)
C
      DIMENSION IND(NEL)
      COMMON/ELPAR/ NPAR(14),NUMNP,NELTYP,NUMEL,NUMNEL,NEQ,NEQG,MBAND,
     .               NEQB,NBLOCK,MTOT,N1,N2,N3,N4,N5,N6
C
C     FORM ELEMENT STIFFNESSES AND STRESS ARRAY FOR EACH ELEMENT TYPE
C
      REWIND 1
      REWIND 2
      REWIND 3
      NUML=NUMEL
      NUMEL=0
      NUMNEL=0
      MBAND=0
      DO 200 M=1,NELTYP
      READ (5,1001) NPAR
      MTYPE=NPAR(1)
C
      GO TO ( 1,2,3,4,5 ) MTYPE
C
C     THREE DIMENSIONAL TRUSS ELEMENTS
C
    1 CALL TRUSS
      GO TO 100
C
C     THREE DIMENSIONAL STRAIGHT BEAM ELEMENTS
C
    2 CALL BEAM
      GO TO 100
C
C     THREE DIMENSIONAL CURVED  BEAM ELEMENTS
C
    3 CALL CBEAM
      GO TO 100
C
C     THREE DIMENSIONAL BOUNDARY SPRING ELEMENTS
C
    4 CALL BOUND
      GO TO 100
C
C     EXPANSION JOINT FICTITIOUS ELEMENTS
C
    5 CALL EXPJE
C
  100 NUMEL=NUMEL+NPAR(2)
  200 CONTINUE
C
C     COUNT NUMBER OF NONLINEAR ELEMENTS
C
      IF (NUML.NE.NUMEL) CALL ERROR (2,NUMEL,THELSTF  )
      DO 300 I=1,NUMEL
      IF (IND(I).GT.0) NUMNEL=NUMNEL+1
  300 CONTINUE
      WRITE(6,2000) NUMNEL
      WRITE(6,2001)
      WRITE(6,2002) (I,IND(I),I=1,NUMEL)
      WRITE (8) IND
      RETURN
C
 1001 FORMAT (14I5)
 2000 FORMAT (1H1,38HTOTAL NUMBER OF NONLINEAR ELEMENTS   =  ,I5  ////)
 2001 FORMAT (1H ,35HELEMENT TYPE OF L/NL INDICATOR.....  //
     .          1H ,5(14H  ELEMENT IND )/1H ,5(14H  NUMBER  TYPE)/)
 2002 FORMAT (1H ,5(I7,I6,1X))
      END



      SUBROUTINE WRITET (MBAND,NDIF)
C
      COMMON/ELPAR/ NPAR(14),NUMNP,NELTYP,NUMEL,NUMNEL,NEQ,NEQG,MBAND,
     .               NEQB,NBLOCK,MTOT,N1,N2,N3,N4,N5,N6
      COMMON/EM/LM(24),ND,NS,S(24,24),P(24,4),XM(24),ST(12,24),TT(12,4)
C
C     CALCULATE BAND WIDTH AND WRITE ELEMENT INFORMATION ON TAPES
C
      MIN=100000
      MAX=0
      DO 450 L=1,ND
      IF (LM(L).EQ. 0 ) GO TO 450
      IF (LM(L).GT.NEQ) GO TO 450
      IF (LM(L).GT.MAX) MAX=LM(L)
      IF (LM(L).LT.MIN) MIN=LM(L)
  450 CONTINUE
      NDIF=MAX-MIN+1
      IF (NDIF.GT.MBAND) MBAND=NDIF
C
      WRITE (1) NS,ND,LM,ST,TT
      WRITE (2) LM,ND,NS,S,P,XM
C
      RETURN
      END



      SUBROUTINE BLCCEQ (NEQB)
C
      COMMON/ELPAR/ NPAR(14),NUMNP,NELTYP,NUMEL,NUMNEL,NEQ,NEQG,MBAND,
     .               NEQB,NBLOCK,MTOT,N1,N2,N3,N4,N5,N6
      COMMON/MISC/ LG,LL,NFNR,NFN,NGM,NAT,NT,NCT,NSTEP
      COMMON/JUNK/ N(10),NB,NSB,NSBB,NN
C
C     CHECK STORAGE FOR SETUP AND STATIC
C
      NEQB=(MTOT-1-13*NUMNP)/2
      NB=(MTOT-1)/(MBAND+LG+2)/2
      IF (NEQB.GT.NB) NEQB=NB
C
      LL=LG+1
   10 NSB=(MBAND+LL)*NEQB
      NSBB=NEQB*LL*(2+(MBAND-1)/NEQB)
      IF (NSBB.LT.NSB) NSBB=NSB
      NB=(MTOT-NSBB-1)/(MBAND+LL+1)
      IF (NEQB.LE.NB) GO TO 20
      NEQB=NB
```

```
SETU 294           GO TO 10
SETU 295   C
SETU 296        20 NB=(MTOT-6*NUMNP-6*LL-1)/LL
SETU 297           IF (NEQB.GT.NB) NEQB=NB
SETU 298   C
SETU 299   C       CHECK STORAGE FOR DYNAMIC LOAD INPUTS
SETU 300   C
SETU 301           NB=(MTOT-NFN*NAT-6*NUMNP-1)/(2*NFN)
SETU 302           IF (NEQB.GT.NB) NEQB=NB
SETU 303   C
SETU 304           IF (NGM-1) 60,30,40
SETU 305   C
SETU 306        30 NBLOCK=(NEQ-1)/NEQB+1
SETU 307           NB=(MTOT-NFN*NAT-6*NUMNP-1)/(2*NFN*NBLOCK+1)
SETU 308           GO TO 50
SETU 309        40 NB=(MTOT-NFN*NAT-6*NUMNP-3*LG-1)/(LG+1)
SETU 310   C
SETU 311        50 IF (NEQB.GT.NB) NEQB=NB
SETU 312   C
SETU 313        60 NB=(MTOT NFN*NAT-NAT-3*LG-1)/(NFN*NAT*2*NFN+LG)
SETU 314           IF (NEQB.GT.NB) NEQB=NB
SETU 315   C
SETU 316           M =2*NFN*NAT+NAT+NFN*NT+10*NFN*NAT+1
SETU 317           IF (M.GT.MTOT) CALL ERROR (1,M-MTOT,7HLOADH   )
SETU 318           NB=(MTOT-10*NFN*NAT-1)/(10+NFN*NAT)
SETU 319           IF (NEQB.GT.NB) NEQB=NB
SETU 320           NBLOCK=(NEQ-1)/NEQB+1
SETU 321           NB=(NEQ-1)/NBLOCK+1
SETU 322           IF (NEQB.GT.NB) NEQB=NB
SETU 323   C
SETU 324   C       CHECK STORAGE FOR STEP-BY-STEP SOLUTION IN CORE
SETU 325   C
SETU 326           NC=NEQB*NBLOCK
SETU 327           NN=1+NEQ*E+NUMNEL*26
SETU 328           N(1)=NC*(MBAND+LG)
SETU 329           N(2)=NN+2*NEQ*MBAND+3*NEQ*LG+2*LG+NFN*NAT+NUMEL
SETU 330           N(3)=NN+2*NEQ*MBAND+3*NEQ*LG+2*LG+NFN*NAT+NEQ
SETU 331           M=0
SETU 332           DO 100 I=1,3
SETU 333           IF (M.LT.N(I)) M=N(I)
SETU 334       100 CONTINUE
SETU 335   C
SETU 336           IF (MSTEP-1) 150,200,300
SETU 337   C
SETU 338       150 IF (M.GT.MTOT) GO TO 160
SETU 339           WRITE(6,2000) M
SETU 340           STOP
SETU 341       160 WRITE(6,2001) M
SETU 342           STOP
SETU 343   C
SETU 344       200 IF (M.GT.MTOT) CALL ERROR (1,M-MTOT,7HSTEP    )
SETU 345           WRITE(6,2000) M
SETU 346           GO TO 500
SETU 347   C
SETU 348       300 IF (M.GT.MTOT) GO TO 400
SETU 349           WRITE(6,2000) M
SETU 350           MSTEP=1
SETU 351           GO TO 500
SETU 352   C
SETU 353   C       CHECK STORAGE FOR STEP-BY-STEP SOLUTION IN BLOCKS
SETU 354   C
SETU 355       400 WRITE(6,2001) M
SETU 356           NB=(MTOT-NN-2*NEQ)/MBAND
SETU 357           IF (NEQB.GT.NB) NEQB=NB
SETU 358       410 NSB=NEQB*(MBAND+1)
SETU 359           NSBB=NEQB*(2+(MBAND-1)/NEQB)
SETU 360           IF (NSBB.LT.NSB) NSBB=NSB
SETU 361           NB=(MTOT-NN-NSBB)/(MBAND+2)
SETU 362           IF (NEQB.LE.NB) GO TO 420
SETU 363           NEQB=NB
SETU 364           GO TO 410
SETU 365   C
SETU 366       420 IF (LG) 450,450,430
SETU 367       430 NB=(MTOT-NN)/(MBAND+LG)/2
SETU 368           IF (NEQB.GT.NB) NEQB=NB
SETU 369       435 NSB=NEQB*(MBAND+LG)
SETU 370           NSBB=NEQB*LG*(2+(MBAND-1)/NEQB)
SETU 371           IF (NSBB.LT.NSB) NSBB=NSB
SETU 372           NB=(MTOT-NN-NSBB)/(MBAND+LG+1)
SETU 373           IF (NEQB.LE.NB) GO TO 440
SETU 374           NEQB=NB
SETU 375           GO TO 435
SETU 376   C
SETU 377       440 NB=(MTOT-NN-2*LG-NFN*NAT)/(2*LG)
SETU 378           IF (NEQB.GT.NB) NEQB=NB
SETU 379   C
SETU 380       450 MSTEP=2
SETU 381           NBLOCK=(NEQ-1)/NEQB+1
SETU 382           NB=(NEQ-1)/NBLOCK+1
SETU 383           IF (NEQB.GT.NB) NEQB=NB
SETU 384       500 RETURN
SETU 385   C
SETU 386      2000 FORMAT (55H1STORAGE CHECK OK FOR STEP-BY-STEP SOLUTION IN CORE....
SETU 387          .       //17H STORAGE NEEDED =  I8 )
SETU 388      2001 FORMAT (40H1FOR STEP-BY-STEP SOLUTION IN CORE......
SETU 389          .           28H MTOT SHOULD BE INCREASED TO   I8)
SETU 390           END


SETU 391           SUBROUTINE INLM (ID,IN,TL,TM,B,TMASS,NUMNP,NEQB,NEQ)
SETU 392   C
SETU 393           DIMENSION ID(NUMNP,6),IN(NUMNP),TL(NUMNP,6),TM(NUMNP,6),B(NEQB),
SETU 394          .          TMASS(NEQB)
SETU 395           COMMON/JUNK/ NT,KSHF,NE,N,NN,II,JJ,R(6)
SETU 396   C
SETU 397           NT=10
SETU 398           REWIND 10
SETU 399           KSHF=0
SETU 400           DO 5 I=1,NUMNP
SETU 401           IN(I)=0
SETU 402           DO 5 J=1,6
SETU 403           TL(I,J)=0.
SETU 404         5 TM(I,J)=0.
SETU 405           DO 10 I=1,NEQB
SETU 406           B(I)=0.
SETU 407        10 TMASS(I)=0.
SETU 408   C
SETU 409   C       INPUT NODAL MASSES
SETU 410   C
SETU 411           NE=0
SETU 412       100 READ (5,1001) N,R
SETU 413           NE=NE+1
```

```
SETU 414          IF (N.EQ.0) GO TO 150
SETU 415          IF (NE.EQ.1) WRITE(6,2003)
SETU 416          WRITE(6,2001) N,R
SETU 417          DO 140 J=1,6
SETU 418      140 TM(N,J)=TM(N,J)+R(J)
SETU 419          IN(N)=1
SETU 420          GO TO 100
SETU 421    C
SETU 422    C     INPUT NODAL LOADS
SETU 423    C
SETU 424      150 NE=0
SETU 425      200 READ (5,1001) N,R
SETU 426          NE=NE+1
SETU 427          IF (N.EQ.0) GO TO 300
SETU 428          IF (NE.EQ.1) WRITE(6,2002)
SETU 429          WRITE(6,2001) N,R
SETU 430          DO 240 J=1,6
SETU 431      240 TL(N,J)=TL(N,J)+R(J)
SETU 432          IN(N)=1
SETU 433          GO TO 200
SETU 434    C
SETU 435    C     ASSEMBLE LOADS AND MASSES IN BLOCKS
SETU 436    C
SETU 437      300 DO 900 NN=1,NUMNP
SETU 438          IF (IN(NN).EQ.0) GO TO 900
SETU 439          DO 800 J=1,6
SETU 440          II=ID(NN,J)
SETU 441          IF (II) 800,800,400
SETU 442      400 IF (II.GT.NEQ) GO TO 800
SETU 443      500 JJ=II-KSHF
SETU 444          IF (JJ.LE.NEQB) GO TO 700
SETU 445          WRITE (NT) B,TMASS
SETU 446          KSHF=KSHF+NEQB
SETU 447          DO 600 I=1,NEQB
SETU 448          B(I)=0.
SETU 449      600 TMASS(I)=0.
SETU 450          GO TO 500
SETU 451      700 B(JJ)=TL(NN,J)
SETU 452          TMASS(JJ)=TM(NN,J)
SETU 453      800 CONTINUE
SETU 454      900 CONTINUE
SETU 455    C
SETU 456          WRITE (NT) B,TMASS
SETU 457          RETURN
SETU 458    C
SETU 459     1001 FORMAT (I5,5X,6F10.0)
SETU 460     2001 FORMAT (I5,5X,6F12.3)
SETU 461     2002 FORMAT (23H1NODAL POINT LOADS.....   // 10H NODE        27X
SETU 462         .  14HAPPLIED LOADS                /  10H  NO.        8X 2HRX 10X
SETU 463         .    2HRY 10X 2HRZ 10X 2HMX 10X 2HMY 10X 2HMZ )
SETU 464     2003 FORMAT (23H1NODAL POINT MASSES....  // 10H NODE        27X
SETU 465         .  19HCONCENTRATED MASSES          /  10H  NO.        8X 2HMX 10X
SETU 466         .    2HMY 10X 2HMZ 10X 3HI/X 9X 3HI/Y 9X 3HI/Z )
SETU 467          END


SETU 468          SUBROUTINE ADDSTF (A,B,TMASS,C,NE2B,MBAND,LG,NUMEL,NEQ,NBLOCK)
SETU 469    C
SETU 470          DIMENSION A(NE2B,MBAND),B(NE2B),TMASS(NE2B),C(NE2B,LG),STIF(722)
SETU 471          COMMON/EM/ LM(24),ND,NS,S(24,24),P(24,4),XM(24),ST(12,24),TT(12,4)
```

```
SETU 472          COMMON/JUNK/ STR(4),K,MB,NEBB,X,NSHIFT,LMI,LMN,NUME,NUM7,II,JJ,MM
SETU 473          EQUIVALENCE (STIF,LM)
SETU 474    C
SETU 475    C     FORM GLOBAL EQUILIBRIUM EQUATIONS IN BLOCKS---2 BLOCKS AT A TIME
SETU 476    C
SETU 477          NEQB=NE2B/2
SETU 478          K=NEQB+1
SETU 479          X=NBLOCK
SETU 480          MB=SQRT(X)
SETU 481          MB=MB/2+1
SETU 482          NEBB=MB*NE2B
SETU 483          NSHIFT=0
SETU 484          NUM7=0
SETU 485          MM=1
SETU 486          L=1
SETU 487    C
SETU 488          REWIND 4
SETU 489          REWIND 9
SETU 490          REWIND 10
SETU 491          DO 40 I=1,4
SETU 492       40 STR(I)=0.
SETU 493          WRITE(6,2000)
SETU 494          READ (5,1001) STR
SETU 495          WRITE(6,2001) L,(STR(I),I=1,4)
SETU 496    C
SETU 497          DO 1000 M=1,NBLOCK,2
SETU 498          DO 100 I=1,NE2B
SETU 499          DO 80 J=1,MBAND
SETU 500       80 A(I,J)=0.
SETU 501          IF (LG.EQ.0) GO TO 100
SETU 502          DO 90 J=1,LG
SETU 503       90 C(I,J)=0.
SETU 504      100 CONTINUE
SETU 505          READ (10) (B(I),I=1,NEQB),(TMASS(I),I=1,NEQB)
SETU 506          IF (M.EQ.NBLOCK) GO TO 200
SETU 507          READ (10) (B(I),I=K,NE2B),(TMASS(I),I=K,NE2B)
SETU 508      200 CONTINUE
SETU 509    C
SETU 510          REWIND 7
SETU 511          REWIND 2
SETU 512          NA=7
SETU 513          NUME=NUM7
SETU 514          IF (MM.NE.1) GO TO 250
SETU 515          NA=2
SETU 516          NUME=NUMEL
SETU 517          NUM7=0
SETU 518      250 DO 700 N=1,NUME
SETU 519          READ (NA) STIF
SETU 520          DO 600 I=1,ND
SETU 521          LMI=LM(I)
SETU 522          IF (LMI.GT.NEQ) GO TO 600
SETU 523          LMN=I-LMI
SETU 524          II=LMI-NSHIFT
SETU 525          IF ((II.LE.0).OR.(II.GT.NE2B)) GO TO 600
SETU 526          TMASS(II)=TMASS(II)+XM(I)
SETU 527          DO 300 J=1,4
SETU 528      300 B(II)=B(II)+P(I,J)*STR(J)
SETU 529          DO 500 J=1,ND
SETU 530          JJ=LM(J)
SETU 531          IF (JJ.LE.NEQ) GO TO 350
SETU 532          L=JJ-NEQ
SETU 533          C(II,L)=C(II,L)+S(I,J)
```

```
      GO TO 500
  350 JJ=JJ+LPN
      IF (JJ) 500,500,400
  400 A(II,JJ)=A(II,JJ)+S(I,J)
  500 CONTINUE
  600 CONTINUE
C
C     DETERMINE IF ELEMENT INFORMATION IS TO BE PLACED ON TEMPORARY TAPE
C
      IF (MM.GT.1) GO TO 700
      DO 650 I=1,ND
      LMI=LM(I)
      IF (LMI.GT.NEQ) GO TO 650
      II=LMI-NSHIFT
      IF((II.GT.NE2B).AND.(II.LE.NEBA)) GO TO 660
  650 CONTINUE
      GO TO 700
  660 WRITE (7) STIF
      NUM7=NUM7+1
  700 CONTINUE
C
      IF (LG) 800,800,900
  800 WRITE (4) ((A(I,J),I=1,NEQB),J=1,MBAND),(B(I),I=1,NEQB)
      WRITE (9) (TMASS(I),I=1,NEQB)
      IF (M.EQ.NBLOCK) GO TO 1000
      WRITE (4) ((A(I,J),I=K,NE2B),J=1,MBAND),(B(I),I=K,NE2B)
      WRITE (9) (TMASS(I),I=K,NE2B)
      GO TO 950
C
  900 WRITE (4) ((A(I,J),I=1,NEQB),J=1,MBAND),((C(I,L),I=1,NEQB),L=1,LG)
     .              ,(B(I),I=1,NEQB)
      WRITE (9) (TMASS(I),I=1,NEQB)
      IF (M.EQ.NBLOCK) GO TO 1000
      WRITE (4) ((A(I,J),I=K,NE2B),J=1,MBAND),((C(I,L),I=K,NE2B),L=1,LG)
     .              ,(B(I),I=K,NE2B)
      WRITE (9) (TMASS(I),I=K,NE2B)
C
  950 IF (MM.EQ.MB) MM=0
      MM=MM+1
 1000 NSHIFT=NSHIFT+NE2B
C
      RETURN
C
 1001 FORMAT (4F10.0)
 2000 FORMAT (////49H ELEMENT LOAD MULTIPLIER FOR STATIC ANALYSIS.....//
     .  10H STRUCTURE  6X 24H ELEMENT LOAD MULTIPLIER /
     .  10H LOAD CASE 7X 1HA 9X 1HB 9X 1HC 9X 1HD //)
 2001 FORMAT (I6,5X,4F10.3)
      END
```

```
      SUBROUTINE STATIC
C
      COMMON/ELPAR/NPAR(14),NUMNP,NELTYP,NUMEL,NUMKEL,NEW,NEWG,MBAND,
     .                 NEQB,NBLOCK,MTOT,N1,N2,N3,N4,N5,N6,N7
      COMMON/EM/ GCCQ(2043)
      COMMON/MISC/ LG,LL,NFAR,NFN,NGM,NAT,NT,NOT,MSTEP
      COMMON/JUNK/ JUK(205)
      COMMON A(1)
C
C     COMPUTE STATIC RESPONSE---SOLVE FOR DISPLACEMENT UNKNOWNS
C
      LL=LG+1
      NSB=(MBAND+LL)*NEQB
      NSBB=NEQB*LL*(2+(MBAND-1)/NEQB)
      IF (NSBB.LT.NSB) NSBB=NSB
      N2=N1+NEQB*MBAND
      N3=N2+NEQB*LL
      N4=N3+NSBB
      N5=N4+NEQB
      IF (N5.GT.MTOT) CALL ERROR (1,N5-MTOT,7HUSOL      )
      CALL USOL (A(N1),A(N3),A(N4),NEQB,MBAND,LL,NBLOCK,NSB,4,10,7,2,2)
C
C     OUTPUT STATIC DISPLACEMENTS
C
      N2=N1+6*NUMNP
      N3=N2+6*LL
      N4=N3+NEQB*LL
      CALL PRINTD (A(N1),A(N2),A(N3),NUMNP,NEQB,LL,NBLOCK,NEQ,2)
C
C     OUTPUT STATIC STRESSES OF EACH ELEMENT
C
      N2=N1+NEQB*LL
      N3=N2+NEQ
      N4=N3+NUMEL
      CALL STRESS (A(N1),A(N2),A(N3),NEQB,LL,NEQ,NUMEL)
C
      RETURN
      END
```

```
      SUBROUTINE PRINTD (ID,D,B,NUMNP,NEQB,LL,NBLOCK,NEQ,NT)
C
      DIMENSION ID(NUMNP,6),D(6,LL),B(NEQB,LL)
      COMMON/JUNK/ STR(4),M,LG,III,JJJ,I,K,S,PS,NID,NQ,N
C
C     PRINT DISPLACEMENTS OF STATIC ANALYSIS
C
      REWIND NT
      REWIND 8
      READ (8) ID
      M=NEQ
      NC=NEQB*NBLOCK
      LG=LL-1
      IF (LG.EQ.0) WRITE(6,2003)
      IF (LG.GT.0) WRITE(6,2005)
      KKK=0
      N=NUMNP
      S =3H SD
      PS=3HPSD
```

```
C
      DO 500 NK=1,NUMNP
      III=6
      JJJ=6
      I=0
      DO 250 II=1,6
      DO 100 L=1,LL
  100 D(I,L)=0.
      IF (M.GT.NG) GO TO 150
      IF (M.EQ.0 ) GO TO 150
      READ (NT) B
      NG=NG-NEQB
  150 NID=ID(N,I)
      IF (NID.EQ. 0 ) JJJ=JJJ-1
      IF (NID.LT. 0 ) III=III-1
      IF (NID.GT.NEQ) GO TO 240
      IF (NID.LT. 1 ) GO TO 250
      K=M-NG
      M=M-1
      DO 200 L=1,LL
  200 D(I,L)=B(K,L)
      GO TO 250
  240 LF=NID-NEQ
      D(I,LF)=1.0
  250 I=I+1
C
      IF (III.EQ.0) GO TO 400
      IF (JJJ.EQ.0) GO TO 480
      WRITE(6,2004) N,S,(D(I,LL),I=1,6)
      KKK=1
  300 IF (LG.EQ.0) GO TO 500
      WRITE(6,2008)  (PS,(D(I,L),I=1,6),L=1,LG)
      KKK=1
      GO TO 500
  400 WRITE(6,2006) N
      GO TO 500
  450 WRITE(6,2007) N
      GO TO 500
  480 IF (KKK.EQ.0) GO TO 500
      WRITE(6,2009) N,S,(D(I,L),I=1,6)
      KKK=0
      GO TO 300
  500 N=N-1
C
      RETURN
C
 2003 FORMAT (35H1PRINT OF STATIC DISPLACEMENTS.....  //
     .  6X 25H SD---STATIC DISPLACEMENT //
     .  5H NODE 5H TYPE 11X 1HX 11X 1HY 11X 1HZ 9X 2HXX 9X 2HYY 9X 2HZZ/
     .  5H   NO 5H CODE /)
 2004 FORMAT (1H ,I4,2X,A3,1P3E12.3,3E11.2)
 2005 FORMAT (53H1PRINT OF STATIC DISPL. AND PSEUDO-STATIC COEFF..... //
     .  6X 25H SD --STATIC DISPLACEMENT /
     .  5X 32H PSD--PSEUDO-STATIC DISPLACEMENT //
     .  5H NODE 5H TYPE 11X 1HX 11X 1HY 11X 1HZ 9X 2HXX 9X 2HYY 9X 2HZZ/
     .  5H   NO 5H CODE /)
 2006 FORMAT (1H0,I4,8X,18H...SLAVE NODE .... )
 2007 FORMAT (1H0,I4,8X,26H...FIXED NODE ...(ALL 0. ) )
 2008 FORMAT (1H0,6X,A3,1P3E12.3,3E11.2 /(7X,A3,3E12.3,3E11.2) )
 2009 FORMAT (1H ,I4,2X,A3,1P3E12.3,3E11.2)
      END
```

```
      SUBROUTINE STRESS (B,D,IND,NEQB,LL,NEQ,NUMEL)
C
      DIMENSION B(NEQB,LL),D(NEQ),IND(NUMEL)
      COMMON/EM/ ND,NS,LM(24),S(24,24),P(24,4),XM(24),SA(12,24),TT(12,4)
      COMMON/ELPAR/ NPAR(14),NUMNP,NELTYP,MUMEL,NUMMEL,MEQ,NEQG,MBAND,
     .                 MEQB,NBLOCK,MTOT,NNN(6)
      COMMON/JUNK/ STR(4),MM,L,K,NTAG,NDYN,SIG(12),EXRA(184)
C
C     COMPUTE AND PRINT STATIC STRESSES OF EACH ELEMENT
C
      NT=2
      REWIND NT
      READ (8) IND
      NQ=NEQB*NBLOCK
      DO 200 NN=1,NBLOCK
      READ (NT) B
      N=NEQB
      IF (NN.EQ.1) N=NEQB-(NQ-NEQ)
      NQ=NQ-NEQB
      DO 200 J=1,N
      I=NQ+J
  200 D(I)=B(J,LL)
C
      REWIND 1
      L=1
      NTYPE=0
      MM=1
C
      DO 1000 MN=1,NUMEL
      MTYPE=IND(MN)
      IF (MTYPE.LT.0) MTYPE=-MTYPE
      IF (NTYPE.EQ.MTYPE) GO TO 250
      WRITE(6,2000)
      NPAR(1)=0
      NTAG=0
      NTYPE=MTYPE
      MM=1
  250 READ (1) NS,ND,LM,SA,TT
C
C     COMPUTE STRESS FROM DISPLACEMENT AND ADD LOCAL STRESS
C
      DO 300 N=1,NS
      SIG(N)=0.
      DO 280 I=1,4
  280 SIG(N)=SIG(N)+TT(N,I)*STR(I)
      DO 300 J=1,ND
      JJ=LM(J)
      IF (JJ.GT.NEQ) GO TO 300
      IF (JJ) 300,300,290
  290 SIG(N)=SIG(N)+SA(N,J)*D(JJ)
  300 CONTINUE
C
C     PRINT STRESSES FOR EACH ELEMENT
C
      GO TO (1,2,3,4,5) MTYPE
C
    1 CALL TRUSS
      GO TO 900
C
```

```
    2 CALL BEAM
      GO TO 900
C
    3 CALL CBEAM
      GO TO 900
C
    4 CALL BOUND
      GO TO 900
C
    5 CALL EXPJP
C
  900 MM=MM+1
      WRITE (8) NS,SIG
 1000 CONTINUE
      RETURN
C
 2000 FORMAT (27H1 STATIC STRESS OUTPUT.....  )
      END
```

```
      SUBROUTINE USOL (A,B,MAXB,NEQB,MB,LL,NBLOCK,NSB,NORG,NBKS,NT1,
     .                     NT2,NKST)
      DIMENSION A(NSB),B(NSB),MAXB(NEQB)
C
      NC=MB+LL
      NBR=(MB-1)/NEQB+1
      INC=NEQB-1
      NMB=NEQB*MB
      N2=NT2
      N1=NT1
      REWIND NORG
      REWIND NBKS
C
C     REDUCE EQUATIONS  BLOCK-BY-BLOCK
C
      DO 900 N=1,NBLOCK
      IF (N.GT.1.AND.NBR.EQ.1) GO TO 110
      IF (NBR.EQ.1) GO TO 105
      REWIND N1
      REWIND N2
  105 NI=N1
      IF(N.EQ.1) NI=NORG
      READ (NI) A
  110 DO 300 I=1,NEQB
      D=A(I)
      IF(D) 115,300,120
  115 M=NEQB*(N-1)+I
      WRITE(6,2000) M,D
C
  120 II=I
      DO 125 J=2,NC
      II=II+NEQB
  125 A(II)=A(II)/D
      DO 130 J=I,NMB,NEQB
      IF (A(J).NE.0.) MAXB(I)=J
  130 CONTINUE
C
      JL=I+1
      IF (JL.GT.NEQB) GO TO 300
      II=I
      DO 200 J=JL,NEQB
      II=II+NEQB
      IF(II.GT.NMB) GO TO 200
      C=A(II)
      IF (C.EQ.0.0) GO TO 200
      C=C*A(I)
C
      KK=J
      MAX=MAXB(I)
      DO 150 JJ=II,MAX,NEQB
      A(KK)=A(KK)-C*A(JJ)
  150 KK=KK+NEQB
C
      KK=J +NMB
      JJ=I+NMB
      DO 175 L=1,LL
      A(KK)=A(KK)-C*A(JJ)
      KK=KK+NEQB
  175 JJ=JJ+NEQB
  200 CONTINUE
  300 CONTINUE
```

```
USOL  62          WRITE (NBKS) A,MAXB
USOL  63   C
USOL  64   C      SUBSTITUTE INTO REMAINING EQUATIONS
USOL  65   C
USOL  66          DO 800 NN=1,NBR
USOL  67          IF(N+NN.GT.NBLOCK) GO TO 900
USOL  68          NI=N1
USOL  69          IF(N.EQ.1) NI=NORG
USOL  70          IF(NN.EQ.NBR) NI=NORG
USOL  71          READ (NI) B
USOL  72          IL=1+NN*NECB*NEQB
USOL  73          DO 700 I=1,NECB
USOL  74          II=IL
USOL  75          DO 690 K=1,NECB
USOL  76          IF (II.GT.NMB) GO TO 690
USOL  77          C=A(II)
USOL  78          IF (C.EQ.0.0) GO TO 690
USOL  79          C=C*A(K)
USOL  80          MAX=MAXB(K)
USOL  81   C
USOL  82          KK=1
USOL  83          DO 640 JJ=II,MAX,NEQB
USOL  84          B(KK)=B(KK)-C*A(JJ)
USOL  85      640 KK=KK+NEQB
USOL  86          KK=1+NMB
USOL  87          JJ=K+NMB
USOL  88          DO 650 L=1,LL
USOL  89          B(KK)=B(KK)-C*A(JJ)
USOL  90          KK=KK+NEQB
USOL  91      650 JJ=JJ+NEQB
USOL  92      690 II=II-INC
USOL  93      700 IL=IL+NECB
USOL  94   C
USOL  95          IF(NBR.NE.1) GO TO 750
USOL  96          DO 740 I=1,NSB
USOL  97      740 A(I)=B(I)
USOL  98          GO TO 800
USOL  99      750 WRITE (N2) B
USOL 100      800 CONTINUE
USOL 101          M=N1
USOL 102          N1=N2
USOL 103      900 N2=M
USOL 104   C
USOL 105   C      BACKSUBSTITUTION - RESULTS ON TAPE NRST
USOL 106   C
USOL 107          LS=LL*NECB
USOL 108          NEB=NECB*(NBR+1)
USOL 109          NUM=NBR*NEQB
USOL 110          MAX=NEB*LL
USOL 111          DO 905 I=1,MAX
USOL 112      905 B(I)=0.
USOL 113          REWIND NRST
USOL 114   C
USOL 115          DO 1000 N=1,NBLOCK
USOL 116          BACKSPACE NBKS
USOL 117          READ (NBKS) A,MAXB
USOL 118          BACKSPACE NBKS
USOL 119          DO 910 L=1,LL
USOL 120          K=L*NEB
USOL 121          DO 910 J=1,NLP
USOL 122          I=K-NEQB
USOL 123          B(K)=B(I)
USOL 124      910 K=K-1
USOL 125   C
USOL 126          I=NMB
USOL 127          DO 920 L=1,LL
USOL 128          K=(L-1)*NEB
USOL 129          DO 920 J=1,NECB
USOL 130          I=I+1
USOL 131          K=K+1
USOL 132      920 B(K)=A(I)
USOL 133   C
USOL 134          DO 955 I=1,NECB
USOL 135          J=NECB+1-I
USOL 136          MAX=MAXB(J)
USOL 137          IF (A(J).EQ.0.) GO TO 955
USOL 138          DO 950 L=1,LL
USOL 139          KK=J+(L-1)*NEB
USOL 140          JJ=KK+1
USOL 141          IL=J+NEQB
USOL 142          C=B(KK)
USOL 143          DO 940 II=IL,MAX,NEQB
USOL 144          C=C-A(II)*B(JJ)
USOL 145      940 JJ=JJ+1
USOL 146      950 B(KK)=C
USOL 147      955 CONTINUE
USOL 148   C
USOL 149          I=0
USOL 150          DO 960 L=1,LL
USOL 151          K=(L-1)*NEB
USOL 152          DO 960 J=1,NECB
USOL 153          K=K+1
USOL 154          I=I+1
USOL 155      960 A(I)=B(K)
USOL 156   C
USOL 157          WRITE (NRST) (A(I),I=1,LS)
USOL 158     1000 CONTINUE
USOL 159          RETURN
USOL 160   C
USOL 161     2000 FORMAT (33H0SET OF EQUATIONS MAY BE SINGULAR   /
USOL 162         *  26H DIAGONAL TERM OF EQUATION   I6, 8H  EQUALS   1PE12.4)
USOL 163          END
```

