

DTIC DOCUMENTATION PAGE

1a. REPORT SECURITY CLASSIFICATION  
Unclassified

1b. RESTRICTIVE MARKINGS

2a. SECURITY CLASSIFICATION AUTHORITY  
AD-A219 622  
MAR 12 1990

3. DISTRIBUTION / AVAILABILITY OF REPORT  
Approved for public release;  
distribution is unlimited.

6a. NAME OF PERFORMING ORGANIZATION  
Stanford University

5. MONITORING ORGANIZATION REPORT NUMBER(S)  
AFOSR-TR- 90 - 0 2 9 9

6a. NAME OF PERFORMING ORGANIZATION  
Stanford University

6b. OFFICE SYMBOL  
(if applicable)

7a. NAME OF MONITORING ORGANIZATION  
AFOSR/NA

6c. ADDRESS (City, State, and ZIP Code)  
Department of Aeronautics and Astronautics  
Stanford University  
Stanford, California 94305

7b. ADDRESS (City, State, and ZIP Code)  
Building 410, Bolling AFB DC  
20332-6448

8a. NAME OF FUNDING / SPONSORING ORGANIZATION  
AFOSR

8b. OFFICE SYMBOL  
(if applicable)  
NA

9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER  
AFOSR-88-0015

8c. ADDRESS (City, State, and ZIP Code)  
Building 410. Bolling AFB DC  
20332-6448

10. SOURCE OF FUNDING NUMBERS  
PROGRAM ELEMENT NO. 61102F  
PROJECT NO. 2302  
TASK NO. B1  
WORK UNIT ACCESSION NO.

11. TITLE (Include Security Classification)  
(U) MODELING AND CONTROL OF FLEXIBLE VEHICLES IN SPACE

12. PERSONAL AUTHOR(S)  
A.E. Bryson, and F.A. Wiesinger

13a. TYPE OF REPORT  
Final Technical

13b. TIME COVERED  
FROM 10/87 TO 10/89

14. DATE OF REPORT (Year, Month, Day)  
2/12/90

15. PAGE COUNT  
47

16. SUPPLEMENTARY NOTATION

17. COSATI CODES		
FIELD	GROUP	SUB-GROUP

18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)  
Modelling of circular flat plates, Conversion of modal data to transfer function form, Shape control of flat plate, Decoupling of high order systems via symmetry,

19. ABSTRACT (Continue on reverse if necessary and identify by block number) - Space Parts - JG  
Several Aspects of modeling and control of large space structures have been investigated. A FORTRAN computer program is included which transforms modal data from commercially available finite element codes for personal computers to the forms needed for control system design. Design of a low order controller for a circular flat plate space structure is investigated by utilizing geometric symmetry of the structure and the control system actuators and sensors to decouple the system into smaller, single input -single output subsystems.

20. DISTRIBUTION / AVAILABILITY OF ABSTRACT  
 UNCLASSIFIED/UNLIMITED  SAME AS RPT  DTIC USERS

21. ABSTRACT SECURITY CLASSIFICATION  
UNCLASSIFIED

22a. NAME OF RESPONSIBLE INDIVIDUAL  
ANTHONY K. AMOS

22b. TELEPHONE (Include Area Code)  
(202)767-4937

22c. OFFICE SYMBOL  
AFOSR/NA

**Final Report for AFOSR 88-0015  
for the Period  
October 15,1988 to October 15,1989  
Modeling and Control of Flexible Vehicles in Space**

Over the period 15 September 1987 to 15 October 1989 several aspects of the modeling and control of flexible space structures have been investigated:

1. The reduction of modal data to forms useful for control system designers.
2. Design of shape control systems for two and three dimensional structures with many vibration modes by decoupling into smaller subsystems, each with fewer controls and outputs.

**Transformation of Modal Data from Finite Element Codes to a Format for Control Design**

The efficient reduction of modal data obtained from finite element codes to a form useful for control system designers was realized through the use of a program we developed called TRANSGEN. This program is useful for producing state space models of two or three dimensional structures having many sensors and actuators. TRANSGEN accepts as input the output of PAL, a personal computer version of NASTRAN. Codes of this type exist for manipulating NASTRAN output [1], but we believe this is the first such code produced for PAL output.

While the generation of state space models for flexible beams or other one dimensional structures can be performed by using standard techniques, such as the assumed modes method, more complex multi-dimensional structures must be analyzed using finite element methods. Typically the designer of a control system for such a structure is given a modal form of the system equations of motion and must determine:

- (1) Which sensor and actuator pairs should be chosen for control design.

- (2) Which of the many modes to keep in the system description
- (3) The relative importance of each mode for each individual sensor/actuator transfer function.

By obtaining the pole/residue, pole/zero, and state space forms of the system description automatically, TRANSGEN allows the designer to experiment with different linear or rotational sensor and actuator placements and different model order reductions. A complete description of TRANSGEN, including operating instructions and examples, is provided in Appendix 1.

### **Two-Dimensional Flexible Structure Control Design by Input/Output Decoupling into Smaller Subsystems**

#### Modelling

One of the salient features of many flexible space structures is a considerable number of modes with closely spaced (or even repeated) eigenvalues with low damping. This arises from symmetries or periodicity in the geometric shape of the structure.

As a representative space structure with these features, we chose a flat, circular plate whose control had been investigated experimentally by previous researchers [2]. By modeling the plate with lumped masses representing actuator/sensor packages at symmetric locations about the x and y axes, we were able to retain symmetry in the mode shapes while increasing the realism of the model. The masses caused some small symmetric warping of the mode shapes compared to the mode shapes of the plate alone, as described below.

The finite element discretization used in PAL, as well as the placement of the sensor/actuator lumped masses, is depicted in Fig 1. Figure 2 shows the mode shapes for the circular plate if no lumped masses are added, while Figure 3 shows the mode shapes for the plate with lumped masses. As seen in Figure 2, nodal (or zero displacement) lines lie at various increments of 30 and 45 degrees around the plate. A translation sensor or force actuator placed anywhere along these lines will cause that mode to be either unobservable or uncontrollable, respectively, from that sensor or actuator. Hence we decided to place 4 sensor/actuator pairs symmetrically about the x and y axes as seen in

	<input checked="" type="checkbox"/>
	<input type="checkbox"/>
	<input type="checkbox"/>
y Codes	
	Special
A-1	

Figure 1. Each pair was modelled as a lumped mass of 1/100th of the plate mass. An additional lumped mass was included at the center of the plate to approximate the mass of a rigid feed tower.

From Figure 3 we see that modes number 4 & 8 show considerable symmetric warping, but translation sensors and actuators at  $15^\circ$  will still be able to observe and control all the modes depicted since none have nodal lines at those locations. Placing the sensor/actuator pairs at the edge of the plate provides maximum observability/controllability since most mode shapes have maximum deflection at the edge.

TRANSGEN was used to create a 16th order state space model of the plate assuming 4 sensor/actuator pairs placed at  $\pm 15$  degrees from the positive and negative y axis. The resulting system F,G, and H matrices are shown in Table 1.

### Control Design

By examining the mode shapes it was seen that they were separable into 4 groups: symmetric about the x axis, symmetric about the y axis, symmetric about the x and y axes, and anti-symmetric about the x and y axes. The groupings are shown in Figure 4 . By forming linear combinations of the inputs and outputs of the system, the G and H matrices were decoupled into four subsystems, as shown in Figure 5. This procedure is similar to one described in [3].

The symmetry to subsystem correspondence is shown in Figure 4. Figure 4 also indicates whether rotations about, or translations of, the x and y axes occur for each subsystem. This information is useful since a quarter plate could be modeled with these sets of boundary conditions in order to produce the same modal displacement and frequency information. Doing so greatly reduces the size of the finite element model and thus allows faster analysis.

Each subsystem is SISO (single input single output) and has the properties of a collocated system , i.e. alternating poles and zeros in the transfer function. Simple lead-controllers could then be used to damp each mode. Root loci versus lead gain are illustrated in Figures 6 through 9 for the four subsystems. The unmodeled, higher frequency modes of each subsystem could also be damped by the lead controllers since they would follow an alternating pole-zero pattern. The resulting compensator for the entire plate was therefore

fourth order.

The performance of the closed loop system was evaluated by simulating impulse responses for a variety of disturbances. Figure 10a shows the rotation about the x-axis at the center of the plate (not sensed, but constructed from the mode responses) for an impulsive disturbance torque about the x-axis. Figure 10b shows the output of sensor 1 (z translation), and figures 10c and d show the control force from actuators 1 and 2 (also in the z direction). Sensors 2, 3, and 4 showed responses either equal to the response of sensor 1 or just the inverse of it. Actuators 3 and 4 responded exactly opposite to actuators 1 and 2.

The plots in Figure 11(a-d) are analogous to those in Figure 10 but for a y-axis impulsive disturbance torque. Figures 12 show the results of a force impulse at the location of actuator 1.

Impulse responses for the total closed loop system showed well-damped behavior and it didn't appear that any of the controllers were "fighting" each other, as might occur with poorly designed controllers for multi-input multi-output systems.

### **Conclusion**

Having gained better understanding of the methods involved in modeling large, two-dimensional space structures via the finite element approach, and some of the simplifications inherent in the control system design by utilizing system symmetry, we believed a logical extension of our work would be to incorporate simultaneous structural and control system design. This is thought to be quite useful since a structure can be designed to be much lighter if an active control system is present to minimize structural deflections, and therefore strains, in response to dynamic loads from disturbances, slew maneuvers, etc. Our research was proceeding in this direction when AFOSR funding was stopped.

## References

- [1] Chu, P. and Plescia, C., "Adaptive Rigid Body Control for an Evolving Space Station, Preliminary Analysis Report", Ford Aerospace document MDL-TR10882, November 1986.
- [2] Aubrun, J. N., Ratner, M. J and Lyons, M. G., "Structural Control for a Circular Plate", *Journal of Guidance, Control and Dynamics*, vol. 7, no. 5, September-October 1984, pp. 535-545.
- [3] Bryson, A. E., "Modeling of Flexible Structures for Active Control", *Workshop on Applications of Distributed System Theory to the Control of Large Space Structures*, NASA JPLPublication 83-46, July 1983, pp. 103-118.