```
      SUBROUTINE LOADS
C
      COMMON/ELPAR/ NPAR(14),NUMNP,NELTYP,NUMEL,NUMNEL,NEQ,NEQG,MBAND,
     .                NEQB,NBLOCK,MTOT,N1,N2,N3,N4,N5,N6,N7
      COMMON/EM/ AT(2043)
      COMMON/MISC/ LG,LL,NFNR,NFN,NGM,NAT,NT,NOT,MSTEP
      COMMON/JUNK/ NARB,DT,ALFA,BETA,INTG,JUK(200)
      COMMON A(1)
C
C     READ AND PRINT LOAD INPUT CONTROLL DATA
C
      READ (5,1000) DT,ALFA,BETA,INTG
      WRITE(6,2000) DT,NT,NOT,NFN,NAT,NGM
      IF (NAT.EQ.0) NAT=1
      WRITE(6,2001) INTG,ALFA,BETA
C
C     INPUT DYNAMIC LOADS AT NODAL POINTS
C
      N1=1
      N2=N1+NFN*NAT
      N3=N2+NUMNP*6
      N4=N3+NEQB*NFN
      N5=N4+NEQB*NFN
      IF (N5.GT.MTOT) CALL ERROR (1,N5-MTOT,7HINDYL   )
      CALL INDYL (A(N1),A(N2),A(N3),A(N4),NEQB,NFN,NAT,NUMNP,NEQ)
C
      IF (NGM-1) 400,200,300
C
C     INPUT RIGID GROUND MOTION IN 3 DIRECTIONS
C
  200 N6=N5+NEQB
      N7=N6+NEQB*NBLOCK
      IF (N7.GT.MTOT) CALL ERROR (1,N7-MTOT,7HINRGM   )
      CALL INRGM (A(N1),A(N2),A(N3),A(N4),A(N5),A(N6),NFN,NUMNP,
     .                NEQB,NEQ,NBLOCK)
      GO TO 400
C
C     INPUT MULTIPLE GROUND MOTION PRESCRIBED AT EACH SUPPORTING POINTS
C
  300 N4=N3+NEQB*LG
      N5=N4+LG
      N6=N5+LG
      N7=N6+LG
      IF (N7.GT.MTOT) CALL ERROR (1,N7-MTOT,7HINMGM   )
      CALL INMGM (A(N1),A(N2),A(N3),A(N4),A(N5),A(N6),
     .                NFN,NAT,NUMNP,NEQB,LG,NEQ,NBLOCK    )
C
C     INPUT DELAY TIME OF LOAD FUNCTIONS
C
  400 N3=N2+NAT
      N4=N3+NAT
      N5=N4+NAT
      CALL INDLT (A(N1),A(N2),A(N3),A(N4),NFN,NAT,NFNR)
C
C     SET UP DYNAMIC LOAD MATRIX
C
      N4=N3+NEQB*NFNR
      N5=N4+NEQB*NFN
      N6=N5+NEQB*NFN
      N7=N6+LG
      N8=N7+LG
      N9=N8+LG
      N10=N9+NEQB*LG
      N11=N10+NEQB
      IF (N11.GT.MTOT) CALL ERROR (1,N11-MTOT,7HLOADM   )
      CALL LOADM (A(N1),A(N2),A(N3),A(N4),A(N5),A(N6),A(N7),A(N8),A(N9),
     .                A(N10),NFN,NAT,NFNR,NEQB,LG,NGM,NBLOCK)
C
C     INPUT TIME HISTORIES OF LOAD FUNCTION
C
      N4=N3+NFN*NT
      MAX=(MTOT-N4)/2
      IF (MAX.GE.NT) GO TO 440
      N5=2*(NT-MAX)
      CALL ERROR (1,N5,7HINHIS   )
  440 N5=N4+MAX
      CALL INHIS (A(N3),A(N4),A(N5),NFN,NT,MAX)
C
C     SET UP LOAD HISTORIES INCLUDING DELAY TIME EFFECT
C
      NTB=(MTOT-N4-NFNR)/NFNR
      NTBB=(MTOT-NEQB*NFNR)/(NEQB+NFNR)
      IF (NTB.GT.NTBB) NTB=NTBB
      NTBB=(MTOT-1)/(NEQB*NBLOCK)
      IF (NTB.GT.NTBB) NTB=NTBB
C
      IF (NTB.GT.NT) NTB=NT
      IF (NTB.GE.10) GO TO 450
      N5=N4+NFNR*10-MTOT
      N6=NEQB*NFNR+10*(NEQB+NFNR)-MTOT
      N7=NEQB*NBLOCK*10-MTOT
      IF (N5.LT.N6) N5=N6
      IF (N5.LT.N7) N5=N7
      CALL ERROR (1,N5,7HLOADH   )
C
  450 NB=(NT-1)/NTB+1
      N5=N4+NFNR*NTB
      N6=N5+NFNR
      CALL LOADH (A(N1),A(N2),A(N3),A(N4),A(N5),
     .                NFN,NAT,NFNR,NT,NTB,NB,NGM)
C
C     FORM DYNAMIC NODAL LOAD VECTOR
C
      N2=N1+NEQB*NTB
      N3=N2+NEQB*NFNR
      N4=N3+NFNR*NTB
      IF (N4.GT.MTOT) CALL ERROR (1,N4-MTOT,7HLOADV   )
      NQ=NEQB*NBLOCK
      N5=N1+NQ*NTB
      IF (N5.GT.MTOT) CALL ERROR (1,N5-MTOT,7HLOADV   )
      CALL LOADV (A(N1),A(N2),A(N3),A(N1),NEQB,NFNR,NTB,
     .                NBLOCK,NB,NT,NEQ,NQ)
C
      RETURN
 1000 FORMAT (3F10.0,I5)
 2000 FORMAT (37H1DYNAMIC LOAD INPUT CONTROL DATA..... ///
     .  32H  TIME INCREMENT DT   (SEC)     =  F8.3//
     .  32H  TOTAL NUMBER OF TIME STEPS    =  I5  //
     .  32H  OUTPUT INTERVAL               =  I5  //
     .  32H  NUMBER OF TIME FUNCTIONS      =  I5  //
     .  32H  NUMBER OF DELAY TIMES         =  I5  //
     .  32H  GROUND MOTION INDICATOR       =  I5     )
 2001 FORMAT (////48H STEP-BY-STEP DYNAMIC ANALYSIS CONTROL DATA.....//L-
```

```
     . 32H  INTEGRATION INDICATOR            =  I5  //
     . 32H  DAMPING FACTOR  ALPHA            =  1PE12.3//
     . 32H  DAMPING FACTOR  BETA             =  1PE12.3   )
C
      END
```

```
      SUBROUTINE INCYL (IOF,ID,FF,IFF,NEQB,NFN,NAT,NUMNP,NEQ)
C
      DIMENSION IOF(NFN,NAT),ID(NUMNP,6),FF(NEQB,NFN),IFF(NEQB,NFN)
      COMMON/JUNK/ NARB,DATA(4),NFAT,NNN,NP,IC,IFN,IAT,P,NN,II,NS,NE
C
C     INPUT ARBITRARY DYNAMIC LOADS AT NODAL POINTS
C
      MT=14
      REWIND MT
      REWIND 8
      READ (8) ID
      NFAT=NFN*NAT
      DO 5 I=1,NFAT
    5 IOF(I)=0
      NNN=NEQB*NFN
      DO 10 I=1,NNN
      FF(I)=0.
   10 IFF(I)=0
C
      NARB=1
      READ (5,1000) NP,IC,IFN,IAT,P
      IF (IAT.EQ.0) IAT=1
      IF (NP.GT.0) GO TO 50
      NARB=0
      RETURN
C
   50 WRITE(6,2000)
      WRITE(6,2002) NP,IC,IFN,IAT,P
      IF (IFN.GT.0) IOF(IFN,IAT)=1
C
      NS=1
      NE=NEQB
      DO 500 NN=1,NUMNP
      DO 500 II=1,6
      N=ID(NN,II)
      IF (N.GT.NEQ) GO TO 300
      IF (N.LE. 0 ) GO TO 300
      IF (N.GE.NS ) GO TO 100
      CALL ERROR (7,NP,7HINDYL   )
C
  100 IF (N.LE.NE ) GO TO 300
      WRITE (MT) FF,IFF
      DO 150 I=1,NNN
      FF(I)=0.
      IFF(I)=0
  150 CONTINUE
      NS=NS+NEQB
      NE=NE+NEQB
      GO TO 100
C
  300 IF ((NP.EQ.NN).AND.(IC.EQ.II)) GO TO 350
      GO TO 500
  350 IF (N.LE. 0 ) GO TO 400
      IF (N.GT.NEQ) GO TO 400
      M=N-NS+1
      IF (IFN.LE.0) GO TO 400
      FF(M,IFN)=P
      IFF(M,IFN)=IAT
  400 READ (5,1000) NP,IC,IFN,IAT,P
      IF (IAT.EQ.0) IAT=1
      IF (NP.LE.0) GO TO 300
      WRITE(6,2002) NP,IC,IFN,IAT,P
      IF (IFN.GT.0) IOF(IFN,IAT)=1
      GO TO 300
  500 CONTINUE
      WRITE (MT) FF,IFF
      RETURN
C
 1000 FORMAT (4I5,F10.0)
 2000 FORMAT (24H1DYNAMIC LOAD INPUT.....  //
     .  57H  NODE   DISPLACEMENT  FUNCTION   ARRIVAL TIME    FUNCTION /
     .  59H   NO.  - COMPONENT     NUMBER        NUMBER       MULTIPLIER/)
 2002 FORMAT (I6,2I11,I14,F15.3)
C
      END
```

```
      SUBROUTINE INRGM (IOF,ID,FF,IFF,XM,MASS,NFN,NUMNP,NEQB,NEQ,NBLOCK)
C
      DIMENSION IOF(NFN,1),ID(NUMNP,1),FF(NEQB,NFN),IFF(NEQB,NFN),
     .              XM(NEQB),MASS(1)
      COMMON/JUNK/ NARB,DATA(4),JFN(3),PFN(3),JAT(3),JT,MT,KU,L,II,JJ
C
C     GENERATE PSEUDO-STATIC COEFFICIENTS FOR RIGID GROUND MOTION INPUT
C
      JT=9
      KT=10
      MT=14
      NT=15
      REWIND JT
      REWIND KT
      REWIND MT
      REWIND NT
      NC=NEQB*NBLOCK
      DO 5 I=1,NC
    5 MASS(I)=0
C
      L=1
      DO 50 N=1,NUMNP
      DO 40 I=1,6
      II=ID(N,I)
      IF (II.LE.0   ) GO TO 40
      IF (II.GT.NEQ) GO TO 40
      IF (I  .GT.3   ) GO TO 30
      MASS(L)=I
   30 L=L+1
   40 CONTINUE
   50 CONTINUE
      WRITE (NT) (MASS(I),I=1,NEQ)
C
      READ (5,1000) JFN,JAT,PFN
      DO 100 I=1,3
      IF (JAT(I).EQ.0  ) JAT(I)=1
```

```
      JJ=JFN(I)
      II=JAT(I)
      IF (JJ.GT.0) IDF(JJ,II)=1
  100 CONTINUE
      WRITE(6.2000) JFN,JAT,PFN
C
      NNN=NEQB*NFN
      DO 500 N=1,NBLOCK
      NN=(N-1)*NEQB
      READ (JT) XM
      IF (NARB.EQ.0) GO TO 200
      READ (MT) FF,IFF
      GO TO 300
  200 DO 250 I=1,NNN
      FF(I)=0.
      IFF(I)=0
  250 CONTINUE
C
  300 DO 400 I'=1,NEQB
      I=NN+I!
      J=MASS(I)
      IF (J.LE.0) GO TO 400
      JJ=JFN(J)
      IF (JJ.LE.0) GO TO 400
      FF(II,JJ)=-XM(II)*PFN(J)
      IFF(II,JJ)=JAT(J)
  400 CONTINUE
      WRITE (K') FF,IFF
  500 CONTINUE
      RETURN
C
 1000 FORMAT (6I5,3F10.0)
 2000 FORMAT (3CH1GROUND MOTION INPUT KEY...... ///
     .  30X 9HDIRECTION /24X 1HX 7X 1HY 7X 1HZ  //
     .  20H  FUNCTION NUMBER... I5,2I8/20H  ARRIVAL TIME NO... I5,2I8 /
     .  20H  SCALE FACTOR...... 3F8.3 )
C
      END


      SUBROUTINE INPGM (IOF,IO,R,NFNG,PFNG,NATG,NFN,NAT,NUMNP,NEQB,
     .                    LG,NEQ,NBLOCK)
C
      DIMENSION IOF(NFN,1),IO(NUMNP,1),R(NEQB,LG),
     .          NFNG(LG),PFNG(LG),NATG(LG)
C
C     GENERATE PSEUDO-STATIC COEFFICIENTS FOR MULTI-GROUND-MOTION INPUT
C
      MT=2
      IT=13
      REWIND IT
      L=0
      DO 100 I=1,LG
      NFNG(I)=0
      PFNG(I)=0.
      NATG(I)=0
  100 CONTINUE
C
      WRITE(6,2000)
  150 L=L+1
      READ (5,1000) NP,IC,IFN,IAT,P
      IF (IAT.EQ.0) IAT=1
      IF (P.EQ.0.0) P=1.0
      IF ( NP.LE.0) GO TO 300
      WRITE(6,2001) NP,IC,IFN,IAT,P
      IF (IFN.GT.0) IOF(IFN,IAT)=1
C
      JJ=IO(NP,IC)
      IF (JJ.LE.NEQ) CALL ERROR (8,NP,7HINMGM   )
      JJ=JJ-NEQ
      NFNG(JJ)=IFN
      NATG(JJ)=IAT
      PFNG(JJ)=P
      GO TO 150
  300 IF (L.LE.LG) GO TO 350
      CALL ERROR (8,NP,7HINMGM   )
  350 WRITE (IT) NFNG,PFNG,NATG
C
      DO 500 N=1,NBLOCK
      BACKSPACE MT
      READ (MT) R
      BACKSPACE MT
      WRITE (IT) R
  500 CONTINUE
C
      RETURN
 1000 FORMAT (4I5,F10.0)
 2000 FORMAT (35H1MULTIPLE GROUND MOTION INPUT DATA.....  //
     .  54H  NODE   DISPLACEMENT  FUNCTION   ARRIVAL TIME  SCALE  /
     .  54H  NO.    COMPONENT     NUMBER        NUMBER     FACTOR /)
 2001 FORMAT (I6,2I11,I14,F12.3)
      END


      SUBROUTINE INCLT (IOF,LAT.KAT,AT,NFN,NAT,NFNP)
C
      DIMENSION IOF(NFN,NAT),LAT(NAT),KAT(NAT),AT(NAT)
      COMMON/JUNK/ NARB,DT,ALFA,BETA,INTG,JFN(3),PFN(3),JAT(3),IAT,NA,NF
C
C     INPUT DELAY TIMES
C
      READ (5,1000) (AT(I),I=1,NAT)
      WRITE(6,2000) (I,AT(I),I=1,NAT)
C
      DO 100 I=1,NAT
      AT(I)=AT(I)/DT
      IAT=AT(I)
      IF ((AT(I)-IAT).GE.0.5) IAT=IAT+1
      IAT=IAT+1
      LAT(I)=IAT
  100 CONTINUE
C
      DO 150 I=1,NAT
  150 KAT(I)=0
      IF (NAT.LT.2) GO TO 300
      NA=NAT-1
      DO 250 I=1,NA
      IF (KAT(I).GT.0) GO TO 250
      IAT=LAT(I)
      JJ=I+1
```

```
      DO 22C J=JJ,NAT
      IF (LAT(J).EG.IAT) KAT(J)=I
  220 CONT(NUE
  250 CCNT(NUE
C
C     COUNT TIME FUNCTION NUMBER INCLUDING OELAY TIME EFFECT
C
  300 NF=0
      DO 45C I=1,NFN
      DO 40C J=1,NAT
      IF ((CF((,J).LE.0) GO TO 400
      JJ=KAT(J)
      IF (JJ.LE.0) GO TO 350
      IF (ICF((,JJ).LE.0) GO TO 350
      ICF(I,J)=IDF(I,JJ)
      GC TC 400
  350 NF=NF+1
      IDF(I,J)=NF
  4CO CONTINUE
  450 CONTINUE
      NFNR=NF
      RETURN
C
 1000 FORMAT (8F10.2)
 2000 FORMAT (/////15H OELAY TIMES...// 15X 7H DELAY  /8H  NUMBER 8X
     .  4HTIME /(I6,F14.2)I
C
      END
```

```
      SUBRCLTINE LCAOM (IDF,LAT,FR,FF,IFF,NFNG,PFNG,NATG,R,XM,
     .                 NFN,NAT,NFNR,NEQB,LG,NGM,NBLOCK)
C
      DIMENSION IDF(NFN,NAT),LAT(NAT),FR(NEQB,NFNRI,FF(NEQB,NFN),
     .          IFF(NEQB,NFN),NFNG(LG),PFNG(LG),NATG(LG),R(NEQB,LG),
     .          XM(NEQB)
      CCMMON/JUNK/ NARB,DT,ALFA,BETA,INTG,JFN(3),PFN(3),JAT(3),NTREAD,
     .             IFN,IAT,NF
C
C     FORM LOAD MATRIX INCLUDING DELAY TIME EFFECT
C
      LT=9
      NT=10
      IT=11
      JT=12
      KT=13
      REWINC IT
      REWINC JT
      REWINC KT
      REWINC LT
      REWIND NT
      NTREAC=1
      IF ((NARB.EQ.C).AND.(NGM.NE.1)) NTREAD=0
      IF (NGM.NE.2I GO TO 20
      READ (KT) NFNG,PFNG,NATG
      DO 10 L=1,LG
      IFN=NFNG(L)
      IAT=NATG(L)
      IF (IFN.LE.0) GO TO 10
      IF (IAT.LE.0) GO TO 1D
      NF=IDF(IFN,IAT)
      NFNG(L)=NF
   10 CONTINUE
      WRITE (ITI NFNG,PFNG,NATG
C
   20 IF (NGM.NE.1) GO TO 50
      DO 40 I=1,3
      IFN=JFN(II
      IAT=JAT(I)
      IF (IFN.LE.0) GO TO 40
      IF (IAT.LE.0) GO TO 40
      NF=IDF(IFN,IAT)
      JFN(I)=NF
   40 CONTINUE
C
   50 NNR=NEQB*NFNR
      DO 800 N=1,NBLOCK
      DO 100 I=I,NNR
  100 FR(I)=0.
      IF (NTREAC.EC.0) GO TO 400
      READ (NT) FF,IFF
      DO 30C I=I,NECB
      DO 20C J=I,NFN
      IAT=IFF(I,JI
      IF ((AT.LE.0) GO TO 200
      NF=IDF(J,IAT)
      IF (NF.LT.1) GO TD 200
      FR(I,NFI=FR(I,NF)+FF(I,J)
  200 CONTINUE
  300 CONTINUE
  400 IF (NGM.NE.2I GO TO 700
      READ (LT) XM.
      READ (KT) R
      WRITE (IT) R
      DO 60C L=1,LG
      NF=NFNG(L)
      IF (NF.LT.1) GO TO 600
      DO  5CO I=1,NEQB
  500 FR(I,NF)=FR(I,NF)-XM(I)*R(I,L)*PFNG(LI
  600 CONT(NUE
  700 WRITE (JT) FR
  800 CONT(NUE
      RETURN
      END
```

```
      SUBROLTINE INPIS (PP,P,T,NFN,NT,MAXI
C
      DIMENSION PP(NFN,NT),P(MAXI,T(MAX)
      COMMON/JUNK/ NARB,DT,ALFA,BETA,INTG,JFN(3I,PFN(3I,NLP,SFTR,TINCR,
     .             MEU(12),TL,TIME,OUT,DUP,SLOPE
C
C     INPUT LOAD TIME FUNCTIONS
C
      DO 5CC I=I,NFN
      READ (5,1COO) MEU,NLP,SFTR,TINCR
      IF (SFTR.EQ.0.I SFTR=1.0
      WRITE(6,2COC) I,MEU,NLP,SFTR
      IF (NLP.LE.MAX) GO TO 300
      L=2*(NLP-MAX)
```

```
      CALL ERROR (1,L,7HINHIS  )
  300 IF (TINCR.LE.0.) GO TO 350
      DO 310 L=1,NLP
      TL=L-1
  310 T(L)=TINCR*TL
      READ (5,1003) (P(L),L=1,NLP)
      GO TO 380
  350 READ (5,1001) (T(L),P(L),L=1,NLP)
  380 WRITE(6,2001) (T(L),P(L),L=1,NLP)
C
C     GENERATE EQUAL INTERVAL INCREMENTAL LOAD TIME FUNCTIONS
C
      TIME=DT
      TEMP=P(1)*SFTR
      L=1
      K=1
  410 L=L+1
      LL=L-1
  412 DDT=T(L)-T(LL)
      DDP=P(L)-P(LL)
      IF (DDT) 415,410,420
  415 WRITE(6,2003) T(L),P(L)
      L=L+1
      GO TO 412
  420 SLOPE=DDP/DDT
  423 IF (T(L)-TIME) 410,425,425
  425 PPP=P(LL)+(TIME-T(LL))*SLOPE
      PPP=PPP*SFTR
      PP(I,K)=PPP-TEMP
      TEMP=PPP
  430 TIME=TIME+DT
      K=K+1
      IF (NT-K) 500,423,423
  500 CONTINUE
      RETURN
C
 1000 FORMAT (12A6/I5,2F10.0)
 1001 FORMAT (12F6.0)
 1003 FORMAT (10F7.4,10X)
 2000 FORMAT (22H1TIME FUNCTION NO......,I4,3X,12A6//
     .  6X 25HNUMBER OF LOAD POINTS    =  I6/
     .  6X 25HSCALE FACTOR             =  F10.3//)
 2001 FORMAT (5(19H   TIME    INPUT     )/(5(F7.3,F8.3,4X)))
 2003 FORMAT (19H   BAD LOAD DATA..  ,F7.3,F8.3,4X 12HDATA IGNORED )
      END


      SUBROUTINE LOADH (IOF,LAT,PP,PR,KR,NFN,NAT,NFNR,NT,NTB,NB,NGM)
C
      DIMENSION IOF(NFN,NAT),LAT(NAT),PP(NFN,NT),PR(NFNR,NTB),KR(NFNR)
C
C     FORM LOAD HISTORIES INCLUDING DELAY TIME EFFECT
C
      IT=2
      KT=10
      REWIND IT
      REWIND KT
      NS=1
      NE=NTB
      NK=NTB
      NNN=NFNR*NTB
C
      DO 800 N=1,NB
      DO 100 I=1,NNN
  100 PR(I)=0.
      DO 200 I=1,NFNR
  200 KR(I)=0
      DO 500 I=1,NFN
      DO 400 J=1,NAT
      NF=IOF(I,J)
      IF (NF.LE.0) GO TO 400
      IF (KR(NF).GT.0) GO TO 400
      JAT=LAT(J)
      NL=NS-JAT
      K=1
      IF (NL.LT.0) K=1-NL
      IF (K.GT.NK) GO TO 350
      DO 300 II=K,NK
      JJ=NL+II
  300 PR(NF,II)=PP(I,JJ)
  350 KR(NF)=1
  400 CONTINUE
  500 CONTINUE
      IF (NGM.EQ.0) GO TO 700
      DO 600 II=1,NK
  600 WRITE (KT) (PR(I,II),I=1,NFNR)
C
  700 WRITE (IT) PR
      NS=NS+NTB
      NE=NE+NTB
      IF (NE.GT.NT) NK=NT-NS+1
  800 CONTINUE
C
      RETURN
      END


      SUBROUTINE LOADV (FPR,FR,PR,DP,NEQB,NFNR,NTB,NBLOCK,NB,NT,NEQ,NG)
C
      DIMENSION FPR(NEQB,NTB),FR(NEQB,NFNR),PR(NFNR,NTB),DP(NQ,NTB)
C
C     FORM TIME HISTORY OF NODAL INCREMENTAL LOAD VECTOR
C
      IT=2
      JT=7
      KT=12
      REWIND IT
      REWIND JT
      REWIND KT
C
      DO 500 NN=1,NB
      READ (IT) PR
      DO 400 N=1,NBLOCK
      READ (KT) FR
      DO 300 J=1,NTB
      DO 300 I=1,NEQB
      DO 300 K=1,NFNR
  300 FPR(I,J)=FR(I,K)*PR(K,J)
      WRITE (JT) FPR
  400 CONTINUE
```

```
      REWIND KT
  500 CONTINUE
C
      REWIND JT
      NL=NTB
      NK=NTB
      DO 800 NN=1,NB
      NS=1
      NE=NEQB
      DO 700 N=1,NBLOCK
      READ (JT) ((CP(I,J),I=NS,NE),J=1,NTB)
      NS=NS+NEQB
  700 NE=NE+NEQB
      IF (NL.GT.NT) NK=NT-(NN-1)*NTB
      DO 750 K=1,NK
  750 WRITE (KT) (DP(I,K),I=1,NEQ)
      NL=NL+NTB
  800 CONTINUE
C
      RETURN
      END
```

```
      SUBROUTINE STEPC
C
      COMMON/ELPAR/ CONT(14),NUMNP,NELTYP,NUMEL,NUMNEL,NEQ,NEQG,MBAND,
     .                NEQB,NBLOCK,MTOT,N1,N2,N3,N4,N5,N6,N7
      COMMON/EM/ QCCQ(2043)
      COMMON/MISC/ LG,LL,NFNR,NFN,NGM,MAT,NT,NOT,MSTEP
      COMMON/JUNK/ NSTIF,DT,ALFA,BETA,INTG,JFN(3),PFN(3),JUK(194)
      COMMON A(1)
C
C     STEP-BY-STEP NONLINEAR DYNAMIC ANALYSIS IN CORE
C
      N1=1
      N2=N1+NEQ
      N3=N2+NEQ
      N4=N3+NEQ
      N5=N4+NEQ
      N6=N5+NEQ
      N7=N6+NEQ
      N8=N7+NEQ
      N9=N8+NEQ*MBAND
      N10=N9+NEQ*LG
      N11=N10+NUMNEL
      N12=N11+NUMNEL
      N13=N12+NUMNEL*12
      N14=N13+NUMNEL*12
      N15=N14+NEQ
      N16=N15+NEQ*MBAND
      N17=N16+NEQ
      IF (NUMEL.GT.NEQ) N17=N16+NUMEL
      N18=N17+NEQ*LG
      N19=N18+NEQ*LG
      N20=N19+LG
      N21=N20+LG
      N22=N21+NFNR
      IF (N22.GT.MTOT) CALL ERROR (1,N22-MTOT,7HSTEPC   )
C
C     INITIALIZATION
C
      MB=MBAND+LG
      NC=NEQB*NBLOCK
      CALL START (A(N1),A(N1),A(N1),A(N2),A(N3),A(N4),A(N5),A(N6),A(N7),
     .                A(N8),A(N9),A(N10),A(N11),A(N12),A(N13),A(N15),A(N16),
     .                A(N18),A(N19),A(N20),MB,NC,MBAND,LG,NUMEL,NUMNEL,NEQ)
      INDMAX=30
      IF (INDMAX.GT.NUMNEL) INDMAX=NUMNEL
      IF (NUMNEL.GT.0) WRITE(6,2000) (I,I=1,INDMAX)
C
C     SOLUTION BY STEP-BY-STEP INTEGRATION
C
      CALL INTGR (A(N1),A(N2),A(N3),A(N4),A(N5),A(N6),A(N7),A(N8),A(N9),
     .             A(N10),A(N11),A(N12),A(N13),A(N14),A(N15),A(N16),A(N17)
     .             A(N18),A(N19),A(N20),A(N21),A(N14),A(N14),NEQ,MBAND,
     .             NUMNEL,LG,NFNR)
      RETURN
C
 2000 FORMAT (40H1STEP-BY-STEP ANALYSIS DATA MONITOR.....  //
     .  12H STEP TAPE   20X 28HNONLINEAR ELEMENT INDICATORS/
     .  12H  NO  STIF        30I3  /)
C
      END
```

```
      SUBROLTINE START (B,D,X,X1,X2,XG,MASS,XM,OPU,AA,CC,NINU,NSS,STR,
     .                 UPL,A,INO,R,NFNG,PFNG,MB,NQ,MBAND,LG,NUMEL,NUMNEL,NEQ)
C
      DIMENSION B(NC,MB),D(NQ,LG),X(NEQ),X1(NEQ),X2(NEQ),XG(NEQ),
     .               MASS(NEQ),XM(NEQ),OPU(NEQ),NINO(NUMNEL),NSS(NUMNEL),
     .               STR(NUMNEL,12),UPL(NUMNEL,12),A(NEQ,MBAND),INO(NUMEL),
     .               R(NEQ,LG),NFNG(LG),PFNG(LG),AA(NEQ,MBAND),CC(NEQ,LG)
      COMMON/ELFAP/ CONT(14),NUMNP,NELTYP,MUMEL,MUMNEL,MEQ,NEQG,NBAND,
     .                     NEQB,NBLOCK,MTOT,N1,N2,N3,N4,N5,N6,N7
      COMMON/JUNK/ NSTIF,DT,ALFA,BETA,INTG,JFN(3),PFN(3),NS,SIG(12),NB
      COMMON/COLLCMB/ CF(20,6)
      DATA CF /120*0.0/
C
C     GENERATE COMPUTATIONAL CONSTANTS
C
      DO 100 I=1,14
  100 CONT(I)=0.
C
      IF (INTG) 150,150,160
C
C     CONSTANT ACCELERATION SCHEME
C
  150 CONT(1)=4.0/DT/DT
      CONT(2)=2.0/DT
      CONT(3)=4.0/DT
      CONT(4)=2.0
      GO TO 200
C
C     LINEAR ACCELERATION SCHEME
C
  160 CONT(1)=6.0/DT/DT
      CONT(2)=3.0/DT
      CONT(3)=6.0/DT
      CONT(4)=3.0
      CONT(5)=DT/2.0
C
C     NORMALIZED CONSTANTS
C
  200 CONT(6)=1.0+BETA*CONT(2)
      CONT(7)=(CONT(1)+ALFA*CONT(2))/CONT(6)
      CONT(8)=1.0/CONT(6)
      CONT(9)=(CONT(3)+ALFA*CONT(4))/CONT(6)
      CONT(10)=BETA*CONT(4)/CONT(6)
      CONT(11)=(CONT(4)+ALFA*CONT(5))/CONT(6)
      CONT(12)=BETA*CONT(5)/CONT(6)
      CONT(13)=CONT(9)-CONT(7)*CONT(10)
      CONT(14)=CONT(11)-CONT(7)*CONT(12)
C
C     SET UP STIFFNESS MATRIX IN CORE
C
      REWIND 4
      NB=1
      NE=NEQB
      DO 300 N=1,NBLOCK
      READ (4) ((B(I,J),I=NB,NE),J=1,MB)
      NB=NB+NEQB
  300 NE=NE+NEQB
      REWIND 4
      WRITE (4) ((B(I,J),I=1,NEQ),J=1,MB)
C
      IF (LG.LE.0) GO TO 500
      REWIND 11
      READ (11)
      NB=1
      NE=NEQB
      DO 400 N=1,NBLOCK
      READ (11) ((D(I,L),I=NB,NE),L=1,LG)
      NB=NB+NEQB
  400 NE=NE+NEQB
      REWIND 11
      READ (11)
      WRITE (11) ((D(I,L),I=1,NEQ),L=1,LG)
C
C     INITIALIZE DISPLACEMENTS, VELOCITIES, ACCELERATIONS
C
  500 DO 510 I=1,NEQ
      X(I)=0.
      X1(I)=0.
      X2(I)=0.
      XG(I)=0.
  510 OPU(I)=0.
C
C     SET UP MASS MATRIX IN CORE
C
      NB=1
      NE=NEQB
      REWIND 9
      REWIND 15
      READ (15) (MASS(I),I=1,NEQ)
      DO 550 N=1,NBLOCK
      READ (9) (XM(I),I=NB,NE)
      NB=NB+NEQB
      NE=NE+NEQB
      IF (NE.GT.NEQ) NE=NEQ
  550 CONTINUE
C
C     INITIALIZE NONLINEAR ELEMENT STRESSES TO BE STATIC STRESSES
C
      READ (8) INC
      K=0
      DO 600 N=1,NUMEL
      READ (8) NS,SIG
      IF (INO(N).LE.0) GO TO 600
      K=K+1
      NINO(K)=N
      NSS(K)=NS
      DO 580 I=1,12
      STR(K,I)=SIG(I)
  580 UPL(K,I)=0.
  600 CONTINUE
C
      REWIND 4
      IF (LG) 700,700,750
  700 READ (4) ((AA(I,J),I=1,NEQ),J=1,MBAND)
      GO TO 800
  750 READ (4) ((AA(I,J),I=1,NEQ),J=1,MBAND),((CC(I,L),I=1,NEQ),L=1,LG)
  800 DO 900 I=1,NEQ
      DO 900 J=1,MBAND
  900 A(I,J)=AA(I,J)
C
      IF (LG.LE.0) RETURN
```