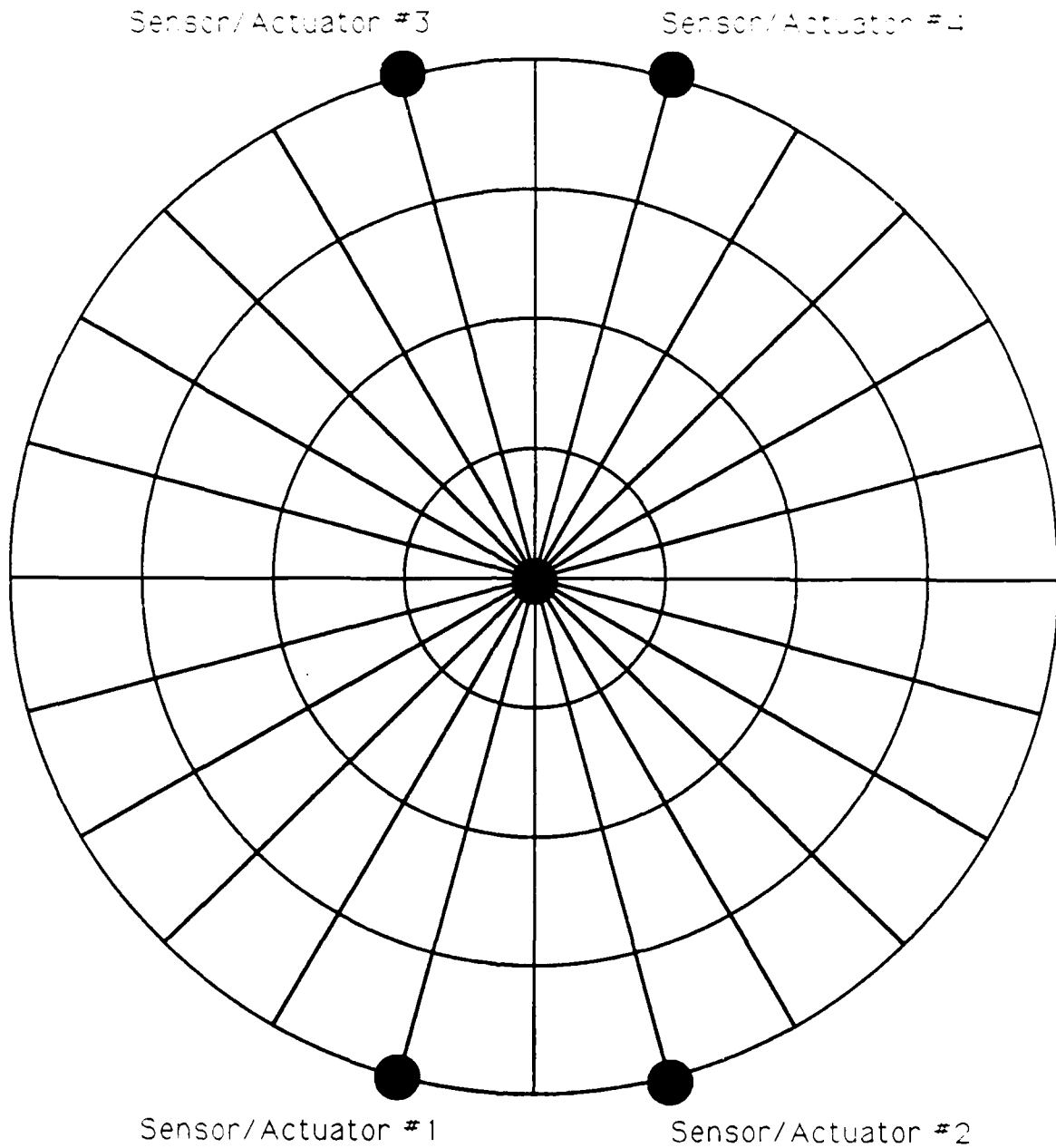
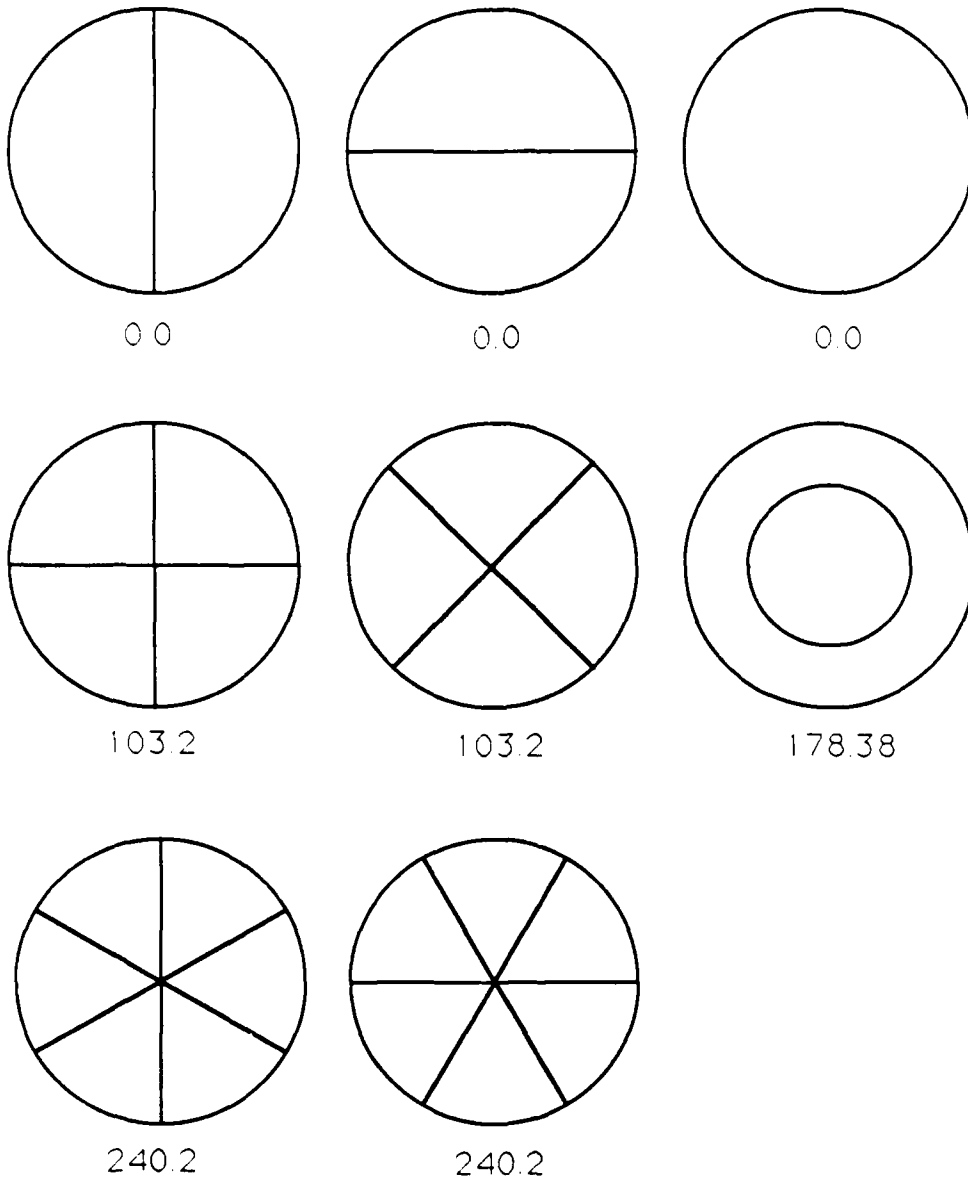
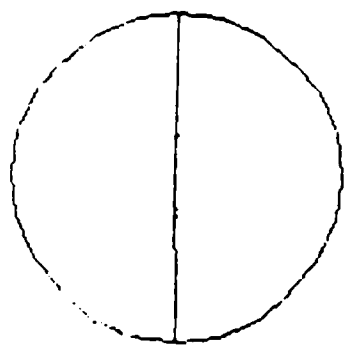


Figure 1- Circular Plate with Sensor/Actuator Masses  
at 15° from Y Axis

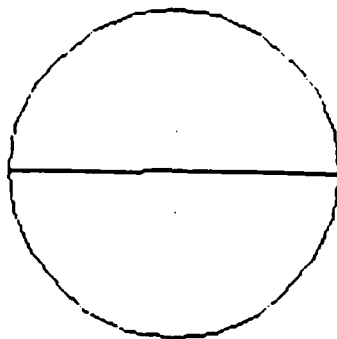


**Figure 2 - Mode Shapes For Uniform Circular Plate**

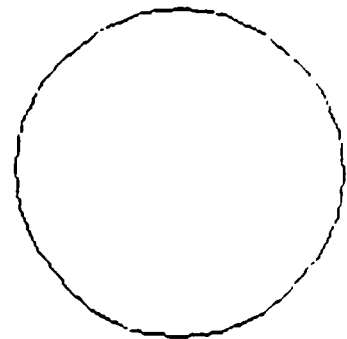
Frequencies in rad/sec



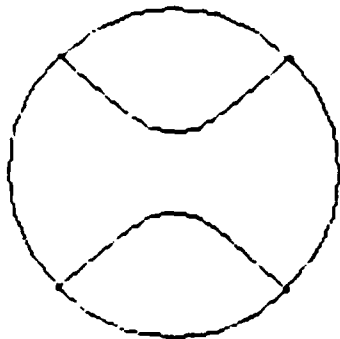
0.0



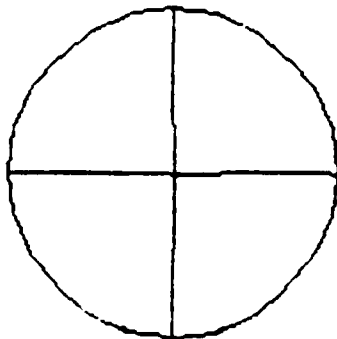