```
      REWIND 11
      READ (11) (NFNG(L),L=1,LG),(PFNG(L),L=1,LG)
      READ (11) ((R(I,L),I=1,NEQ),L=1,LG)
C
      RETURN
      END



      SUBROUTINE INTGR (X,X1,X2,XG,MASS,XM,DPU,B,D,NINO,NSS,STR,UPL,DX,A
     .,           NBMAX,RT,R,NFNG,PFNG,PR,DP,XT,NEQ,MBAND,NUMNEL,LG,NFNR)
C
      DIMENSION X(NEQ),X1(NEQ),X2(NEQ),XG(NEQ),MASS(NEQ),XM(NEQ),
     .          DPU(NEQ),NINO(NUMNEL),NSS(NUMNEL),STR(NUMNEL,12),
     .          UPL(NUMNEL,12),DX(NEQ),A(NEQ,MBAND),NBMAX(NEQ),
     .          RT(NEQ,LG),R(NEQ,LG),NFNG(LG),PFNG(LG),PR(NFNR),
     .          DP(NEQ),XT(NEQ),B(NEQ,MBAND),D(NEQ,LG)
      COMMON/ELPAR/CONT(14),NNNN(17)
      COMMON/EM/ QQCQ(2043)
      COMMON/MISC/ LH,LL,MFNR,NFA,NGM,NAT,NT,NOT,NSTEP
      COMMON/JUNK/ NSTIF,DT,ALFA,BETA,INTG,JFN(3),PFN(3),JUK(194)
C
C     STEP-BY-STEP INTEGRATION ALGORITHM
C
      REWIND 9
      REWIND 10
      REWIND 12
C
C     THE FOLLOWING TOLERANCE SHOULD BE RESET FOR LENGTH DIMENSION
C     OTHER THAN FT ..........
      TOLOX=1.0E 1
C
      NSTIF=4
      NOT=1
      NCUT=NOT
C
C     FORM EFFECTIVE DYNAMIC STIFFNESS MATRIX AND LOAD VECTOR
C
  100 READ (12) (DP(I),I=1,NEQ)
      DO 120 I=1,NEC
      DPI=DP(I)+DPU(I)
      DP(I)=CONT(8)*DPI+CONT(13)*XM(I)*X1(I)+CONT(14)*XM(I)*X2(I)
      A(I,1)=A(I,1)+CONT(7)*XM(I)
  120 CONTINUE
C
C     SOLVE FOR EFFECTIVE DYNAMIC DISPLACEMENT INCREMENTS
C
      CALL TRIA (A,NBMAX,NEQ,MBAND)
      CALL BACK (A,NBMAX,DP ,NEQ  )
C
C     COMPUTE NEW DISPLACEMENTS, VELOCITIES AND ACCELERATIONS
C
      DO 200 I=1,NEC
      DD =DP(I)
      XX1=X1(I)
      XX2=X2(I)
      DXX=DD+CONT(10)*XX1+CONT(12)*XX2
      DX2=CONT(1)*DXX-CONT(3)*XX1-CONT(4)*XX2
      DX1=CONT(2)*DXX-CONT(4)*XX1-CONT(5)*XX2
      X (I)=X (I)+DXX
      X1(I)=X1(I)+DX1
      X2(I)=X2(I)+DX2
      DX(I)=DXX
  200 CONTINUE
C
      DO 300 I=1,NEC
      IF (ABS(DX(I)).LT.TOLOX) GO TO 300
      NT=NOT-1
      WRITE(6,2000) NOT,(DX(I),I=1,NEQ)
      GO TO 1000
  300 CONTINUE
C
C     CHECK NONLINEARITY AND FORM NEW STIFFNESSES IF NEEDED
C
      CALL NELSTF (X,X1,DPU,B,D,NINO,NSS,STR,UPL,DX,A,RT,
     .                NEQ,MBAND,LG,NUMNEL,NOT,NOUT,NSTIF)
C
      IF (NGM-1) 900,800,400
C
  400 IF (NSTIF.EQ.2) GO TO 500
      GO TO 700
C
C     COMPUTE NEW PSEUDO-STATIC COEFF. FOR MULTI-GROUND MOTION INPUT
C
  500 CALL TRIA (A,NBMAX,NEQ,MBAND)
      DO 530 L=1,LG
      DO 510 I=1,NEC
  510 DX(I)=RT(I,L)
      CALL BACK (A,NBMAX,DX,NEQ)
      DO 520 I=1,NEC
  520 RT(I,L)=DX(I)
  530 CONTINUE
      REWIND NSTIF
      READ (NSTIF) ((A(I,J),I=1,NEQ),J=1,MBAND)
C
C     COMPUTE ABSOLUTE ACCELERATION
C
      READ (10) (PR(I),I=1,NFNR)
      DO 600 I=1,NEC
      XX=0.
      YY=0.
      DO 580 L=1,LG
      NF=NFNG(L)
      IF (NF.LE.0) GO TO 580
      XX=XX+RT(I,L)*PFNG(L)*PR(NF)
      YY=YY-XM(I)*(RT(I,L)-R(I,L))*PFNG(L)*PR(NF)
  580 CONTINUE
      XG (I)=XG (I)+XX
      DPU(I)=DPU(I)+YY
  600 CONTINUE
      GO TO 850
C
  700 READ (10) (PR(I),I=1,NFNR)
      DO 720 I=1,NEC
      XX=0.
      DO 710 L=1,LG
      NF=NFNG(L)
      IF (NF.LE.0) GO TO 710
      XX=XX+R(I,L)*PFNG(L)*PR(NF)
  710 CONTINUE
  720 XG(I)=XG(I)+XX
      GO TO 850
C
```

```
STPC 302     800 READ (10) (PR(I),I=1,NFNR)
STPC 303         DO 82C I=1,NEC
STPC 304         J=MASS(I)
STPC 305         IF (J.LE.C) GC TO 820
STPC 306         JJ=JFN(J)
STPC 307         IF (JJ.LE.O) GO TO 820
STPC 308         XG(I)=XG(()+PFN(J)*PR(JJI
STPC 309     820 CCNTINUE
STPC 310   C
STPC 311   C     STORE DISPLACEMENTS AND ABSOLUTE ACCELERATIONS ON TAPE
STPC 312   C
STPC 313     850 DO 860 I=1,NEC
STPC 314     860 XT(I)=X2(I)+XC(II
STPC 315         IF (NCT.EC.NCLT) WRITE (8) (X(I),I=1,NEQI,(XT((),I=1,NEQ)
STPC 316         GO TC 950
STPC 317     900 IF (NCT.EC.NCLT) WRITE (8) (X((),I=1,NEQI,(X2(I),(=1,NEQ)
STPC 318   C
STPC 319   C     CHECK FOR LAST TIME STEP
STPC 320   C
STPC 321     950 IF (NCT.EC.NCUT) NOUT=NOUT+NOT
STPC 322         NOT=NCT+1
STPC 323         (F (NCI-NT) 1C0,100,1000
STPC 324    1000 RETURN
STPC 325   C
STPC 326    2000 FCRMAT (5CH1*****DISPLACEMENT INCREMENTS BLEW UP AT STEP NO  I6//
STPC 327       .          29H       SKIP TO OUTPUT ROUTINE.  ///
STPC 328       .          41H DISPLACEMENT INCREMENTS OF LAST STEP.... //(1P6E12.3))
STPC 329   C
STPC 330         END


STPC 331         SUBRCUTINE TRIA (A,NBMAX,NEQ,MBAND)
STPC 332   C
STPC 333         DIMENSION A(1),NBMAX(1)
STPC 334   C
STPC 335   C     TRIANGULARIZE BANDED MATRIX BY GAUSS ELIMINATION
STPC 336   C
STPC 337         IF (NEC.EC.1) RETURN
STPC 338         MM=NEC*MBAND
STPC 339         NE=NEC-1
STPC 340         DO 3CC N=1,NE
STPC 341   C
STPC 342   C     DETERMINE VARIABLE BAND WIDTH
STPC 343   C
STPC 344         NBMAX(N)=C
STPC 345         DO 1CC I=N,MM,NEQ
STPC 346         IF (A(I).NE.C.0) NBMAX(NI=I
STPC 347     100 CCNTINUE
STPC 348   C
STPC 349   C     REDUCTION OF EQUATIONS WITHIN BAND
STPC 350   C
STPC 351         IF (A(N).EC.C.0I GO TO 300
STPC 352         IL=N+NEQ
STPC 353         IH=NBMAX(N)
STPC 354         L=N
STPC 355         DC 2CC I=IL,IH,NEQ
STPC 356         L=L+1
STPC 357         IF (A(II.EQ.C.0) GO TO 200
STPC 358         C=A(I)/A(N)
STPC 359         J=L-I
```

```
STPC 360         DO 15C K=I,IH,NEQ
STPC 361     150 A(K+JI=A(K+J)-C*A(KI
STPC 362         A(I)=C
STPC 363     200 CONTINUE
STPC 364     300 CONTINUE
STPC 365         RETURN
STPC 366         END
```

```
      SUBROLTINE BACK (A,NBMAX,B,NEQ)
C
      DIMENSION A(1),NBMAX(1),B(1)
C
C     REDUCTION OF LOAD VECTOR AND BACKSUBSTITUTION
C
      IL=NEC
      DO 40C N=1,NEC
      C=B(N)
      IF (A(N).NE.C.D) B(N)=B(N)/A(N)
      IF (N.EQ.NEQ) GO TO 450
      IL=IL+1
      IH=NBMAX(N)
      K=N
      DO 35C I=IL,IH,NEQ
      K=K+1
  350 B(K)=B(K)-A(I)*C
  400 CONTINUE
C
  450 IL=2*NEQ
  500 IL=IL-1
      N=N-1
      IF (N.EQ.C) RETURN
      IH=NBMAX(N)
      K=N
      DO 60C I=IL,IH,NEQ
      K=K+1
  600 B(N)=B(N)-A(I)*B(K)
      GO TO 500
      END


      SUBROLTINE NELSTF (X,X1,DPU,B,D,NIND,NSS,STR,UPL,DX,A,C,
     .                   NEQ,MBAND,LG,NUMNEL,NOT,NOUT,NSTIF)
C
      DIMENSION X(1),X1(1),DX(1),NIND(1),NSS(NUMNEL),STR(NUMNEL,12),
     .          UPL(NUMNEL,12),DPU(1),C(NEQ,LG),A(NEQ,MBAND),STIF(1131),
     .          B(NEQ,MBAND),D(NEQ,LG)
      COMMCN/EM/ MTYPE,LM(24),ND,NS,ASA(24,24),SA(12,24),S(12,12),
     .           ENPAR(24),TTT(72),ASSA(24,24),SSA(12,24),DF(24),
     .           F(12),UP(12)
      COMMON/NEM/ EX(24),EX1(24),DEX(24),DFF(12),DP(12),FF(12),P(12),
     .            SS(12,12),U(12)
      EQUIVALENCE (STIF,MTYPE)
C
C     COMPUTE NEW NONLINEAR ELEMENT STRESSES AND STIFFNESSES
C     ------ CHANGE GLOBAL EQUILIBRIUM EQUATIONS ACCORDINGLY
C
      DO 10 I=1,NEC
   10 DPU(I)=0.
      DC 20 I=1,NEC
      DO 20 J=1,MBAND
   20 A(I,J)=B(I,J)
      IF (LG.LE.D) GO TO 40
      DO 30 I=1,NEC
      DC 30 L=1,LG
   30 C(I,L)=D(I,L)
   40 CONTINUE
      IF (NLMNEL.LE.0) GO TO 1000
C
      M=0
      REWIND 3
      DO 800 N=1,NUMNEL
      DO 50 I=1,72
   50 EX(I)=D.
      READ (3) STIF
      DO 100 I=1,NC
      II=LM(I)
      IF (II.LE.0.CR.II.GT.NEQ) GO TO 100
      EX (I)=X (II)
      EX1(I)=X1(II)
      DEX(I)=DX(II)
  100 CONTINUE
      DO 200 I=1,12
      F (I)=STR(N,I)
  200 UP(I)=UPL(N,I)
C
      IND=NIND(N)
      GO TO (1,2,3,4,5) MTYPE
C
C     NONLINEAR TRUSS ELEMENTS--SKIP
C
    1 GO TC 600
C
C     ELASTC-PLASTIC BEAM ELEMENTS
C
    2 CALL PBEAM (N,IND)
      GO TC 600
C
C     NCNLINEAR CURVED BEAM ELEMENTS--SKIP
C
    3 GO TO 600
C
C     BILINEAR BOUNCARY SPRING ELEMENTS
C
    4 CALL PBOUND (N,IND)
      GO TC 600
C
C     NONLINEAR EXPANSION JOINT ELEMENTS
C
    5 M=M+1
      CALL NEXPJB (M,IND)
C
  600 NIND(N)=IND
C
C     ASSEMBLE NEW TOTAL STIFFNESSES AND UNBALANCED LOAD VECTOR
C
      IF (IND.EC.0) GO TO 700
      DO 65C I=1,NC
      II=LM(I)
      IF (II.LE. 0 ) GO TO 650
      IF (II.GT.NEC) GO TO 650
      DPU(II)=DPU(II)+DF(I)
      LMN=1-II
      DO 640 J=1,ND
      JJ=LM(J)
      IF (JJ.LE.NEC) GO TO 625
      L=JJ-NEQ
      C(II,L)=C(II,L)+ASSA(I,J)-ASA(I,J)
      GO TC 64C
  625 JJ=JJ+LMN
```

```
      IF (JJ) 640,640,635
  635 A(II,JJ)=A(II,JJ)+ASSA(I,J)-ASA(I,J)
  640 CONTINUE
  650 CONTINUE
C
C     UPDATE NONLINEAR ELEMENT STRESSES AND DEFORMATIONS
C
  700 NSS(N)=NS
      DO 750 I=1,12
      STR(N,I)=F(I)
  750 UPL(N,I)=UP(I)
  800 CONTINUE
C
      IF (NCT.NE.NCUT) GO TO 900
      WRITE (9) NSS,STR,UPL
      WRITE(6,2001) NOT,NSTIF,(NIND(I),I=1,NUMNEL)
  900 IF (LC.LE.0) GO TO 1000
      DO 910 N=1,NUMNEL
      IF (NIND(N).NE.0) GO TO 950
  910 CONTINUE
      NSTIF=4
      RETURN
C
  950 NSTIF=2
      REWIND 2
      WRITE (2) ((A(I,J),I=1,NEQ),J=1,MBAND)
 1000 RETURN
C
 2001 FORMAT (2I5,2X,(30I3))
C
      END
```

```
      SUBROUTINE STEPB
C
      COMMON/ELPAR/ CONT(14),NUMNP,NELTYP,NUMEL,NUMNEL,NEQ,NEQG,MBAND,
     *                 NEQB,NBLOCK,MTOT,N1,N2,N3,N4,N5,N6,N7
      COMMON/EM/ QCCQ(2043)
      COMMON/MISC/ LG,LL,NFNR,NFN,NGM,NAT,NT,NOT,MSTEP
      COMMON/JUNK/ NSTIF,DT,ALFA,BETA,INTG,JFN(3),PFN(3),JUK(194)
      COMMON A(1)
C
C     STEP-BY-STEP NONLINEAR DYNAMIC ANALYSIS IN BLOCKS
C
      LT=30
      NIT=1
      NDT=1
      NCUT=NOT
      NSTIF=4
C
      N1=1
      N2=N1+NEQ
      N3=N2+NEQ
      N4=N3+NEQ
      N5=N4+NEQ
      N6=N5+NEQ
      N7=N6+NEQ
      N8=N7+NEQ
      N9=N8+NEQ
C
      N10=NS +NUMNEL
      N11=N10+NUMNEL
      N12=N11+NUMNEL*12
      N13=N12+NUMNEL*12
C
      NC=NEQB*NBLOCK
      N14=N13+NC
      N15=N14+NEQB*MBAND
C
      LL=1
      NSB=NEQB*(MBAND+1)
      NSBB=NEQB*(2+(MBAND-1)/NEQB)
      IF (NSBB.LT.NSB) NSBB=NSB
      N21=N13+NSB
      N22=N21+NSBB
      N23=N22+NEQB
C
      N31=N13+NEQ
C
      NE2B=NEQB*2
      N41=N13+NE2B*MBAND
      N42=N41+NE2B*LG
C
      NGB=NEQB*(MBAND+LG)
      NGBB=NEQB*LG*(2+(MBAND-1)/NEQB)
      IF (NGBB.LT.NGB) NGBB=NGB
      N51=N13+NGB
      N52=N51+NGBB
      N53=N52+NEQB
C
      N61=N13+NEQB*LG
      N62=N61+NEQB*LG
      N63=N62+LG
      N64=N63+LG
```

```
      N65=N64+NFNR
C
      IF (N15.GT.MTCT) CALL ERROR (1,N15-MTOT,7HSTEPB   )
      IF (N23.GT.MTCT) CALL ERROR (1,N23-MTOT,7HSTEPB   )
      IF (N31.GT.MTCT) CALL ERROR (1,N31-MTOT,7HNELSTB  )
      IF (N42.GT.MTCT) CALL ERROR (1,N42-MTOT,7HSTEPB   )
      IF (N53.GT.MTCT) CALL ERROR (1,N53-MTOT,7HSTEPB   )
      IF (N65.GT.MTCT) CALL ERROR (1,N65-MTOT,7HSTEPB   )
C
C     INITIALIZATION
C
      CALL STARTB (A(N1),A(N2),A(N3),A(N4),A(N5),A(N6),A(N7),A(N8),
     .            A(N9),A(N10),A(N11),A(N12),A(N13),NEQ,NUMEL,NUMNEL)
C
      INDMAX=30
      IF (INDMAX.GT.NUMNEL) INDMAX=NUMNEL
      IF (NUMNEL.GT.0) WRITE(6,2000) (I,I=1,INDMAX)
C
C     STEP-BY-STEP SOLUTION IN BLOCKS
C
  100 CALL LOADP (A(N1),A(N2),A(N3),A(N7),A(N8),A(N13),
     .            A(N14),NEQ,NEQB,MBAND,NBLOCK,NQ,NSTIF)
C
      CALL USOL   (A(N13),A(N21),A(N22),NEQB,MBAND,LL,
     .            NBLOCK,NSB,7,13,14,15,15)
C
      CALL PACKB (A(N1),A(N2),A(N3),A(N4),A(N5),A(N6),A(N8),
     .            A(N13),NEQ,NEQB,NBLOCK,NDT,NOUT,NIT,NGM,NFNR)
C
      IF (NIT.EC.0) GO TO 1000
      IF (NUMNEL.LE.0) GO TO 500
C
      CALL NELSTB (A(N1),A(N2),A(N8),A(N9),A(N10),A(N11),A(N12),
     .             A(N13),NEQ,NUMNEL,NDT,NOUT,NSTIF,LT )
C
      CALL ADDB   (A(N9),A(N13),A(N41),NEQ,NE2B,MBAND,
     .            LC,NEQB,NBLOCK,NUMNEL,NSTIF,LT )
C
  500 IF (NCM-1) 900,900,600
C
  600 IF (NSTIF.EQ.4) GO TO 800
      CALL USOL   (A(N13),A(N51),A(N52),NEQB,MBAND,LG,
     .            NBLOCK,NGB,2,13,14,15,15)
C
  800 CALL MGMB   (A(N1),A(N3),A(N4),A(N5),A(N7),A(N8),A(N13),A(N61),
     .            A(N62),A(N63),A(N64),NEQ,NEQB,LG,NFNR,NBLOCK,NDT,
     .            NCUT,NSTIF )
C
  900 IF (NCT.EC.NCUT) NOUT=NOUT+NOT
      NOT=NOT+1
      IF (NCT-NT) 100,100,1000
 1000 RETURN
C
 2000 FORMAT (40H1STEP-BY-STEP ANALYSIS DATA MONITOR.....  //
     . 12H STEP TAPE   20X 28HNONLINEAR ELEMENT INDICATORS /
     . 12H  NO  NSTIF       30I3  /)
C
      END
```

```
      SUBROUTINE STARTB (X,X1,X2,XG,XT,MASS,XM,OPU,NINO,NS,STR,UPL,
     .                  IND,NEQ,NUMEL,NUMNEL )
C
      DIMENSION X(NEQ),X1(NEQ),X2(NEQ),XG(NEQ),XT(NEQ),MASS(NEQ),XM(NEQ)
     .,        OPU(NEQ),NINO(NUMNEL),IND(NUMEL),NS(NUMNEL),STR(NUMNEL,12),
     .         UPL(NUMNEL,12)
      COMMON/ELPAR/ CONT(14),NUMNP,NELTYP,MUMEL,MUMAEL,MEQ,NEQG,MBAND,
     .              NEQB,NBLOCK,MTOT,N1,N2,N3,N4,N5,N6,N7
      COMMON/JUNK/ NSTIF,DT,ALFA,BETA,INTG,JFN(3),PFN(3),NSS,SIG(12),
     .              NB,NE,JUK(179)
      COMMON/COLLCMP/ CF(20,6)
      DATA CF /120*0.0/
C
C     INITIALIZE DISPLACEMENTS, VELOCITIES, ACCELERATIONS AND UNBALANCED
C     FORCE VECTOR
C
      DO 10 I=1,NEQ
      X (I)=0.
      X1(I)=0.
      X2(I)=0.
      XG(I)=0.
      XT(I)=0.
      OPU(I)=0.
   10 CONTINUE
      REWIND 15
      READ (15) (MASS(I),I=1,NEQ)
C
C     INITIALIZE NONLINEAR ELEMENT STRESS TO BE STATIC STRESS
C
      READ (8) IND
      K=0
      DO 400 N=1,NUMEL
      READ (8) NSS,SIG
      IF (IND(N).LE.0) GO TO 400
      K=K+1
      NIND(K)=0
      NS(K)=NSS
      DO 300 I=1,12
      STR(N,I)=SIG(I)
      UPL(N,I)=0.
  300 CONTINUE
  400 CONTINUE
C
C     GENERATE COMPUTATIONAL CONSTANTS
C
      DO 500 I=1,14
  500 CONT(I)=0.
C
      IF (INTG) 600,600,700
C
C     CONSTANT ACCELERATION SCHEME
C
  600 CONT(1)=4.0/DT/DT
      CONT(2)=2.0/DT
      CONT(3)=4.0/DT
      CONT(4)=2.0
      GO TO 800
C
C     LINEAR ACCELERATION SCHEME
C
  700 CONT(1)=6.0/DT/DT
      CONT(2)=3.0/DT
```

```
      CONT(3)=6.0/CT
      CONT(4)=3.0
      CONT(5)=DT/2.C
C
C     NCRMALIZEC CCNSTANTS
C
  800 CONT(6)=1.0+PETA*CONT(2)
      CONT(7)=(CONT(1)+ALFA*CONT(2))/CONT(6)
      CONT(8)=1.0/CCNT(6)
      CONT(9)=(CONT(3)+ALFA*CUNT(4))/CONT(6)
      CONT(10)=PETA*CUNT(4)/CONT(6)
      CONT(11)=(CONT(4)+ALFA*CONT(5))/CONT(6)
      CONT(12)=PETA*CUNT(5)/GONT(6)
      CONT(13)=CONT(9)-CONT(7)*CONT(10)
      CCNT(14)=CCNT(11)-CONT(7)*CONT(12)
C
C     SET UP MASS MATRIX IN CORE
C
      REWINO 9
      NB=1
      NE=NECB
      DO 95C N=1,NBLOCK
      PEAD (9) (XM(I),I=NB,NE)
      NB=NB+NECB
      NE=NE+NECB
      IF (NE.GT.NEQ) NE=NEQ
  950 CCNTINUE
C
C     TAPE SETTING
C
      REWINC 9
      REWINC 10
      REWINC 12
C
      RETURA
      END
```

```
      SUBRCUTINE LCADB (X,X1,X2,XM,DPU,DP,A,NEQ,NEQB,MBAND,NBLOCK,
     .                  NQ,NSTIF )
C
      DIMENSION X(1),X1(1),X2(1),XM(1),DPU(1),DP(NQ),A(NEQB,MBAND)
      COMMCN/ELPAR/ CUNT(14),NNNN(17)
C
C     FORM EFFECTIVE STIFFNESS MATRIX AND LOAD VECTOR
C
      REWINC 7
      REWINO NSTIF
      DO 10 I=1,NG
   10 DP(I)=0.
      READ (12) (DP(I),I=1,NEQ)
      DO 20 I=1,NEC
   20 DP(I)=DP(I)+DPU(I)
C
      NS=1
      NE=NECB
      NK=NECB
      DO 80C N=I,NBLOCK
      READ (NSTIF) A
      DO 60C I=1,NK
      II=NS+I-1
      DP(II)=CONT(8)*DP(II)+CONT(13)*XM(II)*X1(II)
     .                     +CONT(14)*XM(II)*X2(II)
      A(I,I)=A(I,1)+CONT(7)*XM(II)
  600 CONTINUE
C
      WRITE (7) A,(DP(I),I=NS,NE)
      NS=NS+NEQB
      NE=NE+NEQB
      IF (NE.GT.NEQ)  NK=NEQ-NS+1
  800 CONTINUE
C
      RETURN
      END
```

```
      SUBRCUTINE PACKB (X,X1,X2,XG,XT,MASS,DX,PR,NEC,NECB,NBLOCK,
     .                  NOT,NOUT,NIT,NGM,NFNR )
C
      DIMENSION X(1),XI(1),X2(1),XG(1),XT(1),MASS(1),DX(1),PR(1)
      COMMCN/ELPAR/ CONT(14),NNNN(17)
      COMMCN/JUNK/ NSTIF,DT,ALFA,BETA,INTG,JFN(3),PFN(3),NS,NE,
     .                  DD,DX1,DX2,XX1,XX2,DXX
C
C     COMPUTE NEW DISPLACEMENTS, VELOCITIES, AND ACCELERATIONS
C
      NS=1
      NE=NECB
      DO 20C N=1,NBLOCK
      BACKSPACE 15
      READ (15) (DX(I),I=NS,NE)
      BACKSPACE 15
      NS=NS+NECB
      NE=NE+NECB
      IF (NE.GT.NEC) NE=NEQ
  200 CONTINUE
C
      DO 40C I=1,NEC
      DD =DX(I)
      XX1=X1(I)
      XX2=X2(I)
      DXX=DD+CONT(1C)*XX1+CONT(12)*XX2
      DX2=CCNT(1)*DXX-CONT(3)*XX1-CONT(4)*XX2
      DX1=CCNT(2)*CXX-CONT(4)*XX1-CONT(5)*XX2
C
      X (I)=X (I)+DXX
      X1(I)=X1(I)+DX1
      X2(I)=X2(I)+DX2
      OX(I)=DXX
  400 CONTINUE
C
      DO 50C I=1,NEC
      IF (ABS(DX(I)).LT.1.0E+ 1) GO TO 500
      NIT=0                  -
      NT=NOT-1
      WRITE(6,2000) NOT,(DX(I),I=1,NEQ)
      RETURN
  500 CONTINUE
C
      IF (NGM-1) 7CC,600,800
```

```
STPA 298  C
STPB 299    600 READ (10) (PR(I),I=1,NFNR)
STPB 300        DC 65C I=1,NEC
STPB 301        J =MASS(I)
STPB 302        IF (J.LE.0) GC TO 650
STPB 303        JJ=JFN(J)
STPB 304        IF (JJ.LE.0) GC TO 650
STPB 305        XG(I)=XG(I)+PFN(J)*PR(J)
STPB 306        XT(I)=X2(I)+XG(I)
STPA 307    650 CCNTINUE
STPB 308  C
STPB 309        IF (NCT.EC.NCLT) WRITE (8) (X(I),I=1,NEQ),(XT(I),I=1,NEQ)
STPA 310        GO TO 800
STPB 311  C
STPB 312    700 IF (NOT.EC.NCLT) WRITE (8) (X(I),I=1,NEQ),(X2(I),I=1,NEQ)
STPB 313    8CO RETURN
STPB 314  C
STPB 315   20CC FORMAT (5CH1*****DISPLACEMENT INCREMENTS BLEW UP AT STEP NO  I6//
STPA 316       .         29H *****SKIP TO OUTPUT ROUTINE.  ///
STPB 317       .         41H DISPLACEMENT INCREMENTS OF LAST STEP.... //(1P6E12.3))
STPB 318        END



STPB 319        SUBRCLTINE NELSTB (X,X1,DX,NIND,NSS,STR,UPL,DPU,NEQ,NUMNEL,
STPB 320       .                  NOT,NOUT,NSTIF,LT )
STPB 321  C
STPB 322        DIMENSION X(1),X1(1),DX(1),NIND(1),NSS(NUMNEL),STR(NUMNEL,12),
STPB 323       .          UPL(NUMNEL,12),DPU(1),STIF(1131),KM(25)
STPB 324        COMMON/EM/ MTYPE,LM(24),ND,NS,ASA(24,24),SA(12,24),S(12,12),
STPB 325       .           ENPAR(24),TTT(72),ASSA(24,24),SSA(12,24),DF(24),
STPB 326       .           F(12),UP(12)
STPA 327        COMMCN/NEM/ EX(24),EX1(24),OEX(24),DFF(12),DP(12),FF(12),P(12),
STPB 328       .            SS(12,12),U(12)
STPB 329        EQUIVALENCE (STIF,MTYPE)
STPB 330  C
STPB 331  C     COMPUTE ELEMENT STRESS INCREMENTS FOR NONLINEAR ELEMENTS
STPB 332  C
STPA 333        REWINC 3
STPB 334        REWINC LT
STPB 335        DO 10 I=1,NEQ
STPB 336     10 DPU(I)=0.
STPB 337  C
STPB 338        DO 80C N=1,NUMNEL
STPB 339        DO 20 I=1,72
STPB 340     20 EX(I)=0.
STPB 341        READ (3) STIF
STPB 342        M=0
STPB 343        DO 10C I=1,NC
STPB 344        II=LM(I)
STPB 345        IF (II.LE.0.CR.II.GT.NEQ) GO TO 100
STPB 346        EX (I)=X (II)
STPB 347        EX1(I)=X1(II)
STPB 348        DEX(I)=DX(II)
STPB 349    100 CONTINUE
STPB 350        DO 20C I=1,12
STPB 351        F (I)=STR(N,I)
STPB 352        UP(I)=UPL(N,I)
STPB 353    200 CCNTINUE
STPB 354  C
STPB 355        IND=NIND(N)
```