0.0



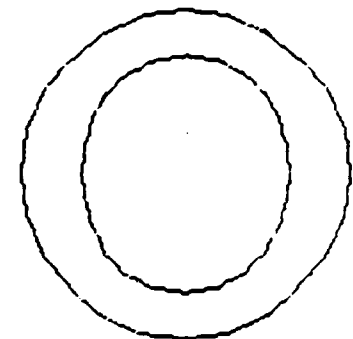
0.0



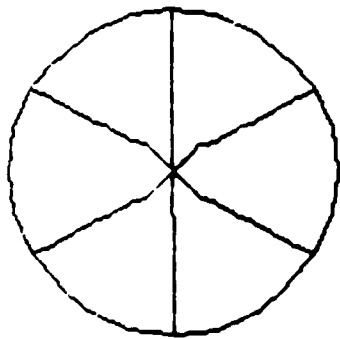
99.7



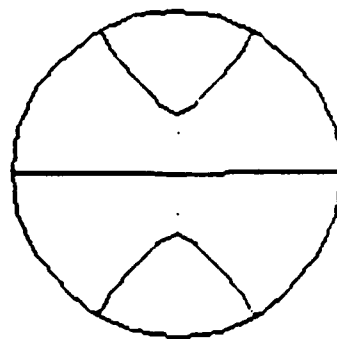
105.4



176.9



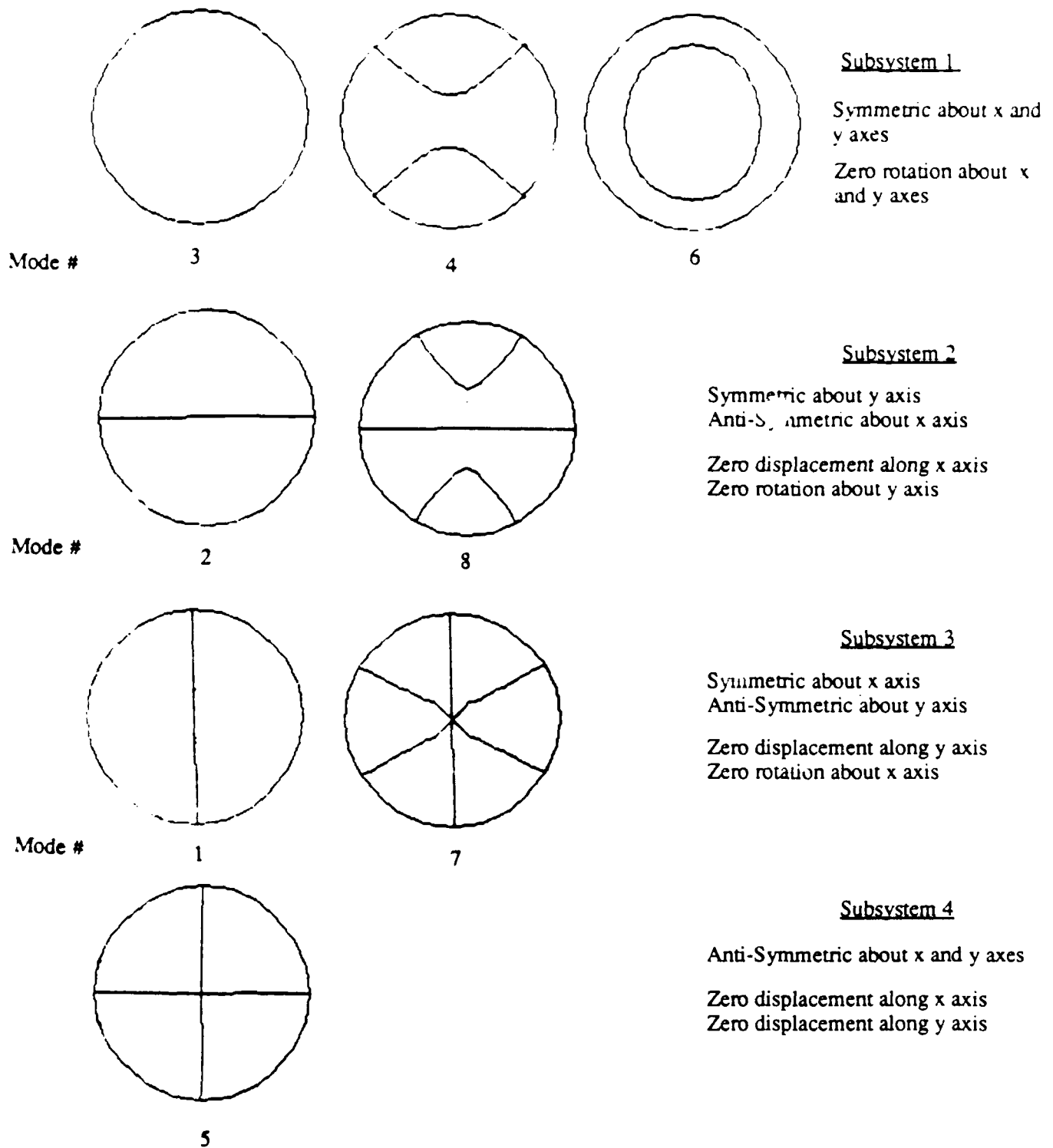
234.4



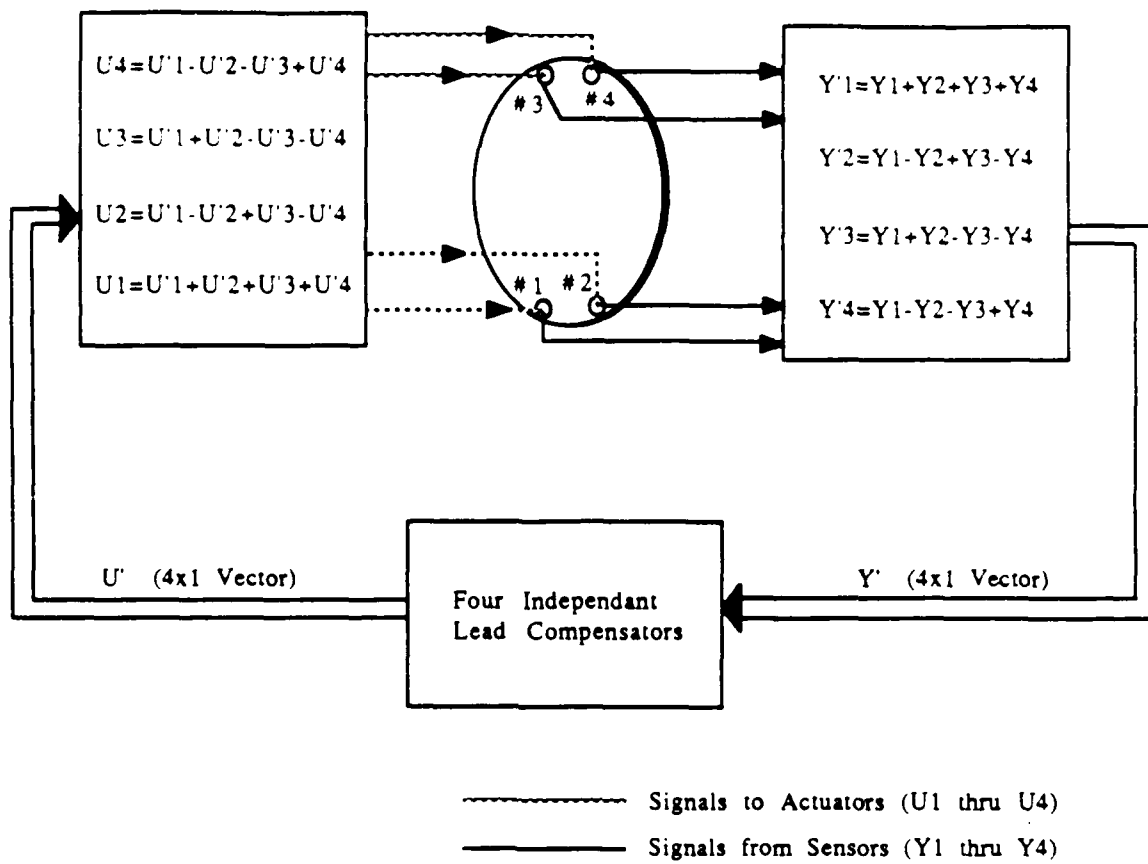
236.7

**Figure 3 - Mode Shapes for Plate with Masses at 15°**

Frequencies in rad/sec

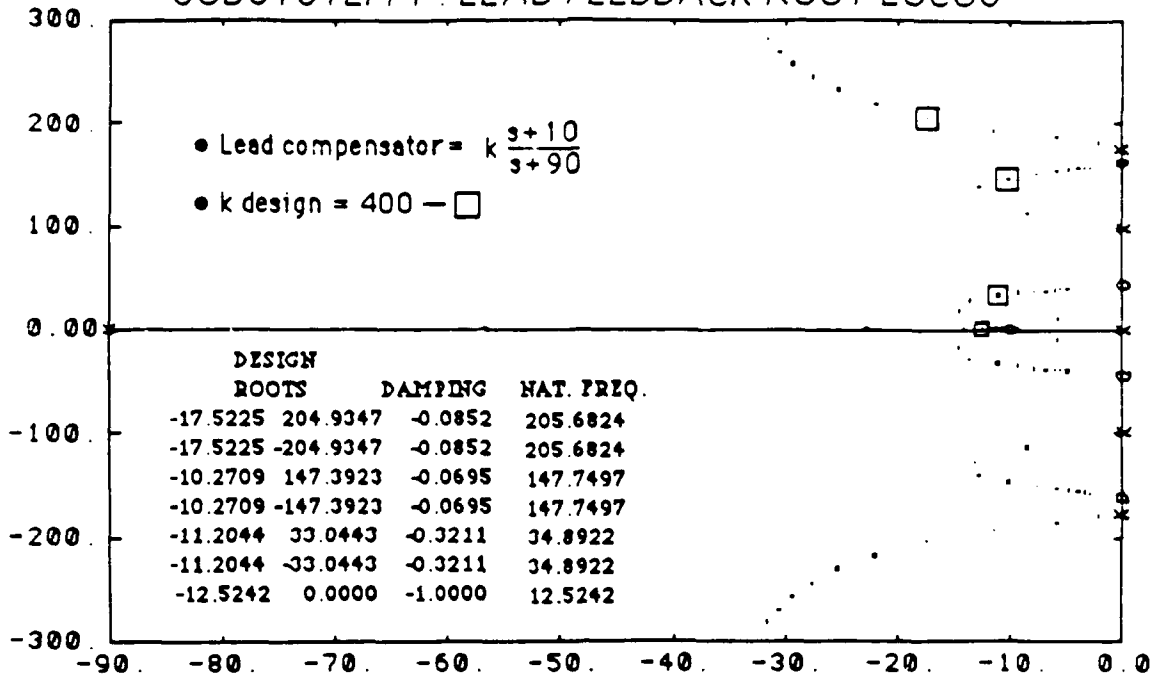


**Figure 4 - Subsystem Groupings and Symmetries**

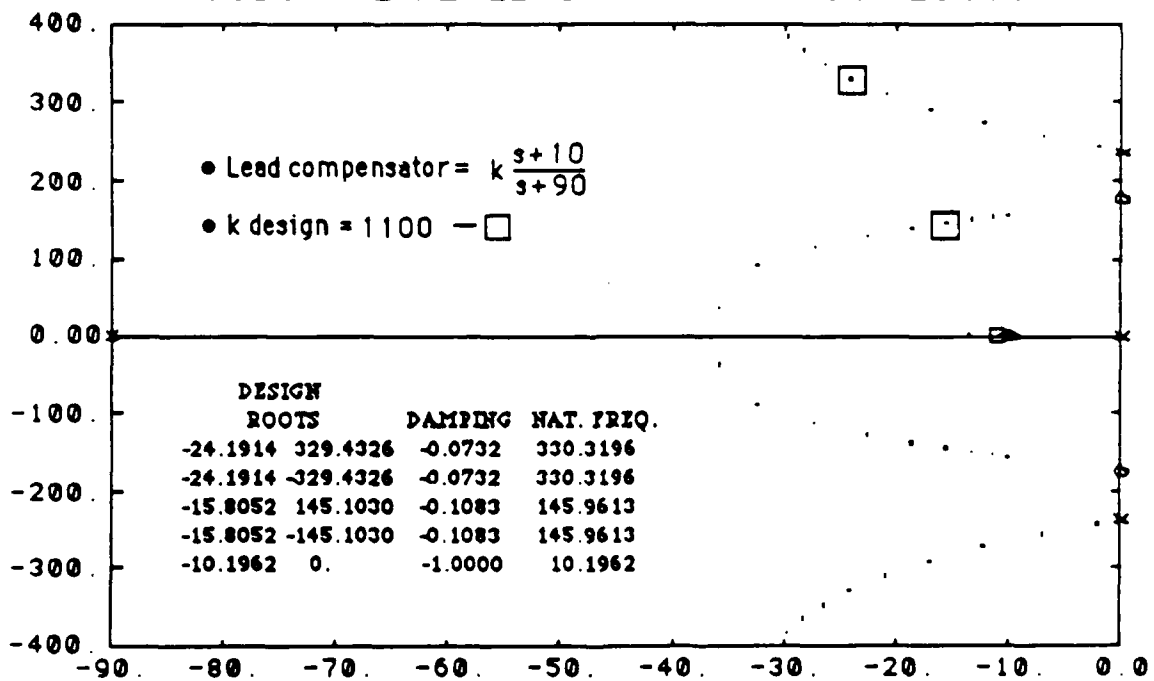


**Figure 5 - System Decoupling Schematic**

### SUBSYSTEM 1 : LEAD FEEDBACK ROOT LOCUS

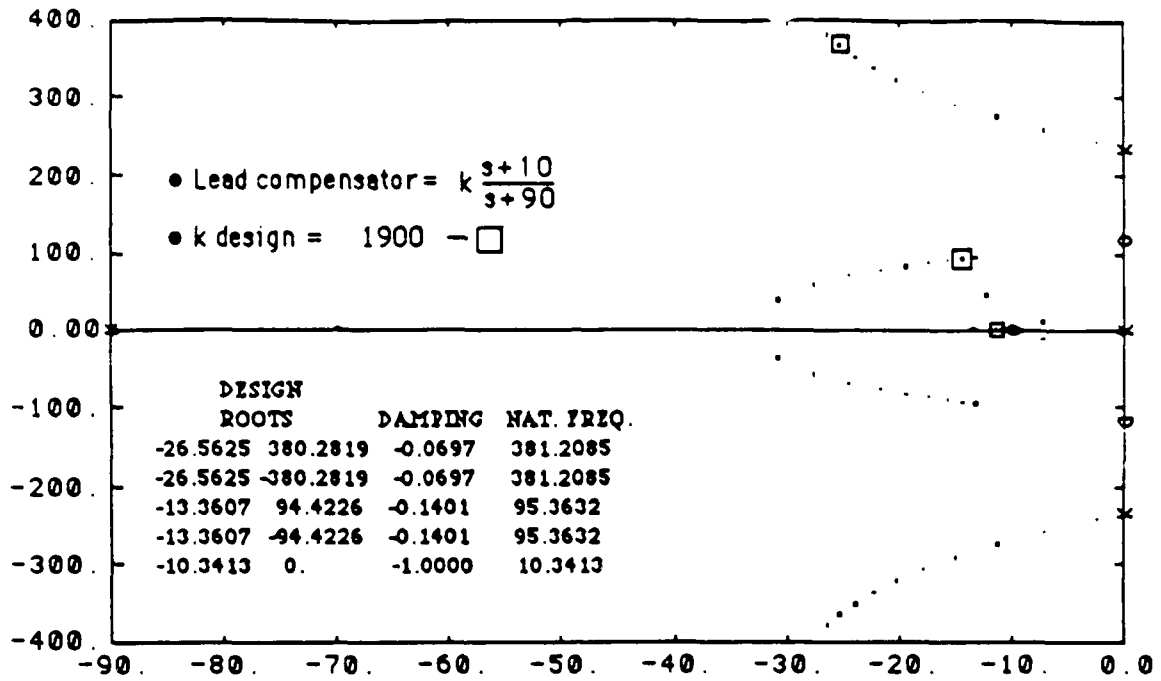