```
STPB 356  C
STPB 357        GO TO (1,2,3,4,5) MTYPE
STPB 358  C
STPB 359  C     NONLINEAR TRUSS ELEMENTS--SKIP
STPB 360  C
STPB 361      1 GO TC 60C
STPB 362  C
STPB 363  C     ELASTC-PLASTIC BEAM ELEMENTS
STPB 364  C
STPB 365      2 CALL FREAM (N,INO)
STPB 366        GO TC 600
STPB 367  C
STPB 368  C     NONLINFAR CURVED BEAM ELEMENTS--SKIP
STPB 369  C
STPB 370      3 GO TC 600
STPB 371  C
STPB 372  C     BILINEAR BOUNCARY ELEMENTS
STPB 373  C
STPB 374      4 CALL PBOUND (N,IND)
STPB 375        GO TO 600
STPB 376  C
STPR 377  C     NONLINEAR EXPANSION JOINT ELEMENTS
STPB 378  C
STPB 379      5 M=M+1
STPB 380        CALL NEXPJB (M,INO)
STPB 381  C
STPB 382    600 NIND(N)=INO
STPB 383        IF (INO.NE.0) WRITE (LT) NO,LM,SSA,ASSA,ASA
STPB 384  C
STPE 385        DO 65C I=1,NC
STPB 386        II=LM(I)
STPB 387        IF (II.LE. 0 ) GO TO 650
STPB 388        IF (II.GT.NEC) GO TO 650
STPB 389        DPU(II)=DPU(II)+DF(I)
STPB 390    650 CCNTINUE
STPB 391  C
STPB 392  C     UPDATE NONLINEAR ELEMENT STRESSES AND DEFORMATIONS
STPB 393  C
STPB 394        NSS(N)=NS
STPB 395        DO 700 I=1,12
STPB 396        STR(N,I)=F (I)
STPB 397        UPL(N,I)=UP(I)
STPB 398    700 CONTINUE
STPB 399  C
STPB 400    800 CONTINUF
STPB 401  C
STPB 402        WRITE(6,2C01) NOT,NSTIF,(NIND(I),I=1,NUMNEL)
STPB 403        DO 900 I=1,NEC
STPB 404    9C0 DX(I)=DPU(I)
STPB 405        IF (NCT.NE.NCLT) RETURN
STPB 406  C
STPB 407        WRITE (9) NSS,STR,UPL
STPB 408        RETURN
STPB 409  C
STPB 410   2001 FORMAT (2I5,2X,(30I3))
STPB 411  C
STPB 412        END



STPB 413        SUBRCUTINE ACCB (NIND,A,C,NEQ,NE2B,MBAND,LG,NEGB,
```

```
STPB 414     .                      NBLOCK,NUMNEL,NSTIF,LT)
STPB 415  C
STPB 416        DIMENSION NINC(1),A(NE2B,MBAND),C(NE2B,LG),STIF(1465)
STPB 417        COMMON/EM/ NO,LM(24),SA(12,24),S(24,24),SS(24,24),QQQQ(576)
STPB 418        EQUIVALENCE (STIF,NO)
STPB 419  C
STPB 420        DO 20 I=1,NUMNEL
STPB 421        IF (NINO(I).NE.0) GO TO 50
STPB 422     20 CONTINUE
STPB 423        NSTIF=4
STPB 424        RETURN
STPB 425  C
STPB 426  C     ASSEMBLE NEW GLOBAL EQUILIBRIUM EQUATIONS IN BLOCKS
STPB 427  C
STPB 428     50 NSTIF=2
STPB 429        K=NEQB+1
STPB 430        NSHIFT=0
STPB 431        REWIND 2
STPB 432        REWIND 4
STPB 433        DO 1000 M=1,NBLOCK,2
STPB 434        IF (LG) 100,100,150
STPB 435    100 READ (4) ((A(I,J),I=1,NEQB),J=1,MBAND)
STPB 436        IF (M.EQ.NBLOCK) GO TO 200
STPB 437        READ (4) ((A(I,J),I=K,NE2B),J=1,MBAND)
STPB 438        GO TO 200
STPB 439    150 READ (4) ((A(I,J),I=1,NEQB),J=1,MBAND),
STPB 440     .              ((C(I,L),I=1,NEQB),L=1,LG    )
STPB 441        IF (M.EQ.NBLOCK) GO TO 200
STPB 442        READ (4) ((A(I,J),I=K,NE2B),J=1,MBAND),
STPB 443     .              ((C(I,L),I=K,NE2B),L=1,LG    )
STPB 444    200 CONTINUE
STPB 445  C
STPB 446        REWIND L
STPB 447        DO 700 N=1,NUMNEL
STPB 448        IF (NIND(N).EQ.0) GO TO 700
STPB 449  C
STPB 450        READ (LT) STIF
STPB 451        DO 600 I=1,NC
STPB 452        LMI=LM(I)
STPB 453        IF (LMI.LE. 0 ) GO TO 600
STPB 454        IF (LMI.GT.NEQ) GO TO 600
STPB 455    300 LMN=1-LMI
STPB 456        II=LMI-NSHIFT
STPB 457        IF (II.LE.0.OR.II.GT.NE2B) GO TO 600
STPB 458        DO 500 J=1,NC
STPB 459        JJ=LM(J)
STPB 460        IF (JJ.LE.NEQ) GO TO 350
STPB 461        L=JJ-NEQ
STPB 462        C(II,L)=C(II,L)+SS(I,J)-S(I,J)
STPB 463        GO TO 500
STPB 464    350 JJ=JJ+LMN
STPB 465        IF (JJ) 500,500,400
STPB 466    400 A(II,JJ)=A(II,JJ)+SS(I,J)-S(I,J)
STPB 467    500 CONTINUE
STPB 468    600 CONTINUE
STPB 469    700 CONTINUE
STPB 470  C
STPB 471        IF (LG) 800,800,900
STPB 472    800 WRITE (2) ((A(I,J),I=1,NEQB),J=1,MBAND)
STPB 473        IF (M.EQ.NBLOCK) GO TO 1000
STPB 474        WRITE (2) ((A(I,J),I=K,NE2B),J=1,MBAND)
STPB 475        GO TO 1000
STPB 476    900 WRITE (2) ((A(I,J),I=1,NEQB),J=1,MBAND),
STPB 477     .              ((C(I,L),I=1,NEQB),L=1,LG    )
STPB 478        IF (M.EQ.NBLOCK) GO TO 1000
STPB 479        WRITE (2) ((A(I,J),I=K,NE2B),J=1,MBAND),
STPB 480     .              ((C(I,L),I=K,NE2B),L=1,LG    )
STPB 481   1000 NSHIFT=NSHIFT+NE2B
STPB 482  C
STPB 483        RETURN
STPB 484        END



STPB 485        SUBROUTINE MGMB (X,X2,XG,XT,XM,DPU,RT,R,NFNG,PFNG,PR,NEQ,NEQB,
STPB 486     .                   LG,NFNR,NBLOCK,NDT,NOUT,NSTIF )
STPB 487  C
STPB 488        DIMENSION XM(1),DPU(1),RT(NEQB,LG),R(NEQB,LG),NFNG(LG),PFNG(LG),
STPB 489     .             PR(NFNR),X(1),X2(1),XG(1),XT(1)
STPB 490  C
STPB 491  C     MODIFY UNBALANCED FORCE VECTOR DUE TO CHANGE OF PSEUDO-STATIC COEF
STPB 492  C
STPB 493        REWIND 11
STPB 494        READ (11) NFNG,PFNG
STPB 495        READ (10) PR
STPB 496  C
STPB 497        NS=1
STPB 498        NE=NEQB
STPB 499        NK=NEQB
STPB 500        DO 500 N=1,NBLOCK
STPB 501        READ (11) R
STPB 502        IF (NSTIF.EQ.2) GO TO 250
STPB 503  C
STPB 504        DO 200 I=1,NK
STPB 505        II=NS+I-1
STPB 506        XX=0.
STPB 507        DO 100 L=1,LG
STPB 508        NF=NFNG(L)
STPB 509        IF (NF.LE.0) GO TO 100
STPB 510        XX=XX+R(I,L)*PFNG(L)*PR(NF)
STPB 511    100 CONTINUE
STPB 512    200 XG(II)=XG(II)+XX
STPB 513        GO TO 450
STPB 514  C
STPB 515    250 BACKSPACE 15
STPB 516        READ (15) RT
STPB 517        BACKSPACE 15
STPB 518        DO 400 I=1,NK
STPB 519        II=NS+I-1
STPB 520        XX=0.
STPB 521        YY=0.
STPB 522        DO 300 L=1,LG
STPB 523        NF=NFNG(L)
STPB 524        IF (NF.LE.0) GO TO 300
STPB 525        XX=XX+RT(I,L)*PFNG(L)*PR(NF)
STPB 526        YY=YY-XM(II)*(RT(I,L)-R(I,L))*PFNG(L)*PR(NF)
STPB 527    300 CONTINUE
STPB 528        XG (II)=XG (II)+XX
STPB 529        DPU(II)=DPU(II)+YY
STPB 530    400 CONTINUE
STPB 531  C
STPB 532    450 NS=NS+NEQB
STPB 533        NE=NE+NEQB
```

```
      IF (NE.GT.NEC) NK=NEQ-NS+1
  500 CCNTIAUE
C
      00 6OC I=1,NEC
  600 XT(I)=X2(I)+XC(II
      IF (ACT.EC.NCUT) WRITE (8) (X(II,I=1,NEQ),(XT(II,I=1,NEQ)
      RETURN
      END
```

```
      SUBROUTINE OLTPUT
C
      COMMON/EL PAR/ PAR(14),NUMNP,NELTYP,NUMEL,NUMNEL,NEQ,NEQG,MBAND,
     .               NEQB,NBLOCK,MTOT,N1,N2,N3,N4,N5,N6,N7
      CCMMON/EM/ QCCO(2043)
      CCMMON/MISC/ LG,LL,NFNR,NFA,NGM,NAT,NT,AOT,MSTEP
      COMMON/JUAK/ AOS,OT,KK1,KK2,ISP1,ISP2,NSO,NSS,NNS,MOIS,MSTR,
     .             NOISB,NSTRB,NBO,NBS,JUK(190)
      COMMON A(I)
C
C     INPUT SPECIFICATIONS FOR OUTPUT OF RESPONSE TIPE HISTORIES
C
      N1=1
      N2=N1+NUMNEL*12
      N3=N2+NEC
      N4=N3+NUMEL
      CALL INOUT (A(N1),A(N2I,A(N3),A(N4I,NUMNP,NEQ,NUMEL,NUMNEL)
C
C     PACK RESPCNSE TIME HISTORIES IN BLOCKS
C
      OT=NOT*OT
      NOS=NT/NOT
      NOISB=(MTCT N3-NEQ*2)/(MOIS*2I
      MOISB=(MTCT-N3-9*NOS)/(MOIS*2)
      IF (NOISB.GT.MOISB) NOISB=MOISB
      IF (NCISB.GT.ACSI NOISB=NOS
      NBO=(NOS-1)/NCISB+1
C
      NSTRB=(MTCT-N3-NUMNEL*25)/(MSTR*2)
      MSTRB=(MTCT-N3-9*NOS)/(MSTR*2)
      IF (NSTRB.GT.MSTRB) NSTRB=MSTRB
      IF (NSTRB.GT.AOSI NSTRB=NOS
      NBS=(AOS-1)/ASTRB+1
C
      N4=N3+NEQ
      N5=N4+NEQ
      N6=N5+MOIS*NCISB
C
      N7=N3+NUMNEL
      N8=N7+NUMNEL*12
      N9=N8+NUMNEL*12
      NIO=N5+MSTR*NSTRB
      N11=N10+MSTR*ASTRB
      CALL REPACK (A(N1I,A(N2),A(N3I,A(N4I,A(N5),A(No),
     .             A(N3),A(N7),A(N8),A(N9),A(N10),AOS,NEQ,
     .             MOIS,NCISB,NUMNEL,MSTR,NSTRB,NBO,NBS I
C
      MT=0
      NFILE=0
      IF (KK1.EC.4) NFILE=NFILE+NSO*2
      IF (KK2.EC.4I NFILE=NFILE+ASS+NNS
      IF (NFILE.EC.C) GO TO 100
      MT=30
      REWINO MT
      WRITE (MT I NFILE,NOS,OT
  100 N4=N3+NOS
      N5=N4+NOS*8
      N6=N5+MOIS*NCISB
      N7=N5+MSTR*NSTRB
      IF=0
C
```

```
C     OUTPUT SELECTED (ABSOLUTE) ACCELERATION TIME HISTORY
C
      REWIND 4
      CALL CUTHIS (A(N1),A(N2),A(N3),A(N4),A(N5),A(N6),NUMNEL,
     .            NOS,MDIS,NDISB,NSD,NBD,2,KK1,ISP1,4,12,7,MT,IF)
C
C     OUTPUT SELECTED (RELATIVE) DISPLACEMENT TIME HISTORY
C
      REWIND 4
      CALL CUTHIS (A(N1),A(N2),A(N3),A(N4),A(N5),A(N6),NUMNEL,
     .            NOS,MDIS,NDISB,NSD,NBD,1,KK1,ISP1,4,3 ,7,MT,IF)
C
C     OUTPUT SELECTED STRESS TIME HISTORIES OF LINEAR ELEMENTS
C
      REWIND 7
      CALL CUTHIS (A(N1),A(N2),A(N3),A(N4),A(N5),A(N6),NUMNEL,
     .            NOS,MDIS,NDISB,NSS,NBD,3,KK2,ISP2,4,3 ,7,MT,IF)
C
C     OUTPUT SELECTED STRESS TIME HISTORIES OF NONLINEAR ELEMENTS
C
      REWIND 2
      CALL CUTHIS (A(N1),A(N2),A(N3),A(N4),A(N5),A(N7),NUMNEL,
     .            NOS,MSTR,NSTRB,NNS,NBS,4,KK2,ISP2,2,10,7,MT,IF)
C
      RETURN
      END


      SUBROUTINE INOUT (ISTR,IDIS,INO,IO,NUMNP,NEQ,NUMEL,NUMNEL)
C
      DIMENSION ISTR(NUMNEL,12),IDIS(NEQ),INO(NUMEL),IO(NUMNP,6)
      COMMON/EM/ NS,ND,LM(24),SA(12,24),KLM(8,24),SSA(8,24),QQQQ(1345)
      COMMON/JUNK/ NOS,DT,KK1,KK2,ISP1,ISP2,NSD,NSS,NNS,MDIS,MSTR,
     .                NDISB,NSTRB,NBD,NBS,KD(4,8),SD(8),NKS(4,8),SND(8),
     .                IC(6),IS(12),NDOF,SIG(12)
C
C     INPUT SPECIFICATIONS FOR OUTPUT RESPONSE TIME HISTORY
C
      REWIND 1
      REWIND 2
      REWIND 4
      REWIND 7
      REWIND 8
C
      READ (8) IO
C
      DO 10 I=1,NEQ
   10 IDIS(I)=0
      DO 20 I=1,NUMNEL
      DO 20 J=1,12
   20 ISTR(I,J)=0
C
C     READ AND PRINT DISPLACEMENT OUTPUT SPECIFICATIONS
C
      L=0
      K=0
      READ (5,1000) KK1,ISP1
  100 READ (5,1000) NP,IC
      IF (NP.GT.0) GO TO 110
      IF (L .EQ.0) GO TO 200
      WRITE (4) KD,SD,L
      K=K+1
      GO TO 200
C
  110 IF (K.EQ.0.AND.L.EQ.0)  WRITE(6,2000)
      WRITE(6,2001) NP,IC
      DO 150 I=1,6
      II=IC(I)
      IF (II.LE.0) GO TO 150
      JJ=IO(NP,II)
      IF (JJ.LE.NEQ) GO TO 120
  115 WRITE(6,4001) NP,II
      GO TO 150
  120 IF (JJ.LE.0) GO TO 115
      L=L+1
      KD(1,L)=6
      KD(2,L)=NP
      KD(3,L)=II
      KD(4,L)=JJ
      SD(  L)=0.
      IDIS(JJ)=1
      IF (L.LT.8) GO TO 150
      WRITE (4) KD,SD,L
      K=K+1
      L=0
  150 CONTINUE
      GO TO 100
C
C     READ AND PRINT STRESS OUTPUT SPECIFICATIONS
C
  200 NSD=K
      WRITE(6,4000) KK1,ISP1
C
      READ (8) INO
      L=0
      LL=0
      K=0
      KK=0
      READ (5,1000) KK2,ISP2
      NREAD=1
      NTYPE=0
      NUMNE=0
C
      DO 600 N=1,NUMEL
      MTYPE=INO(N)
      IF (MTYPE.GT.0) GO TO 205
      MTYPE=-MTYPE
      GO TO 210
  205 NUMNE=NUMNE+1
  210 IF (NTYPE.EQ.MTYPE) GO TO 220
      NTYPE=MTYPE
      NUME=0
  220 NUME=NUME+1
      READ (1) NS,ND,LM,SA
      READ (8) NDOF,SIG
      IF (NREAD.EQ.0) GO TO 300
  225 READ (5,1000) NELTYP,NEL,IS
      IF (NEL.GT.0) GO TO 250
      IF ( L .EQ.0) GO TO 230
      WRITE (4) KD,SD,L
      WRITE (7) KLM,SSA,MO
      K=K+1
```

```
  230 IF (LL .EC.0) GU TO 700
      WRITE (2) NKS,SND,LL
      KK=KK+1
      GC TO 700
C
  250 IF (N.EQ.1)  WRITE(6,3000)
      WRITE(6,3002) NELTYP,NEL,IS
C
  300 IF (NELTYP.EQ.NTYPE.AND.NEL.EQ.NUME) GO TO 330
      NREAO=0
      GO TO 600
  350 NREAD=1
      IF ((ND(N)) 4C0,400,500
C
C     STRESSES FCR LINEAR ELEMENTS
C
  400 OO 450 I=1,NS
      I(=IS(()
      IF (II.EU.C) GO TO 450
      L=L+1
      KD(1,L)=NTYPE
      KD(2,L)=NEL
      KD(3,L)=((
      KD(4,L)=C
      SD(  L)=S(G(I()
      OC 440 J=1,NC
      SSA(L,J)=SA(II,J)
      KLM(L,J)=C
      LMJ=LM(J)
      IF (LMJ.GT.NEC) GO TO 440
      IF (LMJ.LF. C ) GO TO 440
      KLM(L,J)=LMJ
      IDIS(LMJ)=1
  440 CCNTINUE
      IF (L.LT.8) GC TO 450
      WRITE (4) KD,SD,L
      WRITE (7) KLM,SSA,ND
      L=0
      K=K+1
  450 CCNTINUE
      GC TC 600
C
C     STRESSES FOP NONLINEAR ELEMENTS
C
  500 DO 55C I=1,NS
      II=IS(I)
      IF (II.EQ.0) GO TO 550
      LL=LL+1
      NKS(1,LL)=NTYPE
      NKS(2,LL)=NEL
      NKS(3,LL)=II
      NKS(4,LL)=NUMNE
      SNO(  LL)=C.
      LL=LL+1
      NKS(1,LL)=NTYPE
      NKS(2,LL)=NEL
      NKS(3,LL)=II
      NKS(4,LL)=NUMNE
      SND(  LL)=0.
      ISTR(NUMNE,II)= 1
      IF (LL.LT.8) GO TO 550
      WRITE (2) NKS,SND,LL
      LL=0
      KK=KK+1
  550 CONTINUE
  600 CONTINUE
      GO TO 225
C
  700 NSS=K
      NNS=KK
      WRITE(6,4C00) KK2,ISP2
      IF (N.GE.NUMEL) GO TO 750
      N=N+1
      OO 720 I=N,NUMEL
      READ (8)
  720 CCNTINUE
C
  750 MDIS=C
      MSTR=0
      OO 8CC I=1,NEC
      IF (ICIS(I).EG.0) GO TO 800
      MOIS=MOIS+1
      IOIS(I)=MCIS
  800 CCNTINUE
C
      DC 90C I=1,NUMNEL
      OC 900 J=1,12
      IF ((STR(I,J).EQ.0) GO TO 900
      MSTR=MSTR+1
      ISTR(I,J)=MSTR
  900 CCNTINUE
C
      RETURN
C
 1000 FCRMAT (14I5)
 2000 FORMAT (36H1CISPLACEMENT  COMPONENTS  FOR WHICH  /
     .            36H CUTPUT TIME HISTCRY IS REQUIRED....  //
     .             5H NCDE 4X 23HOISPLACEMENT COMPONENTS /)
 2001 FORMAT (I5,4X,6I4)
 3000 FORMAT (36H1ELEMENT STRESS COMPCNENTS FOR WHICH /
     .            36H CUTPUT TIME HISTORY IS REQUIREO....  //
     .            4CH  ELEMENT      CESIREO STRESS COMPCNENTS /
     .            1CH TYPE  NO.   /)
 3002 FORMAT (2I4,5X,12I3)
 4000 FORMAT (/16H CUTPUT TYPE.... ,I2 /
     .             16H PLOT SPACING... ,I2 )
 4001 FORMAT (I5,4X,I4,4X,21HFIXED DCF...NO OUTPUT )
      END


      SUBROLTINE REPACK (ISTR,IDIS,X,X2,XH,X2H,NSS,STR,UPL,STH,UPH,
     .                    NOS,NEQ,MDIS,NOISB,NUMNEL,MSTR,NSTFB,NBD,NBS)
C
      DIMENSION ISTR(NUMNEL,1),IDIS(1),X(NEQ),XH(MCIS,NCISB),
     .               NSS(NUMNEL),STR(NUMNEL,12),UPL(NUMNEL,12),
     .               STH(MSTR,NSTRB),UPH(MSTR,NSTRB),X2(NEQ),X2H(MDIS,NDISB)
C
C     PACK RESPCNSE TIME HISTORY IN OPTIMAL AVAILABLE STORAGE BLOCKS
C
      REWINO 3
      REWINC 9
      REWINC 10
```

```
      REWIND 12
      L=0
      K=0
      DO 200 N=1,NOS
      READ (8) X,X2
      L=L+1
      DO 100 I=1,NEC
      II=IDIS(I)
      IF (II) 100,100,90
   90 XH (II,L)=X (I)
      X2H(II,L)=X2(I)
  100 CONTINUE
      IF (L.LT.NOISB) GO TO 200
      WRITE (3)  L,XH
      WRITE (12) L,X2H
      K=K+1
      L=0
  200 CONTINUE
      IF (L) 220,220,210
  210 WRITE (3)  L,XH
      WRITE (12) L,X2H
      K=K+1
  220 IF (NBD.NE.K) NBD=K
C
      L=0
      K=0
      IF (NUMNEL.LE.0) GO TO 420
      DO 400 N=1,NCS
      READ (9) NSS,STR,UPL
      L=L+1
      DO 300 I=1,NUMNEL
      NS=NSS(I)
      DO 250 J=1,NS
      II=ISTR(I,J)
      IF (II) 250,250,240
  240 STH(II,L)=STR(I,J)
      UPH(II,L)=UPL(I,J)
  250 CONTINUE
  300 CONTINUE
      IF (L.LT.NSTPE) GO TO 400
      WRITE (10) L,STH,UPH
      L=0
      K=K+1
  400 CONTINUE
      IF (L) 420,420,410
  410 WRITE (10) L,STH,UPH
      K=K+1
  420 IF (NBS.NE.K) NBS=K
C
      RETURN
      END


      SUBROUTINE OUTHIS (ISTR,IDIS,T,X,XH,UH,NEL,NCS,NOI,NOJ,
     *                   NOB,NHB,KKK,KKI,ISPI,IT,JT,KT,MT,IF)
C
      DIMENSION ISTR(NEL,1),IDIS(1),T(1),X(8,NOS),XH(NOI,NOJ),
     *          UH(NOI,NOJ)
      COMMON/EM/ KLM(8,24),SA(8,24),ND,SM(8)
      COMMON/JUNK/ NPT,DT,NDATA(13),L,K,KO(4,8),XD(8),TM(8),XM(8)
      DATA SM /1H ,1HA,1H2,1HB,1H3,1HC,1H4,1HD/
C
C     OUTPUT RESPONSE TIME HISTORIES ON SPECIFIED DISPLAY MEDIUM
C
C     TAPE IT   INPUT  TAPE    STORE KO(4,8),XD(8),L
C     TAPE JT   INPUT  TAPE    STORE XH(NOI,NOJ),UH(NOI,NOJ),K
C     TAPE KT   INPUT  TAPE    STORE KLM(8,24),SA(8,24),ND
C     TAPE MT   OUTPUT TAPE    STORE IF,KKK,L,KD,XP,X(8,NCS)
C
      IF (NCB.EQ.0) RETURN
C
      DO 900 M=1,NCB
      IF=IF+1
      REWIND JT
      READ (IT) KO,XO,L
      IF (KKK.EQ.3) READ (KT) KLM,SA,ND
      DO 100 I=1,8
      TM(I)=0.
      XM(I)=0.
  100 CONTINUE
C
C     PRINT APPROPRIAT TITLE
C
      GO TO (110,200,130,110) KKI
C
  110 GO TO (111,112,113,114) KKK
  111 WRITE(6,1001) M,IF
      WRITE(6,1002) (KO(2,I),KO(3,I),I=1,L)
      GO TO 200
  112 WRITE(6,2001) M,IF
      WRITE(6,1002) (KO(2,I),KO(3,I),I=1,L)
      GO TO 200
  113 WRITE(6,3001) M,IF
      WRITE(6,3002) (KO(1,I),KO(2,I),KO(3,I),I=1,L)
      GO TO 200
  114 WRITE(6,4001) M,IF
      WRITE(6,4002) (KO(1,I),KO(2,I),XD(3,I),I=1,L)
      GO TO 200
C
  130 IF (M.GT.1) GO TO 200
      GO TO (131,132,133,131) KKK
  131 WRITE(6,1003)
      WRITE(6,1010)
      GO TO 200
  132 WRITE(6,2003)
      WRITE(6,2010)
      GO TO 200
  133 WRITE(6,3003)
      WRITE(6,3010)
C
C     ARRANGE RESPONSE TIME HISTORIES IN OUTPUT FORM
C
  200 TT=0.
      N =0
      DO 600 NB=1,NHB
      GO TO (210,210,210,220) KKK
  210 READ (JT) K,XH
      GO TO 250
  220 READ (JT) K,XH,UH
  250 DO 500 J=1,K
      N=N+1
      TT=TT+DT
```

```
      MM=-1
      DO 400 I=1,L
C
      GO TO (310,310,330,340) KKK
C
  310 JJ=KD(4,I)
      II=IDIS(JJ)
      XX=XH(II,J)
      GO TO 350
C
  330 XX=XD(I)
      DO 335 KK=1,NC
      JJ=KLM(I,KK)
      IF (JJ) 335,335,334
  334 II=IDIS(JJ)
      XX=XX+SA(I,KK)*XH(II,J)
  335 CONTINUE
      GO TO 350
C
  340 JJ=KD(3,I)
      NN=KD(4,I)
      II=ISTR(NN,JJ)
      IF (MM) 341,341,342
  341 XX=XH(II,J)
      GO TO 345
  342 XX=UH(II,J)
  345 MM=-1*MM
C
  350 AX=ABS(XX)
      IF (AX-XM(I)) 370,370,360
  360 XM(I)=AX
      TM(I)=TT
  370 X (I,N)=XX
      T ( N )=TT
  400 CONTINUE
C
  500 CONTINUE
  600 CONTINUE
C
      GO TO (610,620,630,640) KKI
C
C     PRINT RESPONSE TIME HISTORY
C
  610 DO 611 N=1,NDS
  611 WRITE(6,1004) T(N),(X(I,N),I=1,L)
      WRITE(6,1005) (XM(I),I=1,L)
      WRITE(6,1006) (TM(I),I=1,L)
      GO TO 900
C
C     PLOT RESPONSE TIME HISTORY
C
  620 GO TO (621,623,625,627) KKK
C
  621 ISO=1
      WRITE(6,1008) M
      WRITE(6,1010)
      GO TO 624
  623 ISO=1
      WRITE(6,2008) M
      WRITE(6,2010)
  624 WRITE(6,2011) (KD(2,I),KD(3,I),XM(I),TM(I),I,I=1,L)
      GO TO 629
  625 ISO=1
      WRITE(6,3008) M
      WRITE(6,3010)
      WRITE(6,3011) (KD(1,I),KD(2,I),KD(3,I),XM(I),TM(I),I,I=1,L)
      GO TO 629
  627 ISO=2
      WRITE(6,4009) M
      WRITE(6,3010)
      DO 628 I=1,L,2
      II=1+I
      WRITE(6,4011) KD(1,I),KD(2,I),KD(3,I),XM(I),TM(I),SM(I)
  628 WRITE(6,4012) XM(II),TM(II),SM(II)
C
  629 CALL PLOT (X,XM,L,DT,NDS,ISPI,ISO)
      GO TO 900
C
C     PRINT MAXIMUM RESPONSE ONLY
C
  630 GO TO (631,631,635,637) KKK
  631 WRITE(6,1007) (KD(1,I),KD(2,I),XM(I),TM(I),I=1,L)
      GO TO 900
  635 WRITE(6,3007) (KD(1,I),KD(2,I),KD(3,I),XM(I),TM(I),I=1,L)
      GO TO 900
  637 DO 638 I=1,L,2
      II=I+1
      WRITE(6,4007) KD(1,I),KD(2,I),KD(3,I),XM(I),TM(I)
  638 WRITE(6,4008) XM(II),TM(II)
      GO TO 900
C
C     STORE RESPONSE TIME HISTORIES ON OUTPUT TAPE MT
C
  640 IF (KKI.EQ.4) WRITE (MT) IF,KKK,L,KD,XM,X
      GO TO 610
C
  900 CONTINUE
      RETURN
C
 1001 FORMAT (50H1TIME HISTORY FOR SELECTED DISPLACEMENT COMPONENTS ,
     .  5H..... ,I2,37X,8HFILE NO.,I3 //
     .                     20X,40HNODE NUMBERS AND DISPLACEMENT COMPONENTS  )
 1002 FORMAT (8H    TIME,2X,8(I8,1H-,I2,X))
 1003 FORMAT (55H1MAXIMUM DISPLACEMENT VALUES FROM DYNAMIC RESPONSE ANAL
     .YSIS // )
 1004 FORMAT (0PF8.3,2X,1P8E12.3)
 1005 FORMAT (/24H MAXIMUM ABSOLUTE VALUES  /10H MAXIMUM   ,1P8E12.3)
 1006 FORMAT (10H TIME       ,1P8E12.3)
 1007 FORMAT (I6,I12,1PE18.3,E12.3,5X,2HN/ )
 1008 FORMAT (59H1NORMALIZED PLOT OF DISPLACEMENT RESPONSE TIME HISTORIE
     .S... ,I3 /)
 1010 FORMAT (57H   NODE     DISPLACEMENT    MAXIMUM       TIME AT    PLO
     .T       /58H  NUMBER    COMPONENT        VALUES        MAXIMUM    SYM
     .BOL   )
 2001 FORMAT (50H1TIME HISTORY FOR SELECTED ACCELERATION COMPONENTS ,
     .  5H..... ,I3,37X,8HFILE NO.,I3 //
     .                     20X,40HNODE NUMBERS AND ACCELERATION COMPONENTS  )
 2003 FORMAT (55H1MAXIMUM ACCELERATION VALUES FROM DYNAMIC RESPONSE ANAL
     .YSIS // )
 2008 FORMAT (59H1NORMALIZED PLOT OF ACCELERATION RESPONSE TIME HISTORIE
     .S... ,I3 /)
 2010 FORMAT (57H   NODE     ACCELERATION    MAXIMUM       TIME AT    PLO
     .T       /58H  NUMBER    COMPONENT        VALUES        MAXIMUM    SYM
     .BOL   )
```

```
 2011 FORMAT (I6,I13,1PE18.3,E12.3,I6)
 3001 FORMAT (46H1TIME HISTORIES FOR SELECTED STRESS COMPONENTS ,5H.....
     .    I3,41X,8HFILE NO.,I3 //
     .             2CX,48HELEMENT TYPE - ELEMENT NUMBER - STRESS COMPONENT )
 3002 FORMAT (8H    TIME ,2X,8(I4,2H -,I3,1H-,I2))
 3003 FORMAT (53H1MAXIMUM STRESS VALUES FROM DYNAMIC RESPONSE ANALYSIS
     .  //)
 3007 FORMAT (I6,I6,I10,1PE18.3,E12.3,5X,2HNA)
 3008 FORMAT (53H1NORMALIZED PLOT OF STRESS RESPONSE TIME HISTURIES...
     .             I3/)
 3010 FORMAT (13H ELEMENT      2X 6HSTRESS 10X 7HMAXIMUM 6X 7HTIME AT 4X
     .4HPLOT /13H TYPE  NUMBER 2X 9HCOMPONENT 7X 7HVALUES  6X 7HMAXIMUM
     .4X 6HSYMBOL )
 3011 FORMAT (I6,I6,I10,1PE18.3,E12.3,I6)
 4001 FORMAT (47H1TIME HISTORIES FOR SELECTED NONLINEAR STRESSES  /
     . 6X,46H AND CORRESPONDING NONLINEAR DEFORMATIONS  .....  I3,
     . 38X,8HFILE NO.,I3 //
     .20X,48HELEMENT TYPE - ELEMENT NUMBER - STRESS COMPONENT  )
 4002 FORMAT (12X,4(12H STRESS     ,12HDEFORMATION )/
     .             8H    TIME,2X,8(I4,2H -,I3,1H-,I2))
 4003 FORMAT (68H1MAXIMUM NONLINEAR STRESSES AND CORRESPONDING NONLINEAR
     . DEFORMATIONS //)
 4007 FORMAT (2I6,I10,1PE18.3,E12.3,2X,6HSTRESS )
 4008 FORMAT (22X,1PE18.3,E12.3,2X,11HDEFORMATION )
 4009 FORMAT (60H1NORMALIZED PLOT OF NONLINEAR STRESS RESPONSE TIME HIST
     .ORIES /58H AND CORRESPONDING NONLINEAR DEFORMATION TIME HISTORIES.
     ...   I3/)
 4011 FORMAT (I6,I6,I10,1PE18.3,E12.3,5X,A1)
 4012 FORMAT (22X,1PE18.3,E12.3,5X,A1)
      END


      SUBROUTINE PLOT (X,XM,L,DT,NOS,ISP,ISD)
C
      DIMENSION X(8,1),XM(8),PM(8),QM(8),SM(8),IP(8)
      COMMON/EM/ PP(101)
      DATA PM /1H1,1H2,1H3,1H4,1H5,1H6,1H7,1H8/
      DATA QM /1H1,1HA,1H2,1HB,1H3,1HC,1H4,1HD/
      DATA BL,V,P /1H ,1H.,1H+/
C
C     NORMALIZED PLOT OF TIME HISTORIES OF TIME FUNCTIONS
C
      DO 30 I=1,L
      GO TO (10,20) ISD
   10 SM(I)=PM(I)
      GO TO 30
   20 SM(I)=QM(I)
   30 CONTINUE
C
      DO 100 I=1,L
      IF (XM(I)) 50,100,50
   50 XM(I)=50./XM(I)
  100 CONTINUE
      TT=0.
      WRITE(6,2000)
      WRITE(6,2001)
      WRITE(6,2002) TT,P,(V,I=1,24),P,(V,I=1,24),P,(V,I=1,24),P,
     .                          (V,I=1,24),P,TT
C
      K=1
      DO 200 I=2,100
  200 PP(I)=BL
      NOS1=NOS-1
      DO 500 N=1,NOS1
      PP(  1)=V
      PP( 51)=V
      PP(101)=V
      II=ISP
  210 IF (II.LE.0) GO TO 250
      WRITE(6,2003) PP
      II=II-1
      GO TO 210
  250 TT=TT+DT
      DO 300 I=1,L
      XX=XM(I)*X(I,N)
      M=XX
      M=M+51
      IP(I)=M
  300 PP(M)=SM(I)
      IF (K.LT.10) GO TO 320
      PP(  1)=P
      PP( 51)=P
      PP(101)=P
      K=1
      WRITE(6,2002) TT,PP,TT
      GO TO 340
  320 WRITE(6,2003) PP
      K=K+1
C
C     RESET PP(101)
C
  340 DO 360 I=1,L
      M=IP(I)
  360 PP(M)=BL
  500 CONTINUE
C
      TT=TT+DT
      WRITE(6,2002) TT,P,(V,I=1,24),P,(V,I=1,24),P,(V,I=1,24),P,
     .                          (V,I=1,24),P,TT
      WRITE(6,2001)
      RETURN
C
C     *CARRIAGE CONTROL CHACTER 1 SURPRESS SKIP OVER PAGE FOLDS
C
 2000 FORMAT (/56X,8HORDINATE )
 2001 FORMAT (/12H1  TIME -1.0,21X,4H-0.5,22X,3H0.0,22X,3H0.5,22X,3H1.0,
     .            X,4HTIME )
 2002 FORMAT (1H1,F7.2,2X,101A1,F7.2)
 2003 FORMAT (1H1,5X,101A1)
      END
```

```
      SUBROUTINE TRUSS
C
      COMMON A(1)
      COMMON/ELPAR/ NPAR(14),NUMNP,NELTYP,NUMEL,NUMMEL,NEQ,NEQG,MBAND,
     .              NEQB,NBLOCK,MTOT,N1,N2,N3,N4,N5,N6
      COMMON/JUNK/ STR(4),MM,L,K,NTAG,NOYN,SIG(12),EXRA(184)
C
      IF (NPAR(1).EQ.0) GO TO 500
C
      N7=N6+NPAR(3)
      N8=N7+NPAR(3)
      N9=N8+NPAR(3)
      N10=N9+NPAR(3)
      N11=N10+NPAR(3)
      N12=N11+NPAR(4)*4
      IF (N12.GT.MTOT) CALL ERROR (1,N12-MTOT,7HTRUSS  )
      CALL RUSS (A(N1),A(N2),A(N3),A(N4),A(N5),A(N6),A(N7),A(N8),A(N9),
     .           A(N10),A(N11),NUMNP,NPAR(4))
      RETURN
C
  500 IF (NTAG.EQ.0) WRITE(6,2002)
      WRITE(6,3002) MM,L,SIG(1),SIG(2)
      NTAG=1
      RETURN
C
 2002 FORMAT (//26H0 ....TRUSS MEMBER ACTIONS //
     .           46H0 MEMBER    LOAD           STRESS           FORCE  )
 3002 FORMAT (2I8,F15.5,F15.3)
C
      END
```

```
      SUBROUTINE RUSS (ID,X,Y,Z,INO,E,THERM,DEN,AREA,WT,ENPROP,NUMNP,
     .                 NUMNPAR)
C
      DIMENSION ID(NUMNP,1),X(1),Y(1),Z(1),INO(1),E(1),THERM(1),DEN(1),
     .          AREA(1),WT(1),ENPROP(NUMNPAR,1)
      COMMON/ELPAR/ NPAR(14),NNNNN,NELTYP,NUMEL,NUMMEL,NEQ,NEQG,MBAND,
     .              NEQB,NBLOCK,MTOT,N1,N2,N3,N4,N5,N6
      COMMON/EM/ LM(24),ND,NS,S(24,24),P(24,4),XM(24),ST(12,24),TT(12,4)
     .           ,QQQQ(984)
      COMMON /JUNK/ EMUL(4,4),I,J,K,L,M,N,II,JJ,KK,MTYPE,TEMP,DX,DY,DZ,
     . XL2,XL,XX,YY,F,FT,FX,FY,FZ,MIN,MAX,NOIF,KKK,TEM,MTYP
C
C     CONTROL INFORMATION AND MEMBER PROPERTIES
C
      NUME=NPAR(2)
      NUMMAT=NPAR(3)
      WRITE (6,2000) NUME,NUMMAT,NUMNPAR
      WRITE (6,2001)
      DO 10 I=1,NUMMAT
      READ  (5,1001) N,E(N),THERM(N),DEN(N),AREA(N),WT(N)
   10 WRITE (6,2002) N,E(N),THERM(N),DEN(N),AREA(N),WT(N)
C
C     ELEMENT LOAD MULTIPLIERS
C
      READ (5,1003)  EMUL
      WRITE (6,2003) EMUL
C
C     READ AND PRINT ELEMENT NONLINEAR PARAMETERS
C
      IF (NUMNPAR.EQ.0) GO TO 50
      WRITE(6,2010)
      DO 30 I=1,NUMNPAR
   30 READ (5,1005) N,(ENPROP(N,J),J=1,4)
      WRITE(6,2011) (N,(ENPROP(N,J),J=1,4),N=1,NUMNPAR)
   50 CONTINUE
C
C     ELEMENT INFORMATION
C
      WRITE (6,2005)
      N=1
  100 READ (5,1004)  M,II,JJ,MTYP,TEM,ININO,KK
      IF (ININO.GT.NUMNPAR) CALL ERROR (3,M,7HTRUSS  )
      IF(KK.EQ.0) KK=1
  120 IF(M.NE.N) GO TO 200
      I=II
      J=JJ
      MTYPE=MTYP
      TEMP=TEM
      NINO=ININO
      KKK=KK
C
C     1. FORM ELEMENT STIFFNESS AND STRESS MATRICES
C
  200 DX=X(I)-X(J)
      DY=Y(I)-Y(J)
      DZ=Z(I)-Z(J)
      XL2=DX*DX+DY*DY+DZ*DZ
      XL=SQRT(XL2)
      XX=E(MTYPE)*AREA(MTYPE)*XL
      ST(1,1)=DX/XL2
      ST(1,2)=DY/XL2
      ST(1,3)=DZ/XL2
      ST(1,4)=-ST(1,1)
      ST(1,5)=-ST(1,2)
      ST(1,6)=-ST(1,3)
C
      DO 300 L=1,6
      YY=ST(1,L)*XX
      DO 250 K=L,6
      S(K,L)=ST(1,K)*YY
  250 S(L,K)=S(K,L)
      ST(1,L)=E(MTYPE)*ST(1,L)
  300 ST(2,L)=AREA(MTYPE)*ST(1,L)
C
C     2. INERTIA AND THERMAL LOADS
C
      F=WT(MTYPE)*XL/2.
      FT=TEMP*THERM(MTYPE)*E(MTYPE)*AREA(MTYPE)
      FX=DX*FT/XL
      FY=DY*FT/XL
      FZ=DZ*FT/XL
C
      DO 350 L=1,4
      TT(2,L)=EMUL(L,4)*FT
      TT(1,L)=TT(2,L)/AREA(MTYPE)
      P(1,L)=EMUL(L,1)*F-EMUL(L,4)*FX
      P(2,L)=EMUL(L,2)*F-EMUL(L,4)*FY
      P(3,L)=EMUL(L,3)*F-EMUL(L,4)*FZ
      P(4,L)=EMUL(L,1)*F+EMUL(L,4)*FX
```

```
      P(5,L)=EMUL(L,2)*F+EMUL(L,4)*FY
  350 P(6,L)=EMUL(L,3)*F+EMUL(L,4(*FZ
      F=DEN(MTYPE(*AREA(MTYPE)*XL/2.
      DO 375 L=1,6
  375 XM(L)=F
C
C     3. FORM LOCATION MATRIX AND COMPUTE BAND WIDTH
C
      DO 400 L=1,3
      LM(L)=(D(I,L)
  400 LM(L+3)=IC(J,L(
C
      NO=6
      NS=2
      CALL WRITET(MBAND,NDIF(
C
C     4. INPUT NONLINEAR ELEMENT PROPERTIES
C
      NN=NUMEL+N
      (F (NINO.EQ.C) GO TO 500
      (NO(NN)=1
      WRITE(6,2100(
      STOP
C
  500 IND(NN)=-1
C
C     5. CHECK FOR MORE ELEMENTS
C
      WR(TE (6,2004( N,(,J,MTYPE,TEMP,NDIF,NINO
      (F (N.EQ.NUME) RETURN
      N=N+1
      (=(+KKK
      J=J+KKK
      IF(N.GT.M) GC TO 100
      GC TC 120
C
 1001 FORMAT (I5,5F(0.0)
 1003 FCRMAT (4F10.0)
 1004 FORMAT (4I5,1F10.0,2I5)
 1005 FORMAT (I5,4F10.0)
 2000 FORMAT (38H1THREE DIMENSIONAL TRUSS ELEMENTS..... ///
     . 25H NUMBER CF TRUSS MEMBERS= I5//
     1 25H NUMBER CF DIFF. MEMBERS= (5//
     2 25H NUMBER CF NPAR SETS     = (5)
 2001 FORMAT (///1X,4HTYPE,14X,1HE,10X,5HALPHA,12X,3HOEN,11X,4HAREA
     1 11X,4HWT/L )
 2002 FORMAT (I5,5E15.7)
 2003 FORMAT (///30H ELEMENT LOAD MULTIPL(ERS...../12X,1HA,14X,1HB,14X,
     .  1HC,14X,1HD /6H X-DIR 4E15.6/6H Y-DIR 4E15.6/6H Z-DIR 4E15.6/
     .  6H  TEMP 4E15.6 )
 2004 FORMAT (4I6,F10.2,2I7)
 2005 FORMAT (///42H1    N      (      J  TYPE      TEMP   BAND 7H    IND)
 2010 FORMAT (1H1,35HTRUSS ELEMENT NONLINEAR PARAMETERS.....  //
     .  8H NL PAR. 4X 8HAXIAL-C  4X 8HAXIAL-T  4X 8HPLASTIC  4X
     .  8HPLASTIC /8H    NO  4X 8H    PU    4X 8H    PU    4X 8HC-STIF
     ,  4X 8HT STIF  /)
 2011 FORMAT (I8,4E12.3)
 2100 FORMAT (1H1,65HNONLINEAR TRUSS ELEMENT IS NOT READY YET....EXECUTI
     .ON TERMINATED. )
       END
```

```
      SUBROUTINE BEAM
C
      COMMON A(1)
      COMMON/ELPAR/ NPAR(14),NUMNP,NELTYP,NUMEL,NUMMEL,NEQ,NEQG,MBAND,
     .                 NEQB,NBLOCK,MTOT,N1,N2,N3,N4,N5,N6
      COMMON/JUNK/ STR(4),MM,L,K,NTAG,NO(N,SIG(12),EXRA(184)
C
      IF (NPAR(1).EQ.0) GO TO 500
      N7=N6+NPAR(5)
      N8=N7+NPAR(5)
      N9=N8+NPAR(5)
      N10=N9+NPAR(4)*12
      N11=N10+NPAR(3)*6
      N12=N11+NPAR(6)*12
      IF (N12.GT.MTOT) CALL ERROR (1,N12-MTOT,7HBEAM    )
      CALL TEAM (NPAR(2),NPAR(3),NPAR(4),NPAR(5),NPAR(6),
     .              A(N1),A(N2),A(N3),A(N4),A(N5),A(N6),A(N7),A(N8),
     .              A(N9),A(N10),A(N11),NUMNP,MBAND,NUMEL)
C
      RETURN
C
  500 IF(NTAG.EQ.0) WR(TE (6,2002)
      WRITF(6,3002) MM,L,(SIG(I),I=1,12)
      NTAG=1
      RETURN
C
 2002 FORMAT(/25H0 ....BEAM FORCES AND MOMENTS//
     . 10H0BEAM LOAC 5X 5HAXIAL  2(7X,5HSHEAR),5X 7HTORSION
     .  2(5X,7HBENDING)/ 10H  NO.  NO.  8X  2HP1  10X  2HP2  10X
     .  2HP3 10X 2HM1 10X 2HM2 10X 2HM3)
 3002 FORMAT (I5,I4,1PE11.3,5E12.3/8X,6E12.3/)
C
      END
```

```
      SUBROUTINE TEAM (NBEAM,NUMETP,NUMFIX,NUMMAT,NUMNPAR,ID,X,Y,Z,IND,
     .                 E,G,RO,SF,COPROP,ENPROP,NUMNP,MBAND,NUMEL)
C
      DIMENSION (D(NUMNP,1),X(1),Y(1),Z(1),IND(1),E(1),G(1),RO(1),
     .          SF(NUMFIX,1),COPROP(NUMETP,1),ENPROP(NUMNPAR,1),
     .          STIF(72),LMS(12),EC(12),T(3,3)
      COMMON/EM/ LM(24),ND,NS,ASA(24,24),RF(24,4),XM(24),SA(12,24),
     .           SF(12,4),ENPAR(24),S(12,12),F(48),TTT(72)
      COMMON/JUNK/ LC(4),JK(6),MELTYP,DL,MATTYP,ILC(4),TS(2,2),LS(4),
     .            FMUL(3,4)
      EQUIVALENCE (STIF,LM),(LMS,TTT(1)),(EC,TTT(13)),(T,TTT(25))
C
C     FORMS 3-D BEAM STIFFNESS AND STRESS ARRAYS
C
      WRITE (6,2005) NBEAM,NUMETP,NUMFIX,NUMMAT,NUMNPAR
      DO 5 I=1,1346
    5 STIF(I)=0.
C
C     READ AND PRINT MATERIAL PROPERTY DATA
C
      WRITE (6,2001)
      DO 10 I=1,NUMMAT
      READ  (5,1001) N,E(N),G(N),RO(N)
      WRITE (6,2002) N,E(N),G(N),RO(N)
```

```
   10 G(N)=0.5*E(N)/(1.+G(N))
C
C     READ AND PRINT GEOMETRIC PROPERTIES OF COMMON ELEMENTS.
C
      WRITE (6,2003)
      DO 30 I=1,NUMETP
      READ  (5,1002) N,(COPROP(N,J),J=1,6)
      IF((COPROP(N,1).NE.0.0).AND.(COPROP(N,4).NE.0.0).AND.
     1   (COPROP(N,5).NE.0.0).AND.(COPROP(N,6).NE.0.0))  GO TO 20
      WRITE (6,2013)
      CALL EXIT
   20 WRITE (6,2004) N,(COPROP(N,J),J=1,6)
   30 CONTINUE
C
C     ELEMENT LOAD MULTIPLIERS
C
      READ  (5,1006) ((EMUL(I,J),J=1,4),I=1,3)
      WRITE (6,2006) ((EMUL(I,J),J=1,4),I=1,3)
C
C     READ AND PRINT FIXED END FORCES IN LOCAL COORDINATES
C
      IF(NUMFIX .EQ. 0) GO TO 56
      WRITE (6,2010)
      DO 55 I=1,NUMFIX
      READ  (5,1005) N,(SFT(N,J),J=1,12)
   55 WRITE (6,2011) N,(SFT(N,J),J=1,12)
   56 CONTINUE
C
C     READ AND PRINT ELEMENT NONLINEAR PARAMETERS
C
      IF (NUMNPAR.EQ.0) GO TO 59
      WRITE(6,2020)
      DO 58 I=1,NUMNPAR
   58 READ (5,1007) N,(ENPROP(N,J),J=1,12)
      WRITE(6,2021) (N,(ENPROP(N,J),J=1,12),N=1,NUMNPAR)
   59 CONTINUE
C
C     READ AND PRINT ELEMENT DATA. GENERATE MISSING INPUT.
C
      WRITE (6,4000)
      L=0
   60 KKK=0
      READ (5,3000) INEL,INI,INJ,INK,IMAT,IMEL,ILC,INELKI,INELKJ,INC,IDD
      IF (IDD.GT.NUMNPAR) CALL ERROR (3,INEL,7HBEAM   )
      IF (INEL.NE.1) GO TO 15
      NI=INI
      NJ=INJ
      NK=INK
   15 IF (INC.EQ.0) INC=1
   65 L=L+1
      KKK=KKK+1
      ML=INEL-L
      IF (ML)  66,67,68
   66 CALL ERROR (4,INEL,7HBEAM   )
   67 NEL=INEL
      NI=INI
      NJ=INJ
      NK=INK
      MATTYP=IMAT
      MELTYP=IMEL
      DO 90 I=1,4
   90 LC(I)=ILC(I)
      NEKODI=INELKI
      NEKODJ=INELKJ
      NIND=IDD
      GO TO 69
   68 NEL=INEL-ML
      NI=IN+KKK*INCR
      NJ=JN+KKK*INCR
   69 CONTINUE
      WRITE (6,4001) NEL,NI,NJ,NK,MATTYP,MELTYP,LC,NEKODI,NEKODJ,NIND
C
   74 DX=X(NJ)-X(NI)
      DY=Y(NJ) Y(NI)
      DZ=Z(NJ)-Z(NI)
      DL=SQRT(DX*DX+DY*DY+DZ*DZ)
      IF(DL) 75,75,76
   75 CALL ERROR (5,NEL,7HBEAM   )
C
C     FORM GLOBAL TO LOCAL COORDINATE TRANSFORMATION.
C
   76 T(1,1)=DX/DL
      T(1,2)=DY/DL
      T(1,3)=DZ/DL
C
      A1=X(NJ)-X(NI)
      A2=Y(NJ)-Y(NI)
      A3=Z(NJ)-Z(NI)
      B1=X(NK)-X(NI)
      B2=Y(NK)-Y(NI)
      B3=Z(NK)-Z(NI)
      AA=A1*A1+A2*A2+A3*A3
      AB=A1*B1+A2*B2+A3*B3
      U1=AA*B1 AB*A1
      U2=AA*B2- AB*A2
      U3=AA*B3- AB*A3
      UU=U1*U1+U2*U2+U3*U3
      UU=SQRT(UU)
      IF (UU.GT.0.) GO TO 77
      CALL ERROR (6,INEL,7H BEAM  )
   77 T(2.1)=U1/UU
      T(2,2)=U2/UU
      T(2,3)=U3/UU
      T(3,1)=T(1,2)*T(2,3)-T(1,3)*T(2,2)
      T(3,2)=T(1,3)*T(2,1)-T(1,1)*T(2,3)
      T(3,3)=T(1,1)*T(2,2)-T(1,2)*T(2,1)
C
C     CHECK IF NEW STIFFNESS NEEDED
C
      IF (NEL.EQ.1) GO TO 80
      IF  (ABS(DS-DL).GT.DL/100.)  GO TO 80
      IF ((MT.NE.MATTYP).OR.(ME.NE.MELTYP)) GO TO 80
      IF ((JK(1).NE.NEKODI).OR.(JK(2).NE.NEKODJ)) GO TO 80
      DO 78 I=1,4
      IF (LS(I).NE.LC(I)) GO TO 80
   78 CONTINUE
      DO 79 I=1,2
      DO 79 J=1,2
      IF (ABS(TS(I,J)-T(I,J)).GT.ABS(T(I,J)/100.)) GO TO 80
   79 CONTINUE
      GO TO 150
C
   80 DS=DL
      MT=MATTYP
```

```
      ME=MELTYP
      DO 81 I=1,2
      DO 81 J=1,2
   81 TS(I,J)=T(I,J)
      DO 82 I=1,4
   82 LS(I)=LC(I)
      JK(1)=NEKCOI
      JK(2)=NEKCOJ
C
C     FORM NEW STIFFNESS
C
      CALL NEWBM (E,G,RO,COPROP,SFT,NUMFIX,NUMETP)
C
C     FORM ELEMENT LOCATION MATRIX   .
C
  150 DO 170 M=1,6
      LM(M)=ID(NI,M)
      LM(M+12)=0
      LM(M+18)=0
  170 LM(M+6)=ID(NJ,M)
      NS=12
      ND=12
C
C     TRANSFORM TO MASTER DEGREES OF FREEDOM
C
      CALL SLAVE (X,Y,Z,ID,NUMNP,NI,NJ)
C
C     WRITE ELEMENT INFORMATION ON TAPE
C
      CALL WRITET (MBAND,NOIF)
C
C     SET NONLINEAR ELEMENT INDICATOR AND STORE NONLINEAR INFORMATION
C
      NN=NL*FL+NEL
      IF (NIND.LE.0) GO TO 200
      IND(NN)=2
      DO 190 I=1,12
  190 ENPAR(I)=ENPROP(NIND,I)
      MTYPE=2
      WRITE (3) MTYPE,LM,ND,NS,ASA,SA,S,ENPAR,TT)
      GO TO 250
  200 IND(NN)=-2
C
C     CHECK FOR LAST ELEMENT
C
  250 IF (NBEAM-NEL) 66,500,260
  260 IF (ML.GT.0)  GO TO 65
      IN=INI
      JN=INJ
      INCN=INC
      GO TO 60
  500 RETURN
C
 1001 FORMAT(I5,3F10.0)
 1002 FORMAT(I5,6F10.0)
 1005 FORMAT(I5,6F10.0/F15.0,5F10.0)
 1006 FORMAT (4F10.0)
 1007 FORMAT (I5,4F10.0/8F10.0)
 2001 FORMAT (24H1MATERIAL PROPERTIES.... ///
     .          54H MATERIAL    YOUNG S      POISSON S               MASS       /
     .          54H  NUMBER     MODULUS        RATIO               DENSITY      /)
 2002 FORMAT(1H ,I5,3X,F12.0,F14.5,F14.5)
 2003 FORMAT (/////20H BEAM GEOMETRIC PROPERTIES....  //
     1   48H ELEMENT       AREA          AREA          AREA
     2  60H INERTIA     INERTIA      INERTIA
     3 / 48H  TYPE              X             Y             Z
     4                  30H     X             Y             Z              )
 2004 FORMAT(1H ,I5,2X,6F12.3)
 2005 FORMAT(30H1.......THREE DIMENSIONAL BEAM ELEMENTS///
     .  36H  NUMBER OF BEAMS                   =,I5//
     .  36H  NUMBER OF GEOMETRIC PROPERTY SETS=,I5//
     .  36H  NUMBER OF FIXED END FORCE SETS    =,I5//
     .  36H  NUMBER OF MATERIALS               =,I5//
     .  36H  NUMBER OF BEAM N/L PROPERTY SETS =,I5)
 2006 FORMAT(///30H ELEMENT LOAD MULTIPLIERS...../12X,1HA,14X,1HB,14X,
     .  1HC,14X,1HD/6H X-DIR 4E15.6/6H Y-DIR 4E15.6/6H Z-DIR 4E15.6/ )
 2010 FORMAT (1H1
     1  30X40H FIXED END FORCES IN LOCAL COORDINATES
     2//53H TYPE   NODE          FORCE X      FORCE Y      FORCE Z
     3                       35H MOMENT X     MOMENT Y     MOMENT Z            )
 2011 FORMAT(1H ,I3,6X,1HI,3X,6F12.3/1H ,9X,1HJ,3X,6F12.3/)
 2013 FORMAT(1H0/
     1  60H SECTION PROPERTIES OTHER THAN SHEAR AREAS MAY NOT BE SPECIF
     2  34HIED AS ZERO. EXECUTION TERMINATED.)
 2020 FORMAT (1H1/30X 33HBEAM ELEMENT NONLINEAR PARAMETERS //
     1  8H   NPAR   4X 8HAXIAL     4X 8HMOMENT   4X 8HMOMENT   4X
     2  8H AXIAL    20X 24HYIELD FUNCTION CONSTANTS /
     3  8H    NC    6X 2HPO 9X 4HMU/Y 8X 4HMU/Z 9X 2HTU 8X 2HA0 6X 2HA1
     4  6X 2HA2 6X 2HA3 6X 2HB0 6X 2HB1 6X 2HB2 6X 2HB3  /)
 2021 FORMAT (I5,2X,4E12.3,8F8.3)
 3000 FORMAT (10I5,2I6,I4,I5)
 4000 FORMAT (23H1BEAM ELEMENT DATA.....  //
     . 5H BEAM 5X 5HNODES 5X 5H MATL 5H GEOM 5X 10HELEM LOADS 4X    10X
     . 12H END CODES   5X 5H L/NL /
     .                         5H  NO      4X 1HI 4X 1HJ 4X 1HK 5H  NO  5H  NO
     .  4X 1HA 4X 1HB 4X 1HC 4X 1HD 9X 1HI 9X 1HJ 5X 5H  IND )
 4001 FORMAT (10I5,2I10)
 4004 FORMAT(1H ,31HNODAL POINT NUMBERS FOR ELEMENT,I5,35HARE IDENTICAL.
     1 EXECUTION TERMINATED.)
C
      END
```

```
      SUBROUTINE NEWBM(E,G,RO,COPROP,SFT,NUMFIX,NUMETP)
C
      DIMENSION E(1),G(1),RO(1),COPROP(NUMETP,1),SFT(NUMFIX,1)
      COMMON/EM/LM(24),ND,NS,ASA(24,24),RF(24,4),XM(24),SA(12,24),
     .     SF(12,4),ENPAR(24),S(12,12),F(48),LMS(12),EC(12),T(3,3)
      COMMON/JUNK/ LC(4),JK(6),MELTYP,OL,MATTYP,ILC(4),TS(2,2),LS(4),
     .               EMUL(3,4),R(12),C(12),EL(3,4)
C
C     FORM NEW BEAM STIFFNESS
C
      DO 5 I=1,144
    5 S(I)=0.
      AX=COPROP(MELTYP,1)
      AY=COPROP(MELTYP,2)
      AZ=COPROP(MELTYP,3)
      AAX=COPROP(MELTYP,4)
      AAY=COPROP(MELTYP,5)
      AAZ=COPROP(MELTYP,6)
      SHFY=0.0
```

```
      SHFZ=0.0
      ZY=E(MATTYP)/(DL*DL)
      EIY=ZY*AAY
      EIZ=ZY*AAZ
      IF(AY.NE.0.0) SHFY=6.*EIZ/(G(MATTYP)*AY)
      IF(AZ.NE.0.0) SHFZ=6.*EIY/(G(MATTYP)*AZ)
      COMMY=EIY/(1.+2.*SHFZ)
      COMMZ=EIZ/(1.+2.*SHFY)
C
C     FIXED END FORCES IN LOCAL COORDS
C
      DO 73 N=1,4
      M=LC(N)
      IF (M.GT.0)  GO TO 71
      DO 70 I=1,12
   70 SF(I,N)=0.
      GO TO 73
   71 DO 72 I=1,12
   72 SF(I,N)=SFT(M,I)
   73 CONTINUE
C
C     ELEMENT LOCAL FORCES DUE TO GRAVITIES IN 3 DIRECTIONS
C
      DO 75 I=1,12
   75 EL(I)=0.
      PCM=RC(MATTYP)*AX
      DO 80 J=1,4
      DO 80 I=1,3
      XX=0.0
      DO 77 K=1,3
   77 XX=XX+PCM*T(I,K)*EMUL(K,J)
   80 EL(I,J)=XX
C
      DO 100 J=1,4
      K=1
      DO 100 I=1,12,6
      K=-1*K
      SF(I  ,J)=SF(I  ,J)-EL(1,J)*DL/2.0
      SF(I+1,J)=SF(I+1,J)-EL(2,J)*DL/2.0
      SF(I+2,J)=SF(I+2,J)-EL(3,J)*DL/2.0
      SF(I+4,J)=SF(I+4,J)-K*EL(3,J)*DL*DL/12.0
      SF(I+5,J)=SF(I+5,J)+K*EL(3,J)*DL*DL/12.0
  100 CONTINUE
C
C     FORM ELEMENT STIFFNESS IN LOCAL COORDINATES
C
      S(1,1)= E(MATTYP)* AX/DL
      S(4,4)= G(MATTYP)*AAX/DL
      S(2,2)= COMMZ*12./DL
      S(3,3)= COMMY*12./DL
      S(5,5)= COMMY* 4.*DL*(1.+0.5*SHFZ)
      S(6,6)= COMMZ* 4.*DL*(1.+0.5*SHFY)
      S(2,6)= COMMZ* 6.
      S(3,5)=-COMMY* 6.
      DO 102 I=1,6
      J=I+6
  102 S(J,J)=S(I,I)
      DO 104 I=1,4
      J=I+6
  104 S(I,J)=-S(I,I)
      S(6,12)= S(6,6)*(1.-SHFY)/(2.+SHFY)
      S(5,11)= S(5,5)*(1.-SHFZ)/(2.+SHFZ)
      S(2,12)= S(2,6)
      S(6, 8)=-S(2,6)
      S(8,12)=-S(2,6)
      S(3,11)= S(3,5)
      S(5, 9)=-S(3,5)
      S(9,11)= S(3,5)
      DO 106 I=2,12
      K=I-1
      DO 106 J=1,K
  106 S(I,J)=S(J,I)
C
C     MODIFY STIFFNESS AND FIXED END FORCES FOR ZERO END CONDITIONS
C
      IF ((JK(1)+JK(2)).EQ.0) GO TO 145
      DO 140 K=1,2
      KK=JK(K)
      KD=100000
      I1=6*(K-1)+1
      I2=I1+5
      DO 140 I=I1,I2
      IF (KK.LT.KD)  GO TO 140
      SII=S(I,I)
      DO 125 N=1,12
  125 R(N)=S(I,N)
      DO 130 M=1,12
      C(M)=S(M,I)/SII
      DO 130 N=1,12
  130 S(M,N)=S(M,N)-C(M)*R(N)
      DO 135 N=1,4
      SFI=SF(I,N)
      DO 135 M=1,12
  135 SF(M,N)=SF(M,N)-C(M)*SFI
  136 KK=KK-KD
  140 KD=KD/10
  145 CONTINUE
C
C     PERFORM LOCAL TO GLOBAL TRANSFORMATION
C
      DO 31 I=1,288
   31 SA(I)=0.
      DO 150 LA=1,10,3
      LB=LA+2
      DO 150 MA=1,10,3
      MB=MA-1
      DO 150 I=LA,LB
      DO 150 JM=1,3
      J=JM+MB
      XX=0.
      DO 151 K=1,3
  151 XX=XX+S(I,K+MB)*T(K,JM)
  150 SA(I,J)=XX
C
C     ELEM STIFF ASA(12,12) AND FIXED END FORCES PF(12) IN GLOBAL COORDS
C
      DO 32 I=1,576
   32 ASA(I)=0.
      DO 160 LA=1,10,3
      LB=LA-1
      DO 160 MA=1,10,3
      MB=MA+2
      DO 160 IL=1,3
      I=IL+LB
```