### SUBSYSTEM 2 : LEAD FEEDBACK ROOT LOCUS

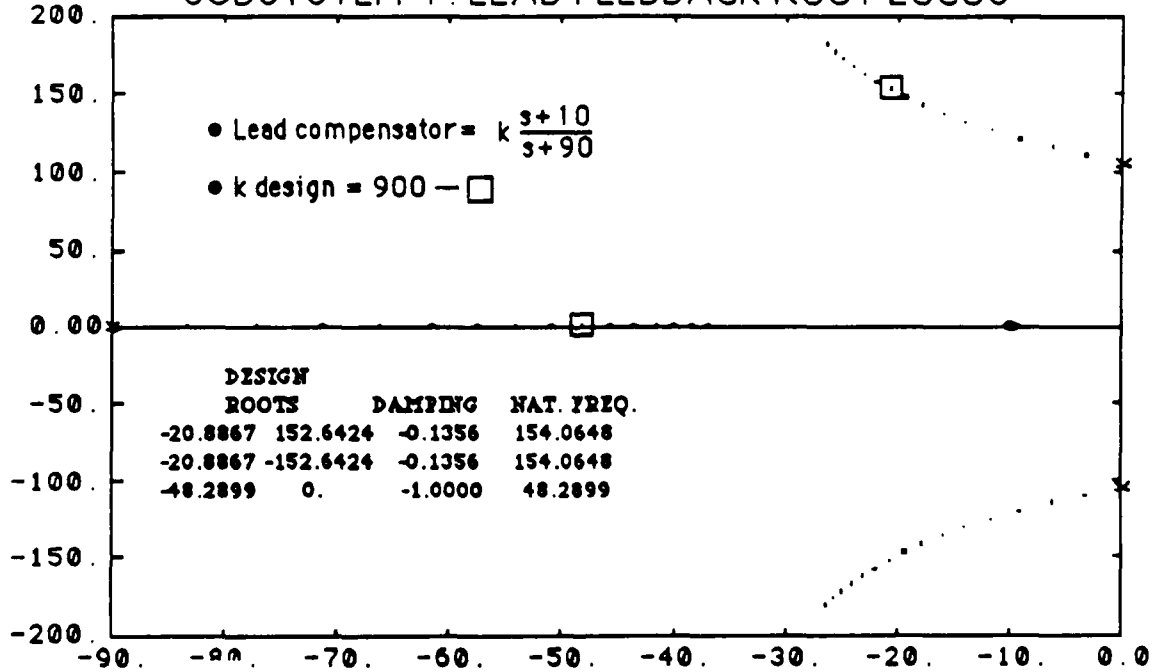


**Figures 6 and 7 Root loci for Subsystems 1 and 2**

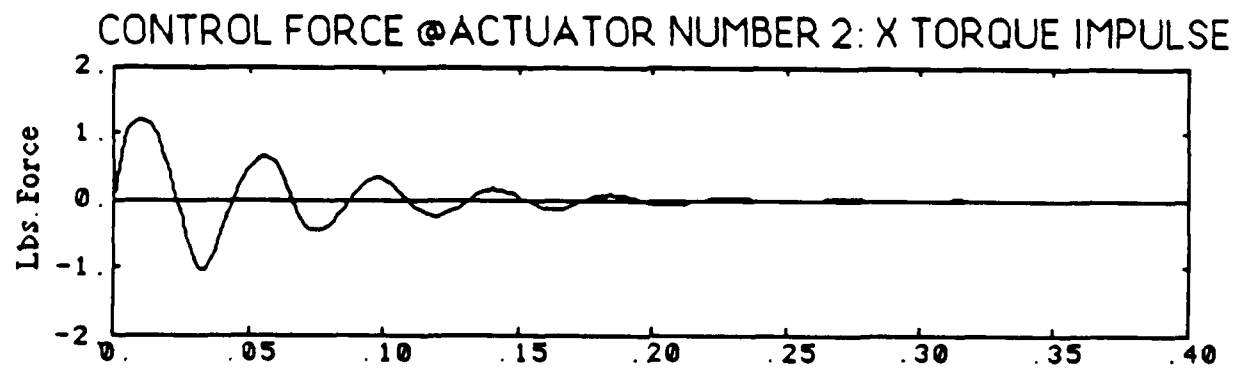
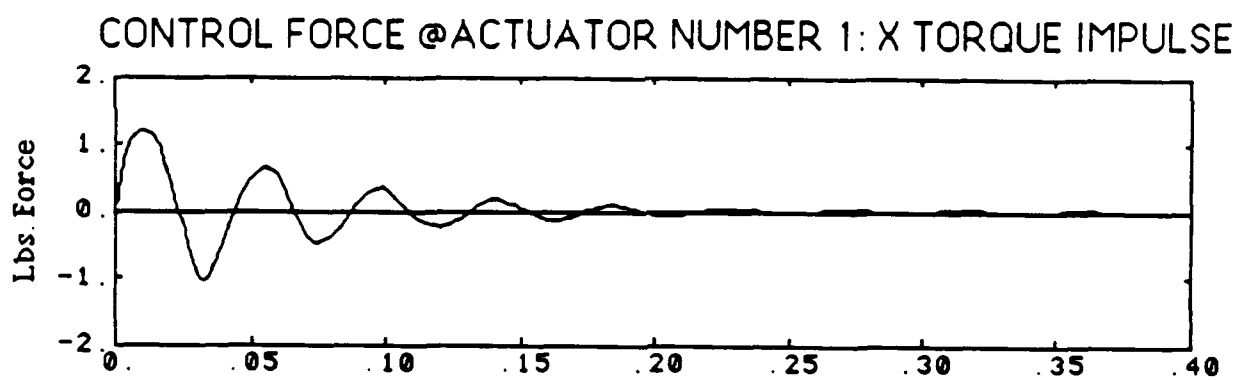
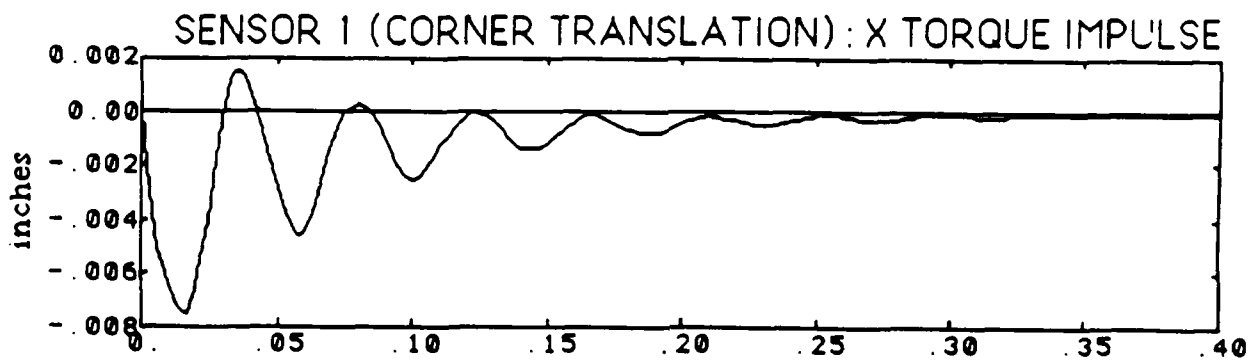
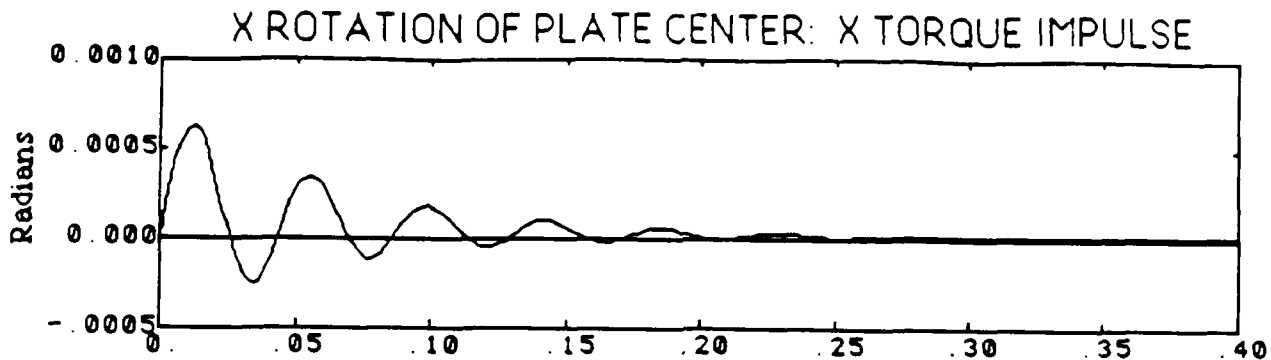
### SUBSYSTEM 3 : LEAD FEEDBACK ROOT LOCUS