```
      DO 160 J=MA,MB
      XX=0.
      DO 161 K=1,3
  161 XX=XX+T(K,IL)*SA(K+LB,J)
  160 ASA(I,J)=XX
C
      DO 165 LA=1,10,3
      LB=LA-1
      DO 165 IL=1,3
      I=IL+LB
      DO 165 N=1,4
      XX=0.
      DO 162 K=1,3
  162 XX=XX-T(K,IL)*SF(K+LB,N)
  165 RF(I,N)=XX
C
C     FORM MASS MATRIX
C
      XXM =ROM*CL/2.0
      XXMX=XXM*(AAX/AX)
      XXMY=XXM*(DL**2/210.+AAY/(AX*5.0))
      XXMZ=XXM*(DL**2/210.+AAZ/(AX*5.0))
      DO 180 M=1,3
      RXM=XXMX*T(1,M)**2+XXMY*T(2,M)**2+XXMZ*T(3,M)**2
      XM(M   )=RXM
      XM(M+3)=RXM
      XM(M+6)=RXM
  180 XM(M+9)=RXM
      RETURN
      END
```

```
      SUBROUTINE SLAVE (X,Y,Z,ID,NUMNP,NI,NJ)
C
      DIMENSION X(1),Y(1),Z(1),ID(NUMNP,1)
      COMMON/EM/ LM(24),ND,NS,S(24,24),P(24,4),XM(24),SA(12,24),TT(12,4)
     +              ,ENPAR(24),SS(12,12),F(48),LMS(12),EC(12),T(3,3)
C
C     PERFORMS SLAVE...MASTER DISPLACEMENT TRANSFORMATION
C
      DO 10 I=1,12
      LMS(I)=0
   10 EC(I)=0.
      KK=0
      DO 100 NF=1,12,6
      NOD=NI
      IF (NF.EQ.7) NOD=NJ
      DO 80 K=1,3
      I=K+NF-1
      IF (LM(I).GE.0) GO TO 80
      LMS(I)=1
      M=-LM(I)
      LM(I)=ID(M,K)
      IF (K-2) 35,40,45
   35 D1=-(Y(NOD)-Y(M))
      D2=   Z(NOD)-Z(M)
      LM(ND+1)=ID(M,6)
      LM(ND+2)=ID(M,5)
      GO TO 50
   40 D1=-(Z(NOD)-Z(M))
      D2=   X(NOD)-X(M)
      LM(ND+1)=ID(M,4)
      LM(ND+2)=ID(M,6)
      GO TO 50
   45 D1=-(X(NOD)-X(M))
      D2=   Y(NOD)-Y(M)
      LM(ND+1)=ID(M,5)
      LM(ND+2)=ID(M,4)
   50 KK=KK+1
      EC(KK)=D1
      KK=KK+1
      EC(KK)=D2
C
C     TRANSFORMATION...ARRAYS INCREASE IN SIZE
C
      DO 60 II=1,NC
      S(ND+1,II)=S(I,II)*D1
      S(ND+2,II)=S(I,II)*D2
      XM(ND+1)=XM(I)*D1*D1
      XM(ND+2)=XM(I)*D2*D2
      S(II,ND+1)=S(II,I)*D1
      S(II,ND+2)=S(II,I)*D2
      DO 55 J=1,4
      P(ND+1,J)=P(I,J)*D1
      P(ND+2,J)=P(I,J)*D2
   55 CONTINUE
   60 CONTINUE
      DO 70 II=1,NS
      SA(II,ND+1)=SA(II,I)*D1
   70 SA(II,ND+2)=SA(II,I)*D2
C
      S(ND+1,ND+1)=S(I,I)*D1**2
      S(ND+2,ND+2)=S(I,I)*D2**2
```

```
      S(ND+1,ND+2)=S(1,1)*D1*D2
      S(ND+2,ND+1)=S(ND+1,ND+2)
      ND=ND+2
   80 CONTINUE
C
C     SET RCTATIONS
C
      DO 90 J=1,3
      K=NF+J+2
      IF(LM(K).GE.C)  GO TO 90
      M=-LM(K)
      LM(K)=ID(M,J+3)
   90 CONTINUE
C
  100 CONTINUE
C
      RETURN
      END
```

```
      SUBROLTINE CBEAM
C
      COMMCN A(1)
      COMMON/ELPAR/ NPAR(14),NUMNP,NELTYP,NUMEL,NUMNEL,NEQ,NEQG,MBAND,
     .              NEQB,NBLOCK,MTOT,N1,N2,N3,N4,N5,N6
      COMMON/JUNK/ STR(4),MM,L,K,NTAG,NDYN,SIG(12),EXPA(184)
C
      IF(NPAR(1).EC.0) GO TO 500
C
      N7=N6+NPAR(5)
      N8=N7+NPAR(5)
      N9=N8+NPAR(5)
      N10=N9+NPAR(4)*12
      N11=N10+NPAR(3)*6
      N12=N11+NPAR(6)
      N13=N12+NPAR(7)*12
      IF (N13.GT.MTCT) CALL ERROR (1,N13-MTOT,7HCBEAM  )
      CALL CTEAM (NPAR(2),NPAR(3),NPAR(4),NPAR(5),NPAR(6),NPAR(7),
     .            A(N1),A(N2),A(N3),A(N4),A(N5),A(N6),A(N7),A(N8),
     .            A(N9),A(N10),A(N11),A(N12),NUMNP,MBANC,NUMEL)
      RETURN
C
  500 IF(NTAG.EC.0) WRITE (6,2002)
      WRITE(6,3CO2) MM,L,(SIG(I),I=1,12)
      NTAG=1
      RETURN
C
 2002 FORMAT(/36HO ....CURVED BEAM FORCES AND MOMENTS//
     . 10HOBEAM LOAC 5X 5HAXIAL  2(7X,5HSHEAR),5X 7HTORSION
     .  2(5X,7HBENDING)/ 10H  NO.  NO.  8X  2HR1  10X  2HR2  10X
     .  2HR3 10X 2HM1 10X 2HM2 10X 2HM3)
 3002 FORMAT (I5,I4,1PE11.3,5E12.3/8X,6E12.3/)
C
      END
```

```
      SUBROLTINE CTEAM (NCBEAM,NUMETP,NUMFIX,NUMMAT,NUMRAD,NUMNPAR,
     .                  ID,X,Y,Z,INC,E,G,RO,SFT,COPRCP,RAD,ENPROP,
     .                  NUMNP,MBAND,NUMEL)
C
C     FORM 3-D CURVED BEAM STIFFNESS AND STRESS-DISP. ARRAYS
C
      DIMENSION ID(NUMNP,1),X(1),Y(1),Z(1),INC(1),E(1),G(1),RO(1),
     .          SFT(NUMFIX,1),COPRCP(NUMETP,1),RAD(1),ENPROP(NUMNPAR,1),
     .          STIF(722),LMS(12),EC(12),TI(3,3),TJ(3,3)
      COMMON/EM/LM(24),ND,NS,ASA(24,24),RF(24,4),XM(24),SA(12,24),
     .          SF(12,4),LNPAR(24),S(12,12),F(48),TT(72)
      CCMMON/JUNK/ LC(4),JK(6),MELTYP,DL,MATTYP,BETA,B,CP,D,EP,ARFF,
     .             EMUL(3,4),ILC(4),LS(4)
      EQUIVALENCE (STIF,LM),(LMS,TTT(1)),(EC,TTT(13)),(TI,TTT(25)),
     .            (TJ,TTT(34))
C
      WRITE (6,2005) NCBEAM,NUMETP,NUMFIX,NUMMAT,NUMRAD,NUMNPAR
      DO 5 I=1,1346
    5 STIF(I)=0.
C
C     READ AND PRINT RADIUS OF CURVATURE DATA
C
      WRITE (6,2000)
```

```
CBEA  58          DO 8 I=1,NUMRAD
CBEA  59          READ (5,1000) N,RAD(N)
CBEA  60          IF (RAD(N).NE.0.0) GO TO 7
CBEA  61          WRITE (6,2014)
CBEA  62          CALL EXIT
CBEA  63        7 WRITE (6,2015) N,RAD(N)
CBEA  64        8 CONTINUE
CBEA  65   C
CBEA  66   C      READ AND PRINT MATERIAL PROPERTY DATA
CBEA  67   C
CBEA  68          WRITE (6,2001)
CBEA  69          DO 10 I=1,NUMMAT
CBEA  70          READ  (5,1001) N,E(N),G(N),RO(N)
CBEA  71          WRITE (6,2002) N,E(N),G(N),RO(N)
CBEA  72       10 G(N)=0.5*E(N)/(1.+G(N))
CBEA  73   C
CBEA  74   C      READ AND PRINT CROSS SECTIONAL PROPERTIES
CBEA  75   C
CBEA  76          WRITE (6,2003)
CBEA  77          DO 30 I=1,NUMETP
CBEA  78          READ  (5,1002) N,(COPROP(N,J),J=1,6)
CBEA  79          IF((COPROP(N,1).NE.0.0).AND.(COPROP(N,4).NE.0.0).AND.
CBEA  80        1    (COPROP(N,5).NE.0.0).AND.(COPROP(N,6).NE.0.0))  GO TO 20
CBEA  81          WRITE (6,2013)
CBEA  82          CALL EXIT
CBEA  83       20 WRITE (6,2004) N,(COPROP(N,J),J=1,6)
CBEA  84       30 CONTINUE
CBEA  85   C
CBEA  86   C      ELEMENT LOAD MULTIPLIERS
CBEA  87   C
CBEA  88          READ  (5,1006) ((EMUL(I,J),J=1,4),I=1,3)
CBEA  89          WRITE (6,2006) ((EMUL(I,J),J=1,4),I=1,3)
CBEA  90   C
CBEA  91   C      READ AND PRINT FIXED END FORCES IN LOCAL COORDINATES
CBEA  92   C
CBEA  93          IF(NUMFIX .EQ. 0) GO TO 56
CBEA  94          WRITE (6,2010)
CBEA  95          DO 55 I=1,NUMFIX
CBEA  96          READ  (5,1005) N,(SFT(N,J),J=1,12)
CBEA  97       55 WRITE (6,2011) N,(SFT(N,J),J=1,12)
CBEA  98       56 CONTINUE
CBEA  99   C
CBEA 100   C      READ AND PRINT ELEMENT NONLINEAR PARAMETERS
CBEA 101   C
CBEA 102          IF (NUMPAR.EQ.0) GO TO 59
CBEA 103          WRITE(6,2020)
CBEA 104          DO 58 I=1,NUMPAR
CBEA 105       58 READ (5,1007) N,(ENPROP(N,J),J=1,12)
CBEA 106          WRITE(6,2021) (N,(ENPROP(N,J),J=1,12),N=1,NUMPAR)
CBEA 107       59 CONTINUE
CBEA 108   C
CBEA 109   C      READ AND PRINT ELEMENT DATA FOR EACH ELEMENT
CBEA 110   C
CBEA 111          WRITE (6,4000)
CBEA 112          L=0
CBEA 113       60 READ (5,3000) INEL,INI,INJ,INK,IMAT,IMEL,ILC,INELKI,INELKJ,IRAD,IQ
CBEA 114          IF (IQ.GT.NUMPAR) CALL ERROR (3,INEL,7HCBEAM  )
CBEA 115          L=L+1
CBEA 116          ML=INEL-L
CBEA 117          IF (ML)  66,67,66
CBEA 118       66 CALL ERROR (4,INEL,7HCBEAM  )
CBEA 119       67 NEL=INEL
```

```
CBEA 120          NI=INI
CBEA 121          NJ=INJ
CBEA 122          NK=INK
CBEA 123          MATTYP=IMAT
CBEA 124          MELTYP=IMEL
CBEA 125          DO 68 I=1,4
CBEA 126       68 LC(I)=ILC(I)
CBEA 127          NEKODI=INELKI
CBEA 128          NEKODJ=INELKJ
CBEA 129          NR=IRAD
CBEA 130          NIND=IQ
CBEA 131          WRITE (6,4001) NEL,NI,NJ,NK,MATTYP,MELTYP,LC,NEKODI,NEKODJ,NR,NIND
CBEA 132   C
CBEA 133   C      FORM GLOBAL TO LOCAL COORDINATE TRANSFORMATION.
CBEA 134   C
CBEA 135       74 DX=X(NJ)-X(NI)
CBEA 136          DY=Y(NJ)- Y(NI)
CBEA 137          DZ=Z(NJ)-Z(NI)
CBEA 138          DL=SQRT(DX*DX+DY*DY+DZ*DZ)
CBEA 139          IF(DL) 75,75,76
CBEA 140       75 CALL ERROR (5,NEL,7HCBEAM  )
CBEA 141       76 RA=RAD(NR)
CBEA 142          ARGU=DL/(2.0*RA)
CBEA 143          BETA=2.0*ASIN(ARGU)
CBEA 144          B =SIN(BETA)
CBEA 145          DP=COS(BETA)
CBEA 146          D =1.0-DP
CBEA 147          BX=X(NI)-X(NK)
CBEA 148          BY=Y(NI)-Y(NK)
CBEA 149          BZ=Z(NI)-Z(NK)
CBEA 150          BB=BX*BX+BY*BY+BZ*BZ
CBEA 151          BB=SQRT(BB)
CBEA 152          IF (BB.GT.0.0) GO TO 77
CBEA 153          CALL ERROR (6,NEL,7HCBEAM  )
CBEA 154       77 CX=DY*BZ-DZ*BY
CBEA 155          CY=DZ*BX-DX*BZ
CBEA 156          CZ=DX*BY DY*BX
CBEA 157          CC=CX*CX+CY*CY+CZ*CZ
CBEA 158          CC=SQRT(CC)
CBEA 159          BX=BX/BB
CBEA 160          BY=BY/BB
CBEA 161          BZ=BZ/BB
CBEA 162          CX=CX/CC
CBEA 163          CY=CY/CC
CBEA 164          CZ=CZ/CC
CBEA 165          AX=BY*CZ-BZ*CY
CBEA 166          AY=BZ*CX-BX*CZ
CBEA 167          AZ=BX*CY BY*CX
CBEA 168   C
CBEA 169          TI(1,1)=AX
CBEA 170          TI(1,2)=AY
CBEA 171          TI(1,3)=AZ
CBEA 172          TI(2,1)=BX
CBEA 173          TI(2,2)=BY
CBEA 174          TI(2,3)=BZ
CBEA 175          TI(3,1)=CX
CBEA 176          TI(3,2)=CY
CBEA 177          TI(3,3)=CZ
CBEA 178   C
CBEA 179          TJ(1,1)=AX*DP-BX*B
CBEA 180          TJ(1,2)=AY*DP-BY*B
CBEA 181          TJ(1,3)=AZ*DP-BZ*B
```

```
      TJ(2,1)=AX*B+BX*OP
      TJ(2,2)=AY*B+BY*OP
      TJ(2,3)=AZ*B+BZ*OP
      TJ(3,1)=CX
      TJ(3,2)=CY
      TJ(3,3)=CZ
C
C     CHECK IF NEW STIFFNESS NEEDED
C
      IF (NEL.EQ.1) GO TO 80
      IF (MR.NE.NR)  GO TO 80
      IF  (ABS(DS-DL).GT.DL/100.)  GO TO 80
      IF ((MT.NE.MATTYP).OR.(ME.NE.MELTYP)) GO TO 80
      IF ((JK(1).NE.NEKODI).OR.(JK(2).NE.NEKODJ)) GO TO 80
      DO 79 I=1,4
      IF (LS(I).NE.LC(I)) GO TO 80
   79 CONTINUE
      GO TO 100
C
   80 DS=DL
      MT=MATTYP
      ME=MELTYP
      MR=NR
      DO 85 I=1,4
   85 LS(I)=LC(I)
      JK(1)=NEKODI
      JK(2)=NEKODJ
C
C     FORM NEW CURVED BEAM STIFFNESS
C
      CALL NEWCEM (E,G,RO,COPROP,SFT,RAD,NUMFIX,NUMETP)
C
  100 CALL TRANSFM (RO,COPROP,NUMETP)
C
C     FORM ELEMENT LOCATION MATRIX
C
      DO 170 M=1,6
      LM(M)=ID(NI,M)
      LM(M+12)=0
      LM(M+18)=0
  170 LM(M+6)=ID(NJ,M)
      NS=12
      ND=12
C
C     TRANSFORM TO MASTER DEGREES OF FREEDOM
C
      CALL SLAVE (X,Y,Z,IO,NUMNP,NI,NJ)
C
C     WRITE ELEMENT INFORMATION ON TAPE
C
      CALL WRITET (IPBAND,NDIF)
C
C     SET NONLINEAR ELEMENT INDICATOR AND STORE NONLINEAR PARAMETERS
C
      NN=NUPEL+NEL
      IF (NIND.LE.0) GO TO 200
      INO(NN)=3
      DO 190 I=1,12
  190 ENPAR(I)=ENPAC(NIND,I)
      MTYPE=3
      WRITE (3) MTYPE,LM,ND,NS,ASA,SA,S,ENPAR,TT
      GO TO 250
```

```
  200 INO(NN)=-3
C
C     CHECK FOR LAST ELEMENT
C
  250 IF (NCBEAM-NEL) 66,500,60
  500 RETURN
C
 1000 FORMAT(I5,F10.0)
 1001 FORMAT(I5,3F10.0)
 1002 FORMAT(I5,6F10.0)
 1005 FORMAT(I5,6F10.0/F15.0,5F10.0)
 1006 FORMAT (4F10.0)
 1007 FORMAT (I5,4F10.0/8F10.0)
 2000 FORMAT (26H1RADII OF CURVED BEAMS....  //
     .          27H RADIUS          RADIUS OF      /
     .          27H NUMBER          CURVATURE      /)
 2001 FORMAT (////24H MATERIAL PROPERTIES....  //
     .          54H MATERIAL   YOUNG S     POISSON S            MASS      /
     .          54H NUMBER     MODULUS       RATIO           DENSITY      /)
 2002 FORMAT(1H ,I5,3X,F12.0,F14.5,F14.5)
 2003 FORMAT (38H1CURVED BEAM GEOMETRIC PROPERTIES.....  //
     1   48H ELEMENT       AREA         AREA          AREA
     2  60H INERTIA      INERTIA      INERTIA
     3 / 48H  TYPE             X            Y             Z
     4                  30H    X            Y             Z              /)
 2004 FORMAT(1H ,I5,2X,6F12.3)
 2005 FORMAT(39H1.......THREE DIM. CURVED BEAM ELEMENTS///
     .  36H  NUMBER OF CURVED BEAMS             =,I5//
     .  36H  NUMBER OF SECTIONAL PROPERTY SETS=,I5//
     .  36H  NUMBER OF FIXED END FORCE SETS    =,I5//
     .  36H  NUMBER OF MATERIALS               =,I5//
     .  36H  NUMBER OF RADII                   =,I5//
     .  36H  NUMBER OF N/L PARAMETER SETS      =,I5)
 2006 FORMAT (///30H ELEMENT LOAD MULTIPLIERS...../12X,1HA,14X,1HB,14X,
     .  1HC,14X,1HD/6H X-DIR 4E15.6/6H Y-DIR 4E15.6/6H Z DIR 4E15.6/  )
 2010 FORMAT (1H1
     1  30X40H FIXED END FORCES IN LOCAL COORDINATES
     2//53H TYPE   NODE        FORCE X       FORCE Y      FORCE Z
     3                    35H MOMENT X     MOMENT Y     MOMENT Z         //)
 2011 FORMAT(1H ,I3,6X,1HI,3X,6F12.3/1H ,9X,1HJ,3X,6F12.3/)
 2013 FORMAT(1H0/
     1  60H SECTION PROPERTIES OTHER THAN SHEAR AREAS MAY NOT BE SPECIF
     2  34HIED AS ZERO. EXECUTION TERMINATED.)
 2014 FORMAT(1H0/
     1  60H RADIUS OF CURVATURE MAY NOT BE SPECIFIED
     2  34H     AS ZERO.. EXECUTION TERMINATED.)
 2015 FORMAT (I5,F20.5)
 2020 FORMAT (1H1/30X 40HCURVED BEAM ELEMENT NONLINEAR PARAMETERS  //
     .  8H NL PAR  4X 8HAXIAL    4X 8HMOMENT   4X 8HMOMENT   4X
     .  8HTORSION  20X 26HYIELD FUNCTION CONSTANTS /
     .  8H    NO    6X 2HPU 9X 4HMU/Y 8X 4HMU/Z 9X 2HTU 8X 2HA0 6X 2HA1
     .  6X 2HA2 6X 2HA3 6X 2HB0 6X 2HB1 6X 2HB2 6X 2HB3  /)
 2021 FORMAT (I5,3X,4E12.3,8F8.3)
 3000 FORMAT (10I5,2I6,I6,I5)
 4000 FORMAT (30H1CURVED BEAM ELEMENT DATA.....  //
     . 5H BEAM 5X 5HNODES 5X 5H MATL 5H GEOM 5X 10HELEM LOADS 4X   10X
     . 12H  END CODES    3X 6HRADIUS 3X 5H NLNO /
     .                        5H  NO      4X 1HI 4X 1HJ 4X 1HK 5H  NO  5H  NO
     .  4X 1HA 4X 1HB 4X 1HC 4X 1HD 9X 1HI 9X 1HJ    8X 2HNO 8H      NO /)
 4001 FORMAT (10I5,2I10,I10,I8)
 4004 FORMAT(1H ,31HNODAL POINT NUMBERS FOR ELEMENT,I5,36HARE IDENTICAL.
     1 EXECUTION TERMINATED.)
```

```
C
      END
```

```
      SUBROUTINE NEWCBM (E,G,RO,COPROP,SFT,RAD,NUMFIX,NUMETP)
C
      DIMENSION E(1),G(1),RO(1),COPROP(NUMETP,1),SFT(NUMFIX,1),RAD(1)
      COMMON/EM/LM(24),ND,NS,ASA(24,24),RF(24,4),XM(24),SA(12,24),
     *  SF(12,4),ENPAR(24),S(12,12),F(48),LMS(12),EC(12),TI(3,3),TJ(3,3)
      COMMON/JUNK/ LC(4),JK(6),MELTYP,DL,MATTYP,BETA,B,DP,D,RA,AX,
     *             EMUL(3,4),ILC(4),LS(4),R(12),C(12),EL(3,4),V(12,4)
C
C     FORM NEW CURVED BEAM STIFFNESS
C
      DO 5 I=1,144
    5 S(I)=0.
      AX=COPROP(MELTYP,1)
      AY=COPROP(MELTYP,2)
      AZ=COPROP(MELTYP,3)
      AAX=COPROP(MELTYP,4)
      AAY=COPROP(MELTYP,5)
      AAZ=COPROP(MELTYP,6)
C
      RA2=RA*RA
      RA3=RA2*RA
      EIY=E(MATTYP)*AAY
      EIZ=E(MATTYP)*AAZ
      PHIZ=EIZ/(AX*RA2)
      PHIY=EIY/(G(MATTYP)*AAX)
C
      IF (AY.LE.0.) GO TO 20
      ZETAZ=EIZ/(AY*G(MATTYP)*RA2)
      GO TO 21
   20 ZETAZ=0.
   21 IF (AZ.LE.0.) GO TO 30
      ZETAY=EIY/(AZ*G(MATTYP)*RA2)
      GO TO 31
   30 ZETAY=0.
   31 CONTINUE
      COMMY=RA3/EIY
      COMMZ=RA3/EIZ
C
C     FIXED END FORCES IN LOCAL COORDINATES
C
      DO 73 N=1,4
      M=LC(N)
      IF (M.GT.0)  GO TO 71
      DO 70 I=1,12
   70 SF(I,N)=0.
      GO TO 73
   71 DO 72 I=1,12
   72 SF(I,N)=SFT(M,I)
   73 CONTINUE
C
C     FORM ELEMENT STIFFNESS IN LOCAL COORDINATES
C
      AA=BETA*0.5 0.5*B*DP
      CC=BETA*0.5+0.5*B*DP
      EE=0.5*B*B
C
      F11=COMMZ*(BETA+CC-2.0*B+PHIZ*CC+ZETAZ*AA)
      F12=COMMZ*(D-EE-PHIZ*EE+ZETAZ*EE)
      F16=(COMMZ/RA)*(B-BETA)
      F22=COMMZ*(AA+PHIZ*AA+ZETAZ*CC)
      F26=(COMMZ/RA)*(-D)
C
      F33=COMMY*(AA+PHIY*(BETA+CC-2.0*B)+ZETAY*BETA)
      F34=(COMMY/RA)*(AA-PHIY*(B-CC))
      F35=(COMMY/RA)*(EE+PHIY*(D-EE))
      F44=(COMMY/RA2)*(AA+PHIY*CC)
      F45=(COMMY/RA2)*(EE-PHIY*EE)
      F55=(COMMY/RA2)*(CC+PHIY*AA)
      F66=(COMMZ/RA2)*BETA
C
      U=F11*F22*F66+2.0*F12*F16*F26-F22*F16*F16-F11*F26*F26-F66*F12*F12
      W=F33*F44*F55+2.0*F34*F45*F35-F44*F35*F35-F33*F45*F45-F55*F34*F34
C
      S(1,1)= (F22*F66-F26*F26)/U
      S(1,2)=-(F12*F66-F16*F26)/U
      S(1,6)= (F12*F26-F22*F16)/U
      S(2,2)= (F11*F66-F16*F16)/U
      S(2,6)=-(F11*F26-F12*F16)/U
      S(6,6)= (F11*F22-F12*F12)/U
C
      S(3,3)= (F44*F55-F45*F45)/W
      S(3,4)=-(F34*F55-F35*F45)/W
      S(3,5)= (F34*F45-F44*F35)/W
      S(4,4)= (F33*F55-F35*F35)/W
      S(4,5)=-(F33*F45-F35*F34)/W
      S(5,5)= (F33*F44-F34*F34)/W
C
      S(1, 7)= S(1,1)*DP+S(1,2)*B
      S(1, 8)= S(1,1)*B-S(1,2)*DP
      S(1,12)= S(1,1)*RA*D+S(1,2)*RA*B-S(1,6)
      S(2, 7)= S(1,2)*DP+S(2,2)*B
      S(2, 8)= S(1,2)*B-S(2,2)*DP
      S(2,12)= S(1,2)*RA*D+S(2,2)*RA*B-S(2,6)
      S(3, 9)= S(3,3)
      S(3,10)= S(3,3)*RA*D-S(3,4)*DP+S(3,5)*B
      S(3,11)=-S(3,3)*RA*B-S(3,4)*B-S(3,5)*DP
      S(4, 9)= S(3,4)
      S(4,10)= S(3,4)*RA*D-S(4,4)*DP+S(4,5)*B
      S(4,11)=-S(3,4)*RA*B-S(4,4)*B -S(4,5)*DP
      S(5, 9)= S(3,5)
      S(5,10)= S(3,5)*RA*D-S(4,5)*DP+S(5,5)*B
      S(5,11)=-S(3,5)*RA*B-S(4,5)*B-S(5,5)*DP
      S(6, 7)= S(1,6)*DP+S(2,6)*B
      S(6, 8)=-S(1,6)*B-S(2,6)*DP
      S(6,12)= S(1,6)*RA*D+S(2,6)*RA*B-S(6,6)
C
      S( 7, 7)= S(1,1)
      S( 7, 8)=-S(1,2)
      S( 7,12)= S(1,6)
      S( 8, 8)= S(2,2)
      S( 8,12)=-S(2,6)
      S( 9, 9)= S(3,3)
      S( 9,10)= S(3,4)
      S( 9,11)=-S(3,5)
      S(10,10)= S(4,4)
      S(10,11)=-S(4,5)
      S(11,11)= S(5,5)
      S(12,12)= S(6,6)
```

```
CBEA 426  C
CBEA 427        DO 106 I=2,12
CBEA 428        K=I-1
CBEA 429        DO 106 J=1,K
CBEA 430    106 S(I,J)=S(J,I)
CBEA 431  C
CBEA 432  C     ELEMENT LOCAL FORCES DUE TO GRAVITIES IN 3 DIRECTIONS
CBEA 433  C
CBEA 434        DO 107 I=1,60
CBEA 435    107 EL(I)=0.
CBEA 436        ROM=RO(MATTYP)*AX
CBEA 437        DO 110 J=1,4
CBEA 438        DO 110 I=1,3
CBEA 439        XX=0.
CBEA 440        DO 109 K=1,3
CBEA 441    109 XX=XX+ROM*TI(I,K)*EMUL(K,J)
CBEA 442    110 EL(I,J)=XX
CBEA 443  C
CBEA 444        V11=COMMZ*((2.0*D-0.75*B*B-BETA*B+0.5*BETA*B*DP+0.25*BETA*BETA)
CBEA 445       .            +PHIZ*(0.5*BETA*B*DP+0.25*BETA*BETA-0.25*B*B))
CBEA 446        V12=COMMZ*((3.0*B-BETA*DP-0.75*B*DP+0.5*BETA*DP*DP-0.75*BETA)
CBEA 447       .            -PHIZ*(0.5*BETA*B*R+0.25*R*DP))
CBEA 448        V21=COMMZ*((0.25*BETA-0.75*B*DP+0.5*BETA*DP*DP)
CBEA 449       .            -PHIZ*(0.5*BETA+0.25*R*DP-0.5*BETA*DP*DP))
CBEA 450        V22=COMMZ*((0.25*BETA*BETA-0.5*BETA*R*DP+0.5*B*B-0.25*DP*DP+DP
CBEA 451       .     -0.75)+PHIZ*(0.25*BETA*BETA-0.5*BETA*B*DP+0.25*B*B))
CBEA 452        V3 =COMMY*((1.0-DP-0.5*B*B)+PHIY*(0.5*BETA*BETA-BETA*B+0.5*B*B))
CBEA 453        V4 =(COMMY/RA)*((1.0-DP-0.5*B*B)-PHIY*(DP+BETA*B-0.5*B*B-1.0))
CBEA 454        V5 =(COMMY/RA)*((B-0.5*BETA-0.5*B*DP)
CBEA 455       .               +PHIY*(B-0.5*BETA-BETA*DP+0.5*B*DP))
CBEA 456        V61=(COMMZ/RA)*(BETA*B-2.0+2.0*DP)
CBEA 457        V62=(COMMZ/RA)*(BETA+BETA*DP-2.0*B)
CBEA 458  C
CBEA 459        R71=-RA*BETA*DP
CBEA 460        R72= RA*BETA*B
CBEA 461        R81=-R72
CBEA 462        R82= R71
CBEA 463        R9 =-RA*BETA
CBEA 464        R10= RA2*(BETA-B)
CBEA 465        R11=-RA2*C
CBEA 466        R121=RA2*(B-BETA*DP)
CBEA 467        R122=RA2*(BETA*B+DP-1.0)
CBEA 468  C
CBEA 469        DO 120 J=1,4
CBEA 470        V(1,J)=EL(1,J)*V11+EL(2,J)*V12
CBEA 471        V(2,J)=EL(1,J)*V21+EL(2,J)*V22
CBEA 472        V(3,J)=EL(3,J)*V3
CBEA 473        V(4,J)=EL(3,J)*V4
CBEA 474        V(5,J)=EL(3,J)*V5
CBEA 475        V(6,J)=EL(1,J)*V61+EL(2,J)*V62
CBEA 476  C
CBEA 477        SF( 7,J)=SF( 7,J)+EL(1,J)*R71+EL(2,J)*R72
CBEA 478        SF( 8,J)=SF( 8,J)+EL(1,J)*R81+EL(2,J)*R82
CBEA 479        SF( 9,J)=SF( 9,J)+EL(3,J)*R9
CBEA 480        SF(10,J)=SF(10,J)+EL(3,J)*R10
CBEA 481        SF(11,J)=SF(11,J)+EL(3,J)*R11
CBEA 482        SF(12,J)=SF(12,J)+EL(1,J)*R121+EL(2,J)*R122
CBEA 483    120 CONTINUE
CBEA 484  C
CBEA 485        DO 124 I=1,12
CBEA 486        DO 124 L=1,4
CBEA 487        XX=0.
CBEA 488        DO 123 K=1,6
CBEA 489    123 XX=XX-S(I,K)*V(K,L)
CBEA 490    124 SF(I,L)=SF(I,L)+XX
CBEA 491  C
CBEA 492  C     MODIFY STIFFNESS AND FIXED END FORCES FOR ZERO END CONDITIONS
CBEA 493  C
CBEA 494        IF ((JK(1)+JK(2)).EQ.0) GO TO 145
CBEA 495        DO 140 K=1,2
CBEA 496        KK=JK(K)
CBEA 497        KD=100000
CBEA 498        I1=6*(K-1)+1
CBEA 499        I2=I1+5
CBEA 500        DO 140 I=I1,I2
CBEA 501        IF (KK.LT.KD)  GO TO 140
CBEA 502        SII=S(I,I)
CBEA 503        DO 125 N=1,12
CBEA 504    125 R(N)=S(I,N)
CBEA 505        DO 130 M=1,12
CBEA 506        C(M)=S(M,I)/SII
CBEA 507        DO 130 N=1,12
CBEA 508    130 S(M,N)=S(M,N)-C(M)*R(N)
CBEA 509        DO 135 N=1,4
CBEA 510        SFI=SF(I,N)
CBEA 511        DO 135 M=1,12
CBEA 512    135 SF(M,N)=SF(M,N)-C(M)*SFI
CBEA 513    136 KK=KK-KD
CBEA 514    140 KD=KD/10
CBEA 515    145 CONTINUE
CBEA 516  C
CBEA 517        RETURN
CBEA 518        END