### SUBSYSTEM 4 : LEAD FEEDBACK ROOT LOCUS

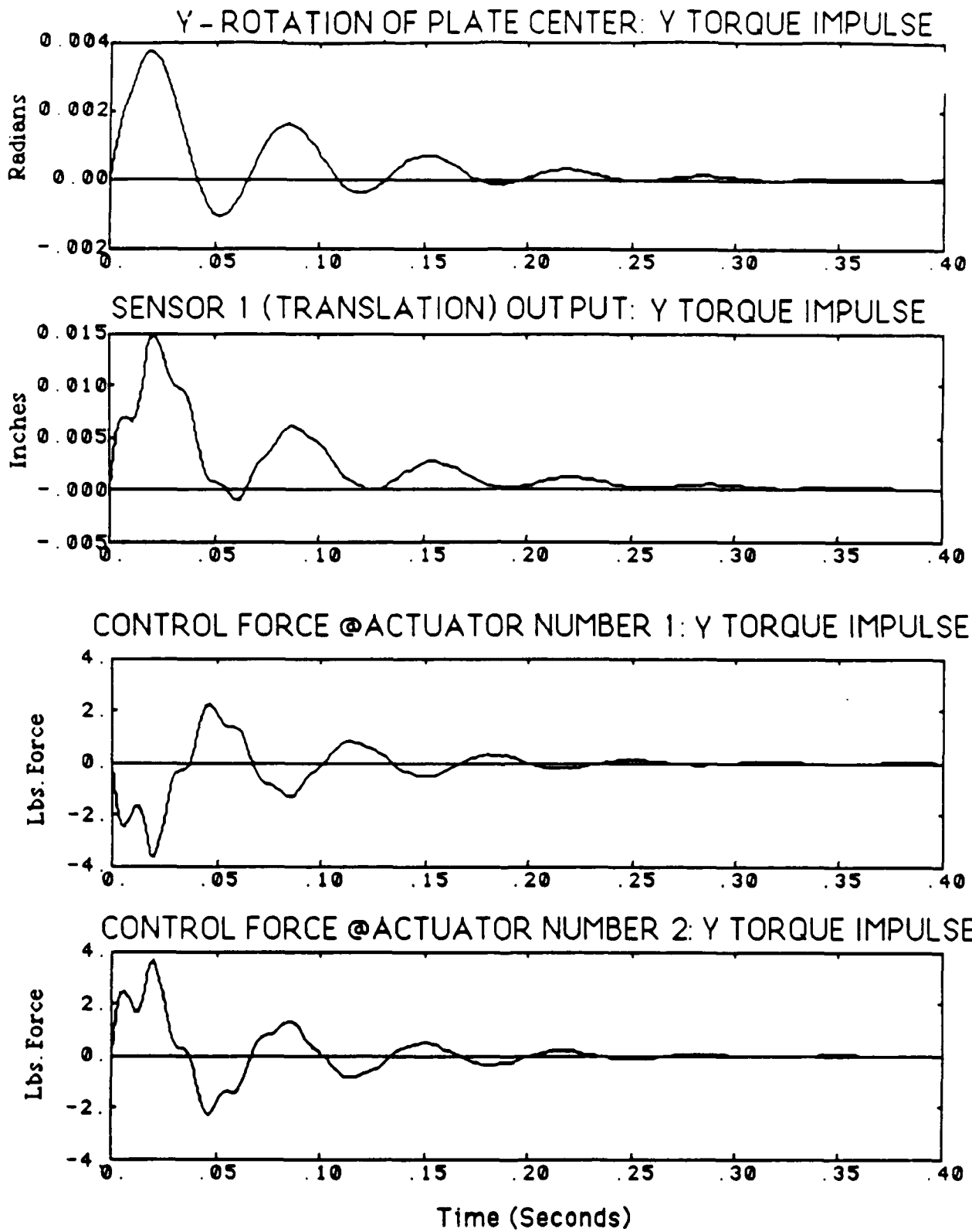


**Figures 8 and 9 Root loci for Subsystems 3 and 4**



Time (Seconds)

**Figure 10 (a) thru (d) - X Torque Impulse Response**



**Figure 11 (a) thru (d) - Y Torque Impulse Response**

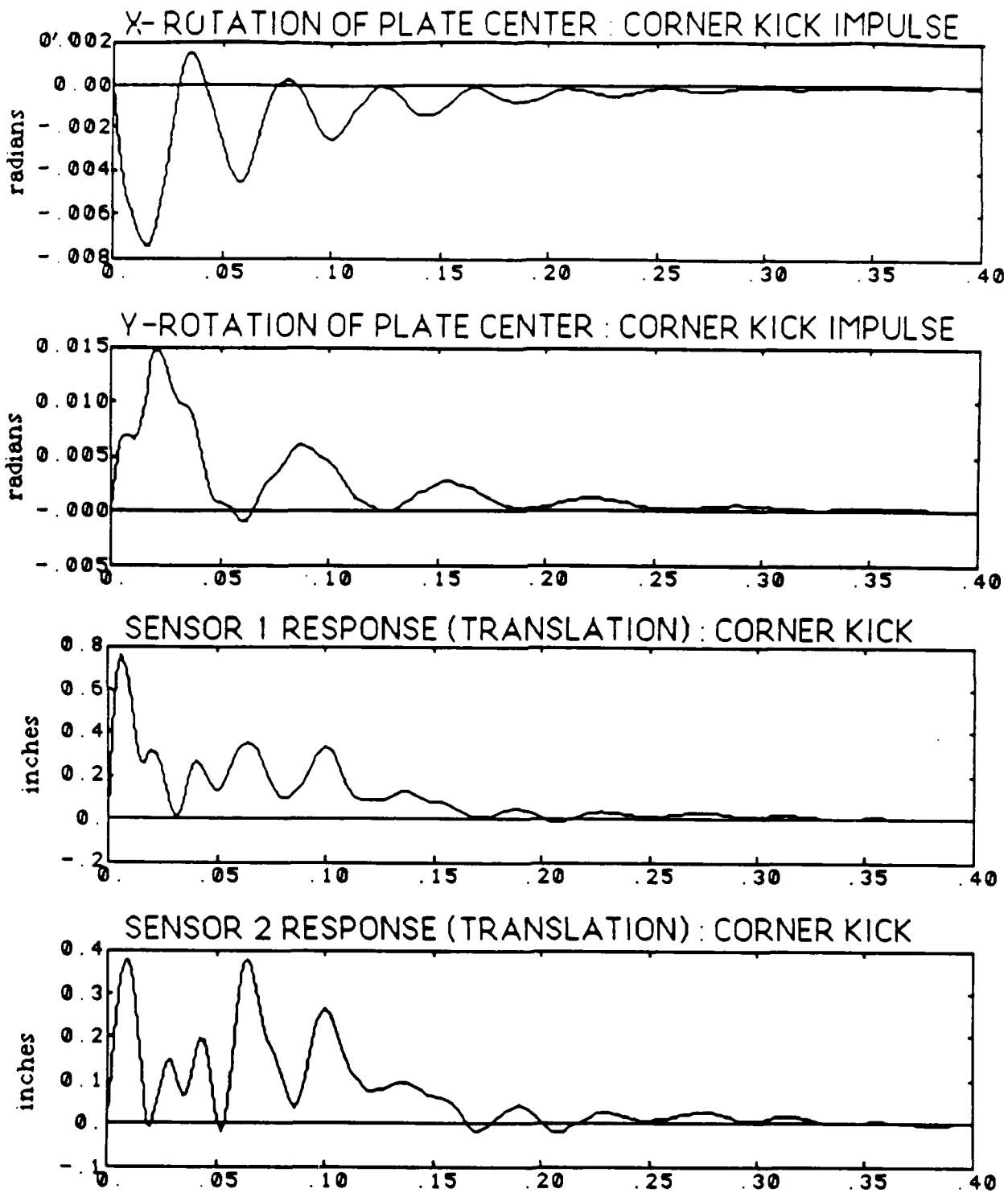
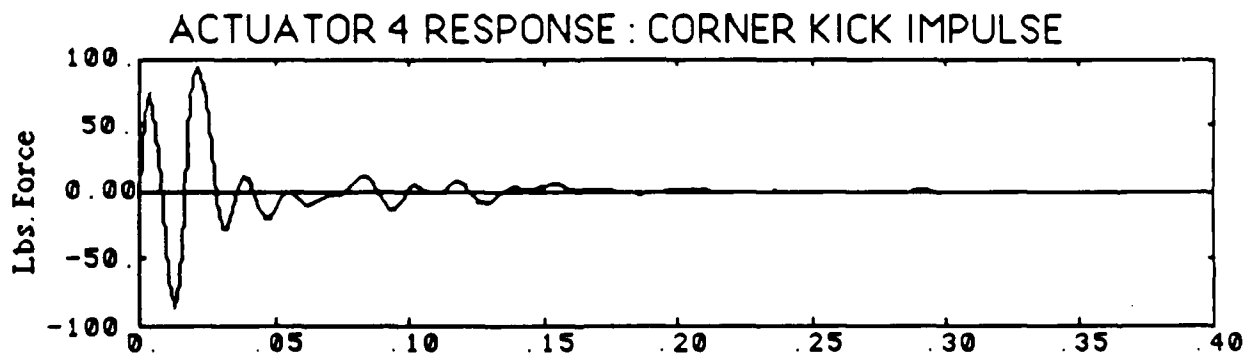