CBEA 519        SUBROUTINE TRANSFM (RO,COPROP,NUMETP)
CBEA 520  C
CBEA 521        DIMENSION RO(1),COPROP(NUMETP,1)
CBEA 522        COMMON/EM/LM(24),NJ,NS,ASA(24,24),PF(24,4),XM(24),SA(12,24),
CBEA 523       .  SF(12,4),FNPAR(24),S(12,12),F(48),LMS(12),EC(12),TI(3,3),TJ(3,3)
CBEA 524        COMMON/JUNK/ LC(4),JK(6),MELTYP,OL,MATTYP,BETA,B,DP,D,RA,AX,
CBEA 525       .              EMUL(3,4),ILC(4),LS(4),R(12),C(12),EL(3,4)
CBEA 526  C
CBEA 527  C     PERFORM LOCAL TO GLOBAL TRANSFORMATION
CBEA 528  C
CBEA 529        DO 31 I=1,288
CBEA 530     31 SA(I)=0.
CBEA 531        DO 150 LA=1,10,3
CBEA 532        LB=LA+2
CBEA 533        DO 150 MA=1,10,3
CBEA 534        MB=MA-1
CBEA 535        DO 150 I=LA,LB
CBEA 536        DO 150 JM=1,3
CBEA 537        J=JM+MB
CBEA 538        XX=0.
CBEA 539        IF (J.GT.6) GO TO 152
CBEA 540        DO 151 K=1,3
CBEA 541    151 XX=XX+S(I,K+MB)*TI(K,JM)
CBEA 542        GO TO 150
CBEA 543    152 DO 153 K=1,3
CBEA 544    153 XX=XX+S(I,K+MB)*TJ(K,JM)
CBEA 545    150 SA(I,J)=XX
```

```
CBEA 546  C
CBEA 547        DO 32 I=1,576
CBEA 548     32 ASA(I)=0.
CBEA 549  C
CBEA 550        DO 160 LA=1,10,3
CBEA 551        LB=LA-1
CBEA 552        DO 160 MA=1,10,3
CBEA 553        MB=MA+2
CBEA 554        DO 160 IL=1,3
CBEA 555        I=IL+LB
CBEA 556        DO 160 J=MA,MB
CBEA 557        XX=0.
CBEA 558        IF (I.GT.6) GO TO 162
CBEA 559        DO 161 K=1,3
CBEA 560    161 XX=XX+TI(K,IL)*SA(K+LB,J)
CBEA 561        GO TO 160
CBEA 562    162 DO 163 K=1,3
CBEA 563    163 XX=XX+TJ(K,IL)*SA(K+LB,J)
CBEA 564    160 ASA(I,J)=XX
CBEA 565  C
CBEA 566        DO 170 LA=1,10,3
CBEA 567        LB=LA-1
CBEA 568        DO 170 IL=1,3
CBEA 569        I=IL+LB
CBEA 570        DO 170 N=1,4
CBEA 571        XX=0.
CBEA 572        IF (I.GT.6) GO TO 172
CBEA 573        DO 171 K=1,3
CBEA 574    171 XX=XX-TI(K,IL)*SF(K+LB,N)
CBEA 575        GO TO 170
CBEA 576    172 DO 173 K=1,3
CBEA 577    173 XX=XX-TJ(K,IL)*SF(K+LB,N)
CBEA 578    170 FF(I,N)=XX
CBEA 579  C
CBEA 580  C     FORM MASS MATRIX
CBEA 581  C
CBEA 582        XL   =RA*BETA
CBEA 583        XXM  =RO(MATTYP)*AX*XL/2.0
CBEA 584        XXMX=XXM*(COPROP(MATTYP,4)/AX)
CBEA 585        XXMY=XXM*(XL**2/210.+COPROP(MATTYP,5)/(5.0*AX))
CBEA 586        XXMZ=XXM*(XL**2/210.+COPROP(MATTYP,6)/(5.0*AX))
CBEA 587        DO 180 M=1,3
CBEA 588        XM(M   )=XXM
CBEA 589        XM(M+6)=XXM
CBEA 590        XM(M+3)=XXMX*TI(1,M)**2+XXMY*TI(2,M)**2+XXMZ*TI(3,M)**2
CBEA 591    180 XM(M+9)=XXMX*TJ(1,M)**2+XXMY*TJ(2,M)**2+XXMZ*TJ(3,M)**2
CBEA 592        RETURN
CBEA 593        END
```

```
BOUN   1        SUBROUTINE BOUND
BOUN   2  C
BOUN   3        COMMON A(1)
BOUN   4        COMMON/ELPAR/ NPAR(14),NUMNP,NELTYP,NUMEL,NUMNEL,NEG,NEQ,MBAND,
BOUN   5       .              NEQB,NBLOCK,MTOT,N1,N2,N3,N4,N5,N6
BOUN   6        COMMON/JUNK/ STR(4),MM,L,K,NTAG,NDYN,SIG(12),EXPA(184)
BOUN   7  C
BOUN   8        IF (NPAR(1).EQ.0) GO TO 500
BOUN   9        N7=N6+NPAR(3)*6
BOUN  10        N8=N7+NPAR(4)*12
BOUN  11        IF (N8.GT.MTOT) CALL ERROR (1,N8-MTOT,7HBOUND   )
BOUN  12        CALL BIND (NPAR(2),NPAR(3),NPAR(4),A(N1),A(N2),A(N3),A(N4),A(N5),
BOUN  13       .           A(N6),A(N7),NUMNP,MBAND,NUMEL)
BOUN  14        RETURN
BOUN  15  C
BOUN  16    500 IF (NTAG.EQ.0) WRITE (6,2002)
BOUN  17        WRITE(6,3002) MM,L,(SIG(I),I=1,6 )
BOUN  18        NTAG=1
BOUN  19        RETURN
BOUN  20  C
BOUN  21   2002 FORMAT (/23HC ....BOUNDARY FORCES AND MOMENTS  //
BOUN  22       .  10H0 BD. LOAD 5X 5HAXIAL 2(7X 5HSHEAR) 5X 7HTORSION
BOUN  23       .  2(5X 7HBENDING) /10H  NO   NO  8X 2HR1 10X 2HR2 10X 2HR3 10X
BOUN  24       .  2HM1 10X 2HM2 10X 2HM3 )
BOUN  25   3002 FORMAT (I5,I4,1PE11.3,5E12.3)
BOUN  26  C
BOUN  27        END


BOUN  28        SUBROUTINE BIND (NBOUND,NSTIF,NUMNPAR,ID,X,Y,Z,INC,STIF,ENPROP,
BOUN  29       .                 NUMNP,MBAND,NUMEL)
BOUN  30  C
BOUN  31        DIMENSION ID(NUMNP,1),X(1),Y(1),Z(1),IND(1),STIF(NSTIF,1),
BOUN  32       .          ENPROP(NUMNPAR,1),ENPAR(24),T(3,3)
BOUN  33        COMMON/EM/LM(24),ND,NS,ASA(24,24),P(24,4),XM(24),SA(12,24),
BOUN  34       .          SF(12,4),SY(6),EE(6),UE(6),PAR(6),S(12,12),F(48),TTT(72)
BOUN  35        COMMON/JUNK/ DX,DY,DZ,DD,AX,AY,AZ,AA,BX,BY,BZ,BB,SPRING(6),NP
BOUN  36        EQUIVALENCE (ENPAR,SY),(T,TTT)
BOUN  37  C
BOUN  38  C     FORM ELEMENT STIFFNESS OF BOUNDARY SPRING ELEMENTS
BOUN  39  C
BOUN  40        DO 5 I=1,1346
BOUN  41      5 LM(I)=0.
BOUN  42        ND=6
BOUN  43        NS=6
BOUN  44        NE=0
BOUN  45        WRITE(6,2000) NBOUND,NSTIF,NUMNPAR
BOUN  46  C
BOUN  47  C     READ AND PRINT SPRING STIFNESS SETS
BOUN  48  C
BOUN  49        READ (5,1010) (N,(STIF(N,J),J=1,6),N=1,NSTIF)
BOUN  50        WRITE(6,2001)
BOUN  51        WRITE(6,2010) (N,(STIF(N,J),J=1,6),N=1,NSTIF)
BOUN  52  C
BOUN  53  C     READ AND PRINT NONLINEAR ELEMENT PARAMETERS
BOUN  54  C
BOUN  55        IF (NUMNPAR.EQ.0) GO TO 9
BOUN  56        READ (5,1020) (N,(ENPROP(N,J),J=1,12),N=1,NUMNPAR)
BOUN  57        WRITE(6,2002)
```

```
      WRITE(6,2010) (N,(ENPROP(N,J),J=1,6),N=1,NUMNPAR)
      WRITE(6,2003)
      WRITE(6,2010) (N,(ENPROP(N,J),J=7,12),N=1,NUMNPAR)
    9 CONTINUE
C
C     READ AND PRINT ELEMENT DATA---DATA NEED TO BE SUPPLIED FOR EACH
C
      WRITE(6,2004)
   10 NE=NE+1
      READ (5,1030) INE,NI,NJ,NK,NS,NIND
      IF (INE.NE.NE) CALL ERROR (4,INE,7HBOUND  )
      IF (NIND.GT.NUMNPAR) CALL ERROR (3,INE,7HBOUND  )
      WRITE(6,2040) NE,NI,NJ,NK,NS,NIND
C
C     FORM LOCAL TO GLOBAL TRANSFORMATION
C
      DX=X(NJ)-X(NI)
      DY=Y(NJ)-Y(NI)
      DZ=Z(NJ)-Z(NI)
      DD=SQRT(DX*DX+DY*DY+DZ*DZ)
      IF (DD.LE.0.) CALL ERROR (5,INE,7HBOUND  )
      AX=X(NK)-X(NI)
      AY=Y(NK)-Y(NI)
      AZ=Z(NK)-Z(NI)
      BX=DY*AZ-DZ*AY
      BY=DZ*AX-DX*AZ
      BZ=DX*AY-DY*AX
      BB=SQRT(BX*BX+BY*BY+BZ*BZ)
      IF (BB.LE.0.0) CALL ERROR (6,NE,7HBOUND  )
C
      T(3,1)=BX/BB
      T(3,2)=BY/BB
      T(3,3)=BZ/BB
      T(1,1)=DX/DD
      T(1,2)=DY/DD
      T(1,3)=DZ/DD
      T(2,1)=T(3,2)*T(1,3)-T(3,3)*T(1,2)
      T(2,2)=T(3,3)*T(1,1)-T(3,1)*T(1,3)
      T(2,3)=T(3,1)*T(1,2)-T(3,2)*T(1,1)
C
C     FORM BOUNDARY STIFFNESSES IN LOCAL COORDINATE SYSTEM
C
      DO 100 I=1,144
  100 S(I)=0.
      DO 110 I=1,6
      SPRING(I)=STIF(NS,I)
      S(I,I)=SPRING(I)
  110 CONTINUE
C
C     FORM LOCAL TO GLOBAL STRESS ARRAY
C
      DO 120 I=1,288
  120 SA(I)=0.
      DO 150 LA=1,6,3
      LB=LA+2
      DO 150 MA=1,6,3
      MB=MA-1
      DO 150 I=LA,LB
      DO 150 JM=1,3
      J=JM+MB
      XX=0.
      DO 151 K=1,3
      YY=S(I,K+MB)
      IF (YY.EQ.0.) GO TO 151
      XX=XX+YY*T(K,JM)
  151 CONTINUE
  150 SA(I,J)=XX
C
C     FORM ELEMENT STIFFNESSES IN GLOBAL COORDINATE SYSTEM
C
      DO 155 I=1,576
  155 ASA(I)=0.
      DO 160 LA=1,6,3
      LB=LA-1
      DO 160 MA=1,6,3
      MB=MA+2
      DO 160 IL=1,3
      I=IL+LB
      DO 160 J=MA,MB
      XX=0.
      DO 161 K=1,3
      YY=SA(K+LB,J)
      IF (YY.EQ.0.) GO TO 161
      XX=XX+T(K,IL)*YY
  161 CONTINUE
  160 ASA(I,J)=XX
C
C     FORM ELEMENT LOCATION MATRIX
C
      DO 170 I=1,6
      LM(I)=ID(NI,I)
      LM(I+6)=0
      LM(I+12)=0
  170 LM(I+18)=0
C
C     WRITE ELEMENT INFORMATION ON TAPE
C
      CALL WRITET (MBAND,NDIF)
C
C     SET NONLINEAR ELEMENT INDICATORS AND STORE NONLINEAR PARAMETERS
C
      NN=NUMEL+NE
      IF (NIND.EQ.0) GO TO 200
      IND(NN)=4
      DO 180 I=1,6
      II=I+6
      SY(I)=ENPROP(NIND,I)
      EP(I)=ENPROP(NIND,II)
      UE(I)=SY(I)/SPRING(I)
  180 CONTINUE
      MTYPE=4
      WRITE (3) MTYPE,LM,ND,NS,ASA,SA,S,ENPAR,TTT
      GO TO 250
  200 IND(NN)=-4
C
  250 IF (NE.LT.NBOUND) GO TO 10
      RETURN
C
 1010 FORMAT (I5,6F10.0)
 1020 FORMAT (I5,6F10.0/5X,6F10.0)
 1030 FORMAT (6I5)
C
 2000 FORMAT (24H1......BOUNDARY ELEMENTS ///
     ,  28H NUMBER OF ELEMENTS          =,I5//
```

```
     .  28H NUMBER OF STIFFNESS SETS  =,I5//
     ,  28H NUMBER OF N/L PAR. SETS   =,I5)
 2001 FORMAT (4H1SET 20X 31HSTIFFNESSES OF BOUNDARY SPRINGS /
     .  4H NC  4X 2HR1 10X 2HR2 10X 2HR3 10X 2HM1 10X 2HM2 10X 2HM3 /)
 2002 FORMAT (4H1SET 20X 35HYIELDING FORCES OF BOUNDARY SPRINGS /
     .  4H NC  4X 2HR1 10X 2HR2 10X 2HR3 10X 2HM1 10X 2HM2 10X 2HM3 /)
 2003 FORMAT (4H1SET 20X 39HPLASTIC STIFFNESSES OF BOUNDARY SPRINGS /
     .  4H NC  4X 2HR1 10X 2HR2 10X 2HR3 10X 2HM1 10X 2HM2 10X 2HM3 /)
 2004 FORMAT (5H1ELE. 5X 5HNODES 5X 5H MATL 5X 5H NINO /
     ,  5H  NO  4X 1HI 4X 2HJ 4X 1HK 5H  NO  5X 5H  NO  /)
 2010 FORMAT (I4,6E12.3)
 2040 FORMAT (5I5,11C)
C
      END
```

```
      SUBROUTINE EXPJB
C
      COMMON/ELPAR/ NPAR(14),NUMNP,NELTYP,NUMEL,NUMNEL,NEQ,NEQG,MBAND,
     .                    NEQB,NBLOCK,MTOT,N1,N2,N3,N4,N5,N6
      COMMON/JUNK/ STR(4),MM,L,K,NTAG,NDYN,SIG(12),EXPA(184)
      COMMON A(1)
C
      IF (NPAR(1).EQ.0) GO TO 500
      N7=N6+NPAR(3)*5
      N8=N7+NPAR(3)
      N9=N8+NPAR(3)*10
      IF (N9.GT.MTOT) CALL ERROR (1,N7-MTOT,7HEXPJB  )
      CALL XPJB (NPAR(2),NPAR(3),A(N1),A(N2),A(N3),A(N4),A(N5),A(N6),
     .               A(N7),A(N8),NUMNP,MBAND,NUMEL)
      RETURN
C
  500 IF (NTAG.EQ.0) WRITE(6,2002)
      WRITE(6,3002) MM,L,(SIG(I),I=1,12)
      NTAG=1
      RETURN
C
 2002 FORMAT (/30HC ....JOINT FORCES AND MOMENTS  //
     .  10H0 JT. LOAD,5X,5HAXIAL,2(7X,5HSHEAR),7X,5HAXIAL,5X,7HBENDING,
     .  5X,7H  SHEAR/10H  NO.  NO. ,8X,2HF1,10X,2HF2,10X,2HF3,
     .  10X,2HR4,10X,2HR5,10X,2HR6)
 3002 FORMAT (I5,I4,1PE11.3,5E12.3/8X,6E12.3/)
C
      END
```

```
      SUBROUTINE XPJP (NEXPJ,NUMNPAR,ID,X,Y,Z,INO,ENPROP,NT,XT,NUMNP,
     .                      MBAND,NUMEL)
C
      DIMENSION ID(NUMNP,1),X(1),Y(1),Z(1),INO(1),ENPROP(NUMNPAR,1),
     .            NT(1),XT(NUMNPAR,1),ENPAR(24),AT(6,6),T(3,3)
      COMMON/EM/ LM(24),ND,NS,ASA(24,24),P(24,4),XM(24),SA(12,24),
     .               SF(12,4),COF,SL,GAP,STIE,SYTIE,NTIE,ATIE(8),TRACE,
     .               D,ULTIF,SJ,XD(6),S(12,12),F(48),TTT(72)
      COMMON/JUNK/ SC,TSJ,CSU,A(6,6)
      EQUIVALENCE (ENPAR,COF),(AT,TTT),(T,TTT(37))
C
C     FORM 3-D STIFFNESSES FOR EXPANSION JOINT ELEMENTS
C
      DO 5 I=1,1344
    5 LM(I)=0.
      ND=12
      NS=12
      NE=0
      WRITE(6,2000) NEXPJ,NUMNPAR
C
C     READ AND PRINT NONLINEAR ELEMENT PARAMETERS
C
      IF (NUMNPAR.EQ.0) GO TO 9
      READ (5,1010) (N,(ENPROP(N,J),J=1,5),NT(N),(XT(N,J),J=1,6),
     .                  N=1,NUMNPAR)
      WRITE(6,2001)
      WRITE(6,2002) (N,(ENPROP(N,J),J=1,5),N=1,NUMNPAR)
      WRITE(6,2003)
      WRITE(6,2004) (N,NT(N),(XT(N,J),J=1,6),N=1,NUMNPAR)
```

```
EXPJ  58       9 CONTINUE
EXPJ  59  C
EXPJ  60  C      READ AND PRINT ELEMENT DATA---DATA NEED TO BE SUPPLIED FOR EACH
EXPJ  61  C
EXPJ  62         WRITE(6,2010)
EXPJ  63      10 NE=NE+1
EXPJ  64         READ (5,1030) INE,NI,NJ,NK,NL,KD,JS,NIND,D,SQ,TRACE
EXPJ  65         IF (NIND.GT.NUMPAR) CALL ERROR (3,INE,7HEXPLB  )
EXPJ  66         IF (INE.NE.NE) CALL ERROR (4,INE,7HEXPJB   )
EXPJ  67         IF (TRACE.EQ.0.) TRACE=1.0E+10
EXPJ  68         WRITE(6,2020) NE,NI,NJ,NK,NL,KD,JS,NIND,D,SQ,TRACE
EXPJ  69  C
EXPJ  70  C      FORM TO LOCAL TRANSFORMATION
EXPJ  71  C
EXPJ  72         IF (SC.EQ.0.) GO TO 20
EXPJ  73         SQ=SQ*3.1415927/180.
EXPJ  74         TSQ=TAN(SC)
EXPJ  75         CSQ=COS(SC)
EXPJ  76         GO TO 21
EXPJ  77      20 SQ=0.
EXPJ  78         TSQ=0.0
EXPJ  79         CSQ=1.0
EXPJ  80      21 DO 25 I=1,6
EXPJ  81         DO 25 J=1,6
EXPJ  82      25 A(I,J)=0.
EXPJ  83  C
EXPJ  84         A(1,1)=1.
EXPJ  85         A(1,6)=0.5*D
EXPJ  86         A(2,2)=1.
EXPJ  87         A(3,3)=1.
EXPJ  88         A(3,4)=-A(1,6)
EXPJ  89         A(3,5)=0.5*D*TSQ
EXPJ  90         A(4,1)=1.
EXPJ  91         A(4,6)=-A(1,6)
EXPJ  92         A(5,5)=1.0/CSC
EXPJ  93         A(6,3)=1.0
EXPJ  94         A(6,4)=A(1,6)
EXPJ  95         A(6,5)=-A(3,5)
EXPJ  96  C
EXPJ  97  C      FORM LOCAL TC GLOBAL TRANSFORMATION
EXPJ  98  C
EXPJ  99         DX=X(NL)-X(NI)
EXPJ 100         DY=Y(NL)-Y(NI)
EXPJ 101         DZ=Z(NL)-Z(NI)
EXPJ 102         DL=SQRT(DX*DX+DY*DY+DZ*DZ)
EXPJ 103         IF (DL) 55,55,60
EXPJ 104      55 CALL ERROR (5,NE,7HEXPJB   )
EXPJ 105  C
EXPJ 106      60 AX=X(NK)-X(NI)
EXPJ 107         AY=Y(NK)-Y(NI)
EXPJ 108         AZ=Z(NK)-Z(NI)
EXPJ 109         BX=AY*DZ-AZ*DY
EXPJ 110         BY=AZ*DX-AX*DZ
EXPJ 111         BZ=AX*DY-AY*DX
EXPJ 112         BB=SQRT(BX*BX+BY*BY+BZ*BZ)
EXPJ 113         IF (BB) 65,65,70
EXPJ 114      65 CALL ERROR (6,NE,7HEXPJB   )
EXPJ 115  C
EXPJ 116      70 T(3,1)=BX/BB
EXPJ 117         T(3,2)=BY/BB
EXPJ 118         T(3,3)=BZ/BB
EXPJ 119         AA=SQRT(AX*AX+AY*AY+AZ*AZ)
```

```
EXPJ 120         IF (AA) 65,65,80
EXPJ 121      80 T(2,1)=-AX/AA
EXPJ 122         T(2,2)=-AY/AA
EXPJ 123         T(2,3)=-AZ/AA
EXPJ 124         T(1,1)=T(2,2)*T(3,3)-T(2,3)*T(3,2)
EXPJ 125         T(1,2)=T(2,3)*T(3,1)-T(2,1)*T(3,3)
EXPJ 126         T(1,3)=T(2,1)*T(3,2)-T(2,2)*T(3,1)
EXPJ 127  C
EXPJ 128  C      FORM JOINT TC GLOBAL TRANSFORMATION
EXPJ 129  C
EXPJ 130         DO 90 I=1,6,3
EXPJ 131      90 AT(I)=0.
EXPJ 132         DO 100 LA=1,6,3
EXPJ 133         LB=LA+2
EXPJ 134         DO 100 MA=1,6,3
EXPJ 135         MB=MA-1
EXPJ 136         DO 100 I=LA,LB
EXPJ 137         DO 100 JM=1,3
EXPJ 138         J=JM+MB
EXPJ 139         XX=0.
EXPJ 140         DO 95 K=1,3
EXPJ 141         XX=XX+A(I,K+MB)*T(K,JM)
EXPJ 142      95 CONTINUE
EXPJ 143     100 AT(I,J)=XX
EXPJ 144  C
EXPJ 145  C      FORM JOINT STIFFNESS IN JOINT COORDINATE SYSTEM
EXPJ 146  C
EXPJ 147         DO 101 I=1,144
EXPJ 148     101 S(I)=0.
EXPJ 149         DO 110 I=1,6
EXPJ 150         IF (KD(I).EQ.0) GO TO 110
EXPJ 151         S(I,I)=TRACE
EXPJ 152         S(I,I+6)=-TRACE
EXPJ 153         S(I+6,I)=-TRACE
EXPJ 154         S(I+6,I+6)=TRACE
EXPJ 155     110 CONTINUE
EXPJ 156  C
EXPJ 157  C      FORM JOINT-GLOBAL STRESS ARRAY
EXPJ 158  C
EXPJ 159         DO 120 I=1,288
EXPJ 160     120 SA(I)=0.
EXPJ 161         DO 130 LA=1,12,6
EXPJ 162         LB=LA+5
EXPJ 163         DO 130 MA=1,12,6
EXPJ 164         MB=MA-1
EXPJ 165         DO 130 I=LA,LB
EXPJ 166         DO 130 JM=1,6
EXPJ 167         J=JM+MB
EXPJ 168         XX=0.
EXPJ 169         DO 125 K=1,6
EXPJ 170         YY=S(I,K+MB)
EXPJ 171         IF (YY.EQ.0.) GO TO 125
EXPJ 172         XX=XX+YY*AT(K,JM)
EXPJ 173     125 CONTINUE
EXPJ 174     130 SA(I,J)=XX
EXPJ 175  C
EXPJ 176  C      FORM JOINT STIFFNESS IN GLOBAL COORDINATE SYSTEM
EXPJ 177  C
EXPJ 178         DO 151 I=1,576
EXPJ 179     151 ASA(I)=0.
EXPJ 180         DO 160 LA=1,12,6
EXPJ 181         LB=LA-1
```

```
EXPJ 182         DO 160 MA=1,12,6
EXPJ 183         MB=MA+5
EXPJ 184         DO 160 IL=1,6
EXPJ 185         I=IL+LB
EXPJ 186         DO 160 J=MA,MB
EXPJ 187         XX=0.
EXPJ 188         DO 155 K=1,6
EXPJ 189         XX=XX+AT(K,IL)*SA(K+LB,J)
EXPJ 190     155 CONTINUE
EXPJ 191     160 ASA(I,J)=XX
EXPJ 192   C
EXPJ 193   C     FORM ELEMENT LOCATION MATRIX
EXPJ 194   C
EXPJ 195         DO 170 I=1,6
EXPJ 196         LM(I)=ID(NI,I)
EXPJ 197         LM(I+6)=ID(NJ,I)
EXPJ 198         LM(I+12)=C
EXPJ 199     170 LM(I+18)=C
EXPJ 200   C
EXPJ 201   C     WRITE ELEMENT INFORMATION ON TAPE
EXPJ 202   C
EXPJ 203         CALL WRITET (MBAND,NDIF)
EXPJ 204   C
EXPJ 205   C     SET NONLINEAR ELEMENT INDICATOR AND STORE NONLINEAR PARAMETERS
EXPJ 206   C
EXPJ 207         NN=NUMEL+NE
EXPJ 208         IF (NIND.EQ.0) GO TO 200
EXPJ 209         IND(NN)=5
EXPJ 210         DO 180 I=1,5
EXPJ 211     180 ENPAR(I)=ENPROP(NIND,I)
EXPJ 212         NTIE=NT(NIND)
EXPJ 213         DO 190 I=1,6
EXPJ 214     190 XTIE(I)=XT(NIND,I)
EXPJ 215         UETIE=SYTIE/STIE
EXPJ 216         SJ=JS
EXPJ 217         MTYPE=5
EXPJ 218         WRITE (3) MTYPE,LM,ND,NS,ASA,SA,S,ENPAR,TTT
EXPJ 219         GO TO 250
EXPJ 220     200 IND(NN)=-5
EXPJ 221   C
EXPJ 222     250 IF (NE.LT.NEXPJ) GO TO 10
EXPJ 223         RETURN
EXPJ 224   C
EXPJ 225    1010 FORMAT (I5,5F10.0,I5/ 6F10.0)
EXPJ 226    1030 FORMAT (5I5,6I1,I4,I5,2F10.0,F20.0)
EXPJ 227    2000 FORMAT (31H1.....EXPANSION JOINT ELEMENTS    ///
EXPJ 228        .  29H NUMBER OF ELEMENTS            =  I5//
EXPJ 229        .  29H NUMBER OF N/L PAR. SETS       =  I5)
EXPJ 230    2001 FORMAT (53H1NONLINEAR PARAMETERS OF EXPANSION JOINT ELEMENTS....//
EXPJ 231        1  53H NPAR  FRICTION  SEAT    TIE     TIE         TIE YIELD     /
EXPJ 232        2  53H  NO.  COEFF.    GAP     GAP     STIFF.        FORCE       )
EXPJ 233    2002 FORMAT (I5,3F8.3,2E12.3)
EXPJ 234    2003 FORMAT (////20H JOINT TIE BARS.....  //
EXPJ 235        1  56H NPAR  NUMBER     TIE POSITIONS RELATIVE TO JOINT CENTER    /
EXPJ 236        2  59H  NO.  OF TIES   1       2       3       4       5       6  )
EXPJ 2004 FORMAT (I5,I5,3X,6F8.3)
EXPJ 238    2010 FORMAT (////34H EXPANSION JOINT ELEMENT DATA.....  //
EXPJ 239        .  5H ELEM 8X 4HNODE 11X 6H JOINT 3X 5HJOINT 5H  N/L
EXPJ 240        1  8H   JOINT 8H    SKEW 8H   JOINT /
EXPJ 241        2  5H  NO. 4X 1HI 4X 1HJ 4X 1HK 4X 1HL 3X 5H CODE 5X 4HSIGN 5H  IND
EXPJ 242        3  8H   WIDTH 9H    ANGLE 2X 9HSTIFFNESS  /)
EXPJ 243    2020 FORMAT (5I5,3X,6I1,3X,2I5,2F8.3,E12.3)
EXPJ 244         END
```

```
      SUBROUTINE PBEAM (NEL,NIND)
C
C     CHECK YIELDING OF BEAM ELEMENT AND FORM ELASTO-PLASTIC STIFFNESSES
C
      COMMON/EM/ MTYPE,LM(24),ND,NS,ASA(24,24),SA(12,24),S(12,12),PU,
     .           UPY,UMZ,TU,A0,A1,A2,A3,B0,B1,B2,B3,EPAR(12),LMS(12),
     .           EC(12),T(3,3),TTT(39),ASSA(24,24),SSA(12,24),DF(24),
     .           F(12),UP(12)
      COMMON/NEM/ EX(24),EX1(24),DEX(24),DFP(12),DP(12),FP(12),P(12),
     .            SS(12,12),U(12)
      DIMENSION FX(2),FY(2),FZ(2),FHI(2),FACT(2),
     .          PX(2),PY(2),PZ(2),PHI(2),PACT(2)
C
      DO 5 I=1,180
    5 DP(I)=0.
      DO 10 I=1,NS
   10 P(I)=F(I)
      DO 20 I=1,ND
   20 DF(I)=0.
C
      DO 60 I=1,NS
      XX=0.
      DO 50 J=1,ND
      XX=XX+SA(I,J)*DEX(J)
   50 CONTINUE
      DFP(I)=XX
   60 FP(I)=P(I)+DFP(I)
C
      FX(1)=-FP(1)/PU
      FY(1)= FP(5)/UMY
      FZ(1)= FP(6)/UMZ
      FX(2)= FP(7 )/PU
      FY(2)= FP(11)/UMY
      FZ(2)= FP(12)/UMZ
      DO 100 I=1,2
      FHI(I)=YIELD (FX(I),FY(I),FZ(I),A0,A1,A2,A3,B0,B1,B2,B3,TU)
  100 CONTINUE
C
C     FORM APPROPRIATE ELASTO-PLASTIC BEAM STIFFNESSES
C
      IF (FHI(1).GE.0.0.AND.FHI(2).LT.0.0) GO TO 300
      IF (FHI(1).LT.0.0.AND.FHI(2).GE.0.0) GO TO 400
      IF (FHI(1).GE.0.0.AND.FHI(2).GE.0.0) GO TO 500
C
C     A. BEAM REMAINING IN ELASTIC STATE
C
      DO 200 I=1,NS
  200 F(I)=FP(I)
      NIND=0
      RETURN
C
C     B. ELASTO PLASTIC BEAM WITH I-END YIELDING
C
  300 DFX=-DFP(1)/PL
      DFY= DFP(5)/UPY
      DFZ= DFP(6)/UPZ
      IF (FHI(1)) 310,310,320
  310 FACT(1)=0.
      GO TO 325
  320 FACT(1)=FACTCR (FX(1),FY(1),FZ(1),DFX,DFY,DFZ,
     .                A0,A1,A2,A3,B0,B1,B2,B3,TU    )
C
  325 EFACT=1.0-FACT(1)
      DO 330 I=1,NS
      P(I)=P(I)+EFACT*DFP(I)
  330 CONTINUE
      EFACT=FACT(1)
      CALL PTEAM (1,EFACT,NEL)
      GO TO 600
C
C     C. ELASTO-PLASTIC BEAM WITH J-END YIELDING
C
  400 DFX=DFP( 7)/PL
      DFY=DFP(11)/UPY
      DFZ=DFP(12)/UPZ
      IF (FHI(2)) 410,410,420
  410 FACT(2)=0.
      GO TO 425
  420 FACT(2)=FACTCR (FX(2),FY(2),FZ(2),DFX,DFY,DFZ,
     .                A0,A1,A2,A3,B0,B1,B2,B3,TU    )
  425 EFACT=1.0-FACT(2)
      DO 430 I=1,NS
      P(I)=P(I)+EFACT*DFP(I)
  430 CONTINUE
      EFACT=FACT(2)
      CALL PTEAM (2,EFACT,NEL)
      GO TO 600
C
C     D. ELASTO-PLASTIC BEAM WITH BOTH ENDS YIELDING
C
  500 DFX=-DFP(1)/PL
      DFY= DFP(5)/UPY
      DFZ= DFP(6)/UPZ
      IF (FHI(1)) 510,510,520
  510 FACT(1)=0.
      GO TO 525
  520 FACT(1)=FACTCR (FX(1),FY(1),FZ(1),DFX,DFY,DFZ,
     .                A0,A1,A2,A3,B0,B1,B2,B3,TU    )
C
  525 DFX=DFP( 7)/PL
      DFY=DFP(11)/UPY
      DFZ=DFP(12)/UPZ
      IF (FHI(2)) 530,530,540
  530 FACT(2)=0.
      GO TO 545
  540 FACT(2)=FACTCR (FX(2),FY(2),FZ(2),DFX,DFY,DFZ,
     .                A0,A1,A2,A3,B0,B1,B2,B3,TU    )
C
  545 EFACT=1.0-FACT(1)
      IF (FACT(1).LT.FACT(2)) EFACT=1.0-FACT(2)
      DO 550 I=1,NS
      P(I)=P(I)+EFACT*DFP(I)
  550 CONTINUE
      EFACT=FACT(1)-FACT(2)
      IF (EFACT) 570,580,560
C
  560 CALL PTEAM (1,EFACT,NEL)
      DO 565 I=1,NS
      XX=0.
      DO 561 J=1,ND
      XX=XX+SSA(I,J)*DEX(J)*EFACT
  561 CONTINUE
  565 P(I)=P(I)+XX
```

```
      EFACT=FACT(2)
      GO TO 585
C
  570 EFACT=FACT(2)-FACT(1)
      CALL PTEAM (2,EFACT,NEL)
      00 575 I=1,NS
      XX=0.
      00 571 J=1,NC
      XX=XX+SSA(I,J)*UEX(J)*EFACT
  571 CONTINUE
  575 P(I)=P(I)+XX
      EFACT=FACT(1)
      GO TO 585
C
  580 EFACT=FACT(1)
  585 CALL PTEAM (3,EFACT,NEL)
C
C     CHECK YIELDING OF NEW STRESS STATE
C
  600 DO 65C I=1,NS
      XX=0.
      DO 62C J=1,NC
      XX=XX+SSA(I,J)*DEX(J)*EFACT
  620 CONTINUE
  650 P(I)=P(I)+XX
C
      PX(1)=-P(1)/PL
      PY(1)= P(5)/LPY
      PZ(1)= P(6)/LPZ
      PX(2)=P( 7)/PU
      PY(2)=P(11)/LPY
      PZ(2)=P(12)/LPZ
      DO 655 I=1,2
      PHI(I)=YIELO (PX(I),PY(I),PZ(I),AO,A1,A2,A3,B0,B1,B2,B3,TU)
  655 CONTINUE
C
      IF (PHI(1).GE.0.0.AND.PHI(2).LT.0.0) GO TO 660
      IF (PHI(1).LT.0.0.AND.PHI(2).GE.0.0) GO TO 670
      IF (PHI(1).GE.0.0.AND.PHI(2).GE.0.0) GO TO 680
C
      NIND=0
      GO TO 760
C
  660 PACT(1)=PACTCR (PX(1),PY(1),PZ(1),AO,A1,A2,A3,B0,B1,B2,B3,TU)
      DO 661 I=1,NS
  661 P(I)=PACT(1)*P(I)
      CALL PTEAM (1,0.0,NEL)
      NIND=1
      GO TO 760
C
  670 PACT(2)=PACTCR (PX(2),PY(2),PZ(2),AO,A1,A2,A3,B0,B1,B2,B3,TU)
      DO 671 I=1,NS
  671 P(I)=PACT(2)*P(I)
      CALL PTEAM (2,0.0,NEL)
      NIND=2
      GO TO 760
C
  680 PACT(1)=PACTCR (PX(1),PY(1),PZ(1),AO,A1,A2,A3,B0,B1,B2,B3,TU)
      PACT(2)=PACTCR (PX(2),PY(2),PZ(2),AO,A1,A2,A3,B0,B1,B2,B3,TU)
      PPACT=PACT(1)
      IF (PACT(2).LT.PACT(1)) PPACT=PACT(2)
      DO 681 I=1,NS
  681 P(I)=PPACT*P(I)
      CALL PTEAM (3,0.0,NEL)
      NIND=3
C
  760 DO 761 I=1,NS
  761 F(I)=P(I)
      RETURN
      END



      SUBROUTINE PTFAM (NCOE,EFACT,NEL)
C
C     FORM NEW ELASTO-PLASTIC BEAM STIFFNESSES IN GLOBAL COORD. SYSTEM
C
      COMMON/EM/ MTYPE,LM(24),NO,NS,ASA(24,24),SA(12,24),S(12,12),PU,
     .           LPY,UMZ,TU,AO,A1,A2,A3,B0,B1,B2,B3,EPAR(12),LMS(12),
     .           EC(12),T(3,3),TTT(39),ASSA(24,24),SSA(12,24),DF(24),
     .           F(12),UP(12)
      COMMON/NEM/ EX(24),EX1(24),OEX(24),DFF(12),DP(12),FF(12),P(12),
     .            SS(12,12),UEXJ(12)
      DIMENSION YS(12,2),BS(12,2),AA(2,2),BB(2,2),UU(12),CPC(2)
C
      DO 10 J=1,2
      DO 10 I=1,12
      YS(I,J)=0.
   10 BS(I,J)=0.
      DO 20 J=1,2
      DO 20 I=1,2
   20 AA(I,J)=0.
C
      NC=1
      IF (NCOE.EQ.3) NC=2
      K=0
      SIGN=-1.0
      IF (NCOE.NE.2) GO TO 30
      K=6
      SIGN= 1.0
   30 DO 100 I=1,NC
      FX=P(K+1)*SIGN/PU
      FY=P(K+5)/UMY
      FZ=P(K+6)/UMZ
      CALL GRAUN (FX,FY,FZ,GFX,GFY,GFZ,AO,A1,A2,A3,B0,B1,B2,B3,TU)
      YS(K+1,I)=GFX*SIGN/PU
      YS(K+5,I)=GFY/UMY
      YS(K+6,I)=GFZ/UMZ
      K=K+6
      SIGN=-1.0*SIGN
  100 CONTINUE
C
      DO 200 K=1,NC
      DO 200 I=1,12
      XX=0.
      DO 125 J=1,12
      XX=XX+S(I,J)*YS(J,K)
  125 CONTINUE
  200 BS(I,K)=XX
C
      DO 300 I=1,NC
      DO 300 J=1,NC
      XX=0.
```

```
      DO 250 K=1,12
      XX=XX+YS(K,I)*BS(K,J)
  250 CONTINUE
  300 BB(I,J)=XX
C
      DET=BB(1,1)
      IF (NCCE.NE.3) GO TO 310
      DET=BB(1,1)*BB(2,2)-BB(1,2)*BB(2,1)
  310 IF (DET.LT.1.0E-30) GO TO 320
      AA(1,1)=1.0/DET
      IF (NCDE.NE.3) GO TO 320
      AA(1,1)= BB(2,2)/DET
      AA(1,2)=-BB(1,2)/DET
      AA(2,1)= AA(1,2)
      AA(2,2)= BB(1,1)/DET
C
  320 DO 400 I=1,12
      DO 380 J=1,12
      XX=0.
      DO 350 K=1,NC
      DO 350 L=1,NC
      XX=XX+BS(I,K)*AA(K,L)*BS(J,L)
  350 CONTINUE
  380 SS(I,J)=S(I,J)-XX
      IF (SS(I,I).LT.0.1*S(I,I)) SS(I,I)=0.1*S(I,I)
  400 CONTINUE
C
C     TRANSFORM TO CLOBAL COORD. SYSTEM
C
      DO 410 I=1,288
  410 SSA(I)=0.
      DO 450 LA=1,10,3
      LB=LA+2
      DO 450 MA=1,10,3
      MB=MA-1
      DO 450 I=LA,LB
      DO 450 JM=1,3
      J=JM+MB
      XX=0.
      DO 440 KK=1,3
      XX=XX+SS(I,KK+MB)*T(KK,JM)
  440 CONTINUE
  450 SSA(I,J)=XX
C
      DO 452 I=1,576
  452 ASSA(I)=0.
      DO 460 LA=1,10,3
      LB=LA-1
      DO 460 MA=1,10,3
      MB=MA+2
      DO 460 IL=1,3
      I=IL+LA
      DO 460 J=MA,MB
      XX=0.
      DO 455 KK=1,3
      XX=XX+T(KK,IL)*SSA(KK+LB,J)
  455 CONTINUE
  460 ASSA(I,J)=XX
C
C     TRANSFORM TO MASTER DEGREES OF FREEDOM
C
      DO 461 I=1,12
  461 UU(I)=DEX(I)
      KK=0
      KD=12
      DO 500 NF=1,12,6
      DO 490 K=1,3
      I=K+NF-1
      IF (LPS(I).LE.0) GO TO 490
      KK=KK+1
      D1=EC(KK)
      KK=KK+1
      D2=EC(KK)
      DO 470 II=1,KC
      ASSA(KD+1,II)=ASSA(I,II)*D1
      ASSA(KD+2,II)=ASSA(I,II)*D2
      ASSA(II,KD+1)=ASSA(II,I)*D1
      ASSA(II,KD+2)=ASSA(II,I)*D2
  470 CONTINUE
      DO 480 II=1,NS
      SSA(II,KD+1)=SSA(II,I)*D1
      SSA(II,KD+2)=SSA(II,I)*D2
  480 CONTINUE
      ASSA(KD+1,KD+1)=ASSA(I,I)*D1**2
      ASSA(KD+2,KD+2)=ASSA(I,I)*D2**2
      ASSA(KD+1,KD+2)=ASSA(I,I)*D1*D2
      ASSA(KD+2,KD+1)=ASSA(KD+1,KD+2)
      UU(I)=DEX(I)+CEX(KD+1)*D1+DEX(KD+2)*D2
      KD=KD+2
  490 CONTINUE
  500 CONTINUE.
C
C     COMPUTE ELEMENT PLASTIC DEFORMATIONS
C
      IF (EFACT.EQ.0.0) GO TO 600
      IF (DET.LT.1.0E-30) GO TO 600
      DO 540 LA=1,10,3
      LB=LA+2
      MB=LA-1
      DO 540 I=LA,LB
      II=I-MB
      XX=0.
      DO 530 K=1,3
      XX=XX+T(K,II)*UU(K+MB)
  530 CONTINUE
  540 DEXJ(I)=XX
C
      DO 550 K=1,NC
      XX=0.
      DO 545 L=1,NC
      DO 545 J=1,12
  545 XX=XX+AA(K,L)*BS(J,L)*DEXJ(J)*EFACT
  550 DPD(K)=ABS(XX)
C
      DO 560 I=1,12
      XX=0.
      DO 555 K=1,NC
  555 XX=XX+YS(I,K)*DPD(K)
  560 UP(I)=UP(I)+XX
C
  600 RETURN
      END
```

```
      FUNCTION YIELD (PX,PY,PZ,A0,A1,A2,A3,B0,B1,B2,B3,TU)
C
C     DETERMINE VALUE OF YIELD FUNCTION
C
      IF (PX.LT.-0.9) GO TO 100
      IF (PX.GT.TU-0.1) GO TO 100
C
      PXY=A0+A1*PX+A2*PX**2+A3*PX**3
      PXZ=B0+B1*PX+B2*PX**2+B3*PX**3
      YIELD=(PY/PXY)**2+(PZ/PXZ)**2-1.0
      RETURN
C
  100 YIELD=1.0
      RETURN
      END


      FUNCTION FACTOR (TX,TY,TZ,DTX,DTY,DTZ,A0,A1,A2,A3,B0,B1,B2,B3,TU)
C
C     DETERMINE OVERSHOOT FACTOR OF STRESS INCREMENT TO YIELD SURFACE
C
      C=0.0
      IF (ABS(DTX).LT.1.0E-30) GO TO 5
      IF (TX.LE.-1.0) C=(TX+1.0)/DTX
      IF (TX.GE. TU ) C=(TX-TU )/DTX
    5 FX=TX-C*DTX
      FY=TY-C*DTY
      FZ=TZ-C*DTZ
      DFX=(1.0 C)*DTX
      DFY=(1.0 C)*DTY
      DFZ=(1.0-C)*DTZ
      FXY=A0+A1*FX+A2*FX**2+A3*FX**3
      FXZ=B0+B1*FX+B2*FX**2+B3*FX**3
C
      N=0
      ACTOR=1.0
      Y0=FXY
      Y1=A1*DFX+2.0*A2*FX*DFX+3.0*A3*FX*FX*DFX
      Y2=A2*DFX*DFX+3.0*A3*FX*DFX*DFX
      Y3=A3*DFX*DFX*DFX
      Z0=FXZ
      Z1=B1*DFX+2.0*B2*FX*DFX+3.0*B3*FX*FX*DFX
      Z2=B2*DFX*DFX+3.0*B3*FX*DFX*DFX
      Z3=B3*DFX*DFX*DFX
C
   10 N=N+1
      YY =Y0-Y1*ACTOR+Y2*ACTOR**2-Y3*ACTOR**3
      YY1=FY-DFY*ACTOR
      YY2=-Y1+2.0*Y2*ACTOR-3.0*Y3*ACTOR**2
      ZZ =Z0-Z1*ACTOR+Z2*ACTOR**2-Z3*ACTOR**3
      ZZ1=FZ-DFZ*ACTOR
      ZZ2=-Z1+2.0*Z2*ACTOR-3.0*Z3*ACTOR**2
      IF (ABS(YY).LT.1.0E-30.OR.ABS(ZZ).LT.1.0E-30) GO TO 120
      P =(YY1/YY)**2+(ZZ1/ZZ)**2-1.0
C
      IF (ABS(P).LE.1.0E-10) GO TO 100
      IF (N.GT.100) GO TO 100
      DP=(2.0*YY1*(-DFY*YY-YY1*YY2)/(YY**3)) +
     *   (2.0*ZZ1*(-DFZ*ZZ-ZZ1*ZZ2)/(ZZ**3))
      IF (ABS(DP).LT.1.0E-30) GO TO 120
      ACTOR=ACTOR-P/DP
      GO TO 10
C
  100 FACTOR=ACTOR*(1.0-C)+C+1.0E-10
      IF (FACTOR.GT.1.0) FACTOR=1.0
      IF (FACTOR.LT.0.0) FACTOR=0.0
      RETURN
C
  120 ACTOR=1.0
      GO TO 100
      END
```

```
      FUNCTION PACTOR (TX,TY,TZ,A0,A1,A2,A3,B0,B1,B2,B3,TU)
C
C     DETERMINE REDUCTION FACTOR OF TOTAL STRESS VECTOR TO YIELD SURFACE
C
      TT=TU-0.1
      C =1.0
      IF (TX.LT.-0.9) C=-0.9/TX
      IF (TX.GT. TT ) C=  TT/TX
      FX=C*TX
      FY=C*TY
      FZ=C*TZ
      N=0
      ACTOR=1.0
C
   10 N=N+1
      PX=ACTOR*FX
      PY=ACTOR*FY
      PZ=ACTOR*FZ
      PXY=A0+A1*PX+A2*PX**2+A3*PX**3
      PXZ=B0+B1*PX+B2*PX**2+B3*PX**3
      IF (PXY.LT.1.0E-10.OR.PXZ.LT.1.0E-10) GO TO 200
      FP=(PY/PXY)**2+(PZ/PXZ)**2 1.0
      IF (ABS(FP).LE.1.0E-10) GO TO 100
      IF (N.GT.100) GO TO 200
      YY=A1*FX+2.0*A2*FX*PX+3.0*A3*FX*PX**2
      ZZ=B1*FX+2.0*B2*FX*PX+3.0*B3*FX*PX**2
      DP=2.0*PY*(FY*PXY-PY*YY)/PXY**3+2.0*PZ*(FZ*PXZ-PZ*ZZ)/PXZ**3
      ACTOR=ACTOR-FP/DP
      IF (ABS(ACTOR).GT.1.0E 10) GO TO 200
      GO TO 10
C
  100 PACTOR=ACTOR*C-1.0E- 5
      IF (PACTOR.GT.1.0) PACTOR=1.0
      IF (PACTOR.LT.0.0) PACTOR=0.0
      RETURN
C
  200 N=0
      PMIN=0.
      PMAX=1.0
      IF (TX.LT.-1.0) PMAX=-1.0/TX
      IF (TX.GT. TU ) PMAX= 1.0/TX
C
  210 N=N+1
      P=(PMAX+PMIN)/2.0
      PX=P*TX
```

```
      PY=P*TY
      PZ=P*TZ
      IF (PX.GE.TU.CR.PX.LE.-1.0) GO TO 300
      PXY=AO+A1*PX+A2*PX**2+A3*PX**3
      FXZ=80+81*PX+82*PX**2+83*PX**3
      IF (PXY.LT.I.CE- 5.OR.PXZ.LT.1.0E- 5) GO TO 450
      FP=(FY/PXY)**2+(PZ/PXZ)**2-1.0
      GO TO 400
  300 FP=1.C
  400 IF (FP.GT.-1.CE-05.AND.FP.LT.0.) GO TO 500
      IF (N.GT.100) GO TO 450
      IF (FP) 41C,5C0,420
  410 PMIN=P
      GO TO 210
  420 PMAX=P
      GO TO 210
C
  450 P=PMIN
  500 PACTOR=P
      RETURN
      END


      SUBROUTINE GRAON (FX,FY,FZ,GFX,GFY,GFZ,AO,A1,A2,A3,80,81,82,83,TU)
C
C     CCMPLTE GRADIENT TO YIELO SURFACE
C
      IF (FX.GT.TU-C.1) GO TO 100
      FXY=AC+A1*FX+A2*FX**2+A3*FX**3
      FXZ=8C+81*FX+82*FX**2+83*FX**3
      FYY=A1+2.C*A2*FX+3.0*A3*FX*FX
      FZZ=81+2.C*82*FX+3.0*83*FX*FX
C
      GFX=(-2.0*FY*FY*FYY)/(FXY**3)+(-2.0*FZ*FZ*FZZ)/(FXZ**3)
      GFY=( 2.0*FY)/(FXY**2)
      GFZ=( 2.0*FZ)/(FXZ**2)
      RETURN
C
  100 GFX=1.0
      GFY=0.0
      GFZ=0.0
      RETURN
      END
```

```
      SUBROUTINE PBCUND (NEL,NINO)
C
C     CHECK YIELOING OF BOUNOARY ELEMENTS AND FORM BILINEAR STIFFNESSES
C
      COMMON/EM/ MTYPE,LM(24),ND,NS,ASA(24,24),SA(12,24),S(12,12),
     .           SY(6),EP(6),UE(6),PAR(6),T(3,3),TTT(63),ASSA(24,24),
     .           SSA(12,24),UF(24),F(12),UP(12)
      COMMON/NEM/ EX(24),EX1(24),DEX(24),OFF(12),OP(12),FF(12),P(12),
     .            SS(6,6),OEXJ(6),KY(6),FACTOR(6)
C
      OO 5 I=1,SC
    5 OP(I)=0.
      OO 1C I=1,NS
      XX=0.
      OO 8 J=1,NO
      XX=XX+SA(I,J)*OEX(J)
    8 CONTINUE
      DFF(I)=XX
      FF(I)=F(I)+OFF(I)
   10 P(I)=F(I)
C
C     CHECK YIELCINC OF BOUNDARY ELEMENTS
C
      OC 20 I=1,NS
      KY(I)=0
      FACTCR(I)=0.
      AA=ABS(DFF(I))
      XX=SY(I)+EP(I)*UP(I)
      YY=SY(I)-EP(I)*UP(I)
      IF (FF(I).LE.XX) GO TO 15
      KY(I)=1
      IF (AA.LT.1.CE-10) GO TO 20
      FACTOR(I)=(FF(I)-XX)/DFF(I)
      GC TO 20
   15 IF (FF(I).GE.-YY) GO TO 20
      KY(I)=-1
      IF (AA.LT.1.CE-10) GO TO 20
      FACTOR(I)=(FF(I)+YY)/DFF(I)
   20 CONTINUE
C
      OO 30 I=1,NS
      IF (KY(I).NE.C) GO TO 100
   30 CONTINUE
C
C     A. ELEMENT IN ELASTIC STATE
C
      OO 50 I=1,NS
   50 F(I)=FF(I)
      NIND=C
      GO TC 500
C
C     B. ELEMENT IN PLASTIC STATE
C
C     81. CCMPUTE ELEMENT STRESSES IN PLASTIC STATE
C
  100 OO 120 LA=1,6,3
      LB=LA+2
      MB=LA-1
      OC 12C I=LA,LB
      II=I-MB
      XX=0.
```

```
      DO 115 K=1,3
  115 XX=XX+T(II,K)*DEX(K+MB)
  120 DEXJ(I)=XX
C
      DO 132 I=1,6
      DO 132 J=1,6
  132 SS(I,J)=S(I,J)
      DO 140 I=1,6
      IF (KY(I)) 135,140,135
  135 SS(I,I)=EP(I)
  140 CONTINUE
C
      DO 150 I=1,NS
      P(I)=P(I)+(1.0-FACTOR(I))*OFF(I)+FACTOR(I)*SS(I,I)*DEXJ(I)
      F(I)=P(I)
      UP(I)=UP(I)+FACTOR(I)*DEXJ(I)
  150 CONTINUE
C
C     B2. FORM BILINEAR STIFFNESSES IN GLOBAL COORD. SYSTEM
C
      DO 180 I=1,288
  180 SSA(I)=0.
      DO 200 LA=1,6,3
      LB=LA+2
      DO 200 MA=1,6,3
      MB=MA-1
      DO 200 I=LA,LB
      DO 200 JM=1,3
      J=JM+MB
      XX=0.
      DO 190 K=1,3
      YY=SS(I,K+MB)
      IF (YY.EQ.0.) GO TO 190
      XX=XX+YY*T(K,JM)
  190 CONTINUE
  200 SA(I,J)=XX
C
      DO 210 I=1,576
  210 ASSA(I)=0.
      DO 250 LA=1,6,3
      LB=LA-1
      DO 250 MA=1,6,3
      MB=MA+2
      DO 250 IL=1,3
      I=IL+MB
      DO 250 J=MA,MB
      XX=0.
      DO 240 K=1,3
      XX=XX+T(K,IL)*SSA(K+LB,J)
  240 CONTINUE
  250 ASSA(I,J)=XX
      NIND=1
C
  500 DO 600 I=1,NC
  600 DF(I)=0.
      RETURN
      END
```

```
      SUBROUTINE NEXPJB (NEL,NIND)
C
C     CHECK JOINT CONDITIONS AND FORM NEW JOINT STIFFNESSES IF NEEDED
C
      COMMON/EM/ MTYPE,LM(24),ND,NS,ASA(24,24),SA(12,24),S(12,12),
     .           CCF,GAP,TG,STIE,SYTIE,NTIE,XTIE(6),TRACE,C,ULTIE,
     .           SJ,KD(6),A(6,6),T(3,3),PTIE,KKD(6),TTT(20),ASSA(24,24),
     .           SSA(12,24),DF(24),F(6),FT(6),U(6),UPTIE(6)
      COMMON/NEW/ EX(24),EXI(24),DEX(24),DFF(12),DP(12),EXJ(12),EXJI(12)
     .           ,SS(12,12),UI(6),TK(6)
      COMMON/COLLOMB/ CF(20,6)
      DIMENSION CFF(6)
C
      DO 10 I=1,12
   10 DP(I)=0.
      DO 12 I=1,ND
   12 DF(I)=0.
C
C     COMPUTE LOCAL JOINT RELATIVE DISPLACEMENTS AND VELOCITIES
C
      DO 20 LA=1,12,6
      LB=LA+5
      MB=LA-1
      DO 20 I=LA,LB
      II=I-MB
      XX=0.
      YY=0.
      DO 15 KK=1,6
      K=KK+MB
      XX=XX+A(II,KK)*EX(K)
      YY=YY+A(II,KK)*EXI(K)
   15 CONTINUE
      EXJ (I)=XX
      EXJI(I)=YY
   20 CONTINUE
      DO 50 I=1,6
      U (I)=EXJ (I+6)-EXJ (I)
      UI(I)=EXJI(I+6)-EXJI(I)
   50 CONTINUE
C
C     CHECK CONNECTIVITY OF THE EXPANSION JOINT
C
      PTIE=1000.
      IF (NTIE.EQ.0) GO TO 61
      DO 60 I=1,NTIE
      IF (PTIE.GT.UPTIE(I)) PTIE=UPTIE(I)
   60 CONTINUE
   61 PTIE=PTIE+TG
      DO 65 I=1,6
   65 KKO(I)=KD(I)
C
      IF (U(1).LT.-CAP.AND.UI(1).LT.0.0) KKD(1)=-1
      IF (U(1).GT.PTIE) KKD(1)= 1
C
      IF (U(4).LT.-CAP.AND.UI(4).LT.0.0) KKD(4)=-1
      IF (U(4).GT.PTIE) KKD(4)= 1
C
      DO 70 I=1,6
      IF (KKD(I).NE.KU(I)) GO TO 100
   70 CONTINUE
C
```

```
      DO 80 I=1,6
      XX=0.
      DO 75 J=1,NO
      XX=XX+SA(I,J)*DEX(J)
   75 CONTINUE
   80 DFF(I)=XX
      NIND=0
      GO TO 120
C
C     FORM NEW EXPANSION JOINT CONDITIONS
C
  100 CALL NXPJB
      DO 110 I=1,6
      XX=0.
      DO 105 J=1,ND
      XX=XX+SSA(I,J)*DEX(J)
  105 CONTINUE
  110 DFF(I)=XX
      NIND=1
C
  120 DO 130 I=1,6,3
      IF (DFF(I).EQ.0.0) F(I)=0.0
  130 CONTINUE
      DO 150 I=1,6
  150 F(I)=F(I)+DFF(I)
C
C     COMPUTE COULOMB FRICTIONAL FORCES
C
      IF (SJ*F(3).LE.0.0.AND.SJ*F(6).LE.0.0) RETURN
      DO 200 I=1,6
  200 CFF(I)=0.
      DO 250 I=1,6,3
      II=I+2
      IF (SJ*F(II).LE.0.0) GO TO 250
  215 IF (U1(I)) 220,220,225
  220 CFF(I)=-CCF*SJ*F(II)
      GO TO 240
  225 CFF(I)= CCF*SJ*F(II)
  240 DCF=CFF(I)-CF(NEL,I)
      DP(I)=DP(I)+DCF
      DP(I+6)=-DP(I)
      CF(NEL,I)=CFF(I)
  250 CONTINUE
C
C     TRANSFORM FRICTIONAL FORCE INCREMENTS TO GOLBAL COORD. SYSTEM
C
      DO 300 LA=1,12,6
      LB=LA-1
      DO 300 IL=1,6
      I=IL+LB
      XX=0.
      DO 255 K=1,6
      XX=XX+A(K,IL)*DP(K+LB)
  255 CONTINUE
  300 DF(I)=XX
      RETURN
      END


      SUBROUTINE NXPJB
C
C     FORM NEW EXPANSION JOINT STIFFNESSES
C
      COMMON/EM/ MTYPE,LM(24),ND,NS,ASA(24,24),SA(12,24),S(12,12),
     *           CCF,GAP,TG,STIE,SYTIE,NTIE,XTIE(8),TRACE,D,UETIE,
     *           SJ,KO(6),A(6,6),T(3,3),PTIE,KKD(6),YTT(20),ASSA(24,24),
     *           SSA(12,24),DF(24),F(6),FT(6),U(6),UPTIE(6)
      COMMON/NEW/ EEEE(72),DFF(12),DP(12),EXJ(12),EXJ1(12),SS(12,12),
     *            U1(6),TK(6)
C
C     COMPUTE TIE STIFFNESSES
C
      DO 5 I=1,6
    5 TK(I)=0.
      IF (NTIE.EQ.0) GO TO 70
C
      DO 12 I=1,NTIE
      XX=(U(1)+U(4))/2.0+XTIE(I)*(U(1)-U(4))/D
      XX=XX-TG
      IF (XX.LT.UPTIE(I)) GO TO 10
      IF (XX-UPTIE(I)-UETIE) 9,9,8
    8 UPTIE(I)=XX-UETIE
      TK(I)=0.05
      FT(I)=SYTIE
      GO TO 12
    9 TK(I)=1.0
      FT(I)=STIE*(XX-UPTIE(I))
      GO TO 12
   10 FT(I)=0.
   12 CONTINUE
C
      TAA=0.
      TAB=0.
      TBB=0.
      DO 20 I=1,NTIE
      TAA=TAA+TK(I)*STIE*(D/2.0+XTIE(I))**2/D**2
      TAB=TAB+TK(I)*STIE*(D/2.0+XTIE(I))*(D/2.0-XTIE(I))/D**2
      TBB=TBB+TK(I)*STIE*(D/2.0-XTIE(I))**2/D**2
   20 CONTINUE
      GO TO 80
C
   70 TAA=0.
      TAB=0.
      TBB=0.
C
C     FORM NEW LOCAL JOINT STIFFNESSES
C
   80 DO 100 I=1,144
  100 SS(I)=0.
C
      IF (KKD(1)) 110,120,115
  110 SS(1,1)=TRACE
      GO TO 120
  115 SS(1,1)=TAA
      SS(1,4)=TAB
      SS(4,1)=TAB
      SS(4,4)=TBB
C
  120 IF (KKD(2)) 140,140,130
  130 SS(2,2)=TRACE
C
  140 IF (KKD(3)) 160,160,150
```

```
NXPJ 182     150 SS(3,3)=TRACE
NXPJ 183   C
NXPJ 184     160 IF (KKD(4)) 170,180,175
NXPJ 185     170 SS(4,4)=TRACE
NXPJ 186         GO TO 180
NXPJ 187     175 SS(1,1)=TAA
NXPJ 188         SS(1,4)=TAB
NXPJ 189         SS(4,1)=TAB
NXPJ 190         SS(4,4)=TBB
NXPJ 191   C
NXPJ 192     180 IF (KKD(5)) 200,200,190
NXPJ 193     190 SS(5,5)=TRACE
NXPJ 194   C
NXPJ 195     200 IF (KKD(6)) 220,220,210
NXPJ 196     210 SS(6,6)=TRACE
NXPJ 197   C
NXPJ 198     220 DO 250 LA=1,12,6
NXPJ 199         LB=LA+5
NXPJ 200         DO 250 MA=1,12,6
NXPJ 201         MB=MA+5
NXPJ 202         IF (LA.EQ.1.AND.MA.EQ.1) GO TO 250
NXPJ 203         SIGN=-1.0
NXPJ 204         IF (LA.EQ.7.AND.MA.EQ.7) SIGN= 1.0
NXPJ 205         DO 240 I=LA,LB
NXPJ 206         II=I-LA+1
NXPJ 207         DO 240 J=MA,MB
NXPJ 209         JJ=J-MA+1
NXPJ 209         SS(I,J)=SIGN*SS(II,JJ)
NXPJ 210     240 CONTINUE
NXPJ 211     250 CONTINUE
NXPJ 212   C
NXPJ 213   C     TRANSFORM TO GLOBAL COORD. SYSTEM
NXPJ 214   C
NXPJ 215         DO 310 I=1,288
NXPJ 216     310 SSA(I)=0.
NXPJ 217         DO 330 LA=1,12,6
NXPJ 218         LB=LA+5
NXPJ 219         DO 330 MA=1,12,6
NXPJ 220         MB=MA-1
NXPJ 221         DO 330 I=LA,LB
NXPJ 222         DO 330 JM=1,6
NXPJ 223         J=JM+MB
NXPJ 224         XX=0.
NXPJ 225         DO 325 K=1,6
NXPJ 226         YY=SS(I,K+MA)
NXPJ 227         IF (YY.EQ.0.) GO TO 325
NXPJ 228         XX=XX+YY*A(K,JM)
NXPJ 229     325 CONTINUE
NXPJ 230     330 SSA(I,J)=XX
NXPJ 231   C
NXPJ 232         DO 351 I=1,576
NXPJ 233     351 ASSA(I)=0.
NXPJ 234         DO 360 LA=1,12,6
NXPJ 235         LB=LA-1
NXPJ 236         DO 360 MA=1,12,6
NXPJ 237         MB=MA+5
NXPJ 238         DO 360 IL=1,6
NXPJ 239         I=IL+LB
NXPJ 240         DO 360 J=MA,MB
NXPJ 241         XX=0.
NXPJ 242         DO 355 K=1,6
NXPJ 243         XX=XX+A(K,IL)*SSA(K+LB,J)
NXPJ 244     355 CONTINUE
NXPJ 245     360 ASSA(I,J)=XX
NXPJ 246         RETURN
NXPJ 247         END
```

FHWA
R&D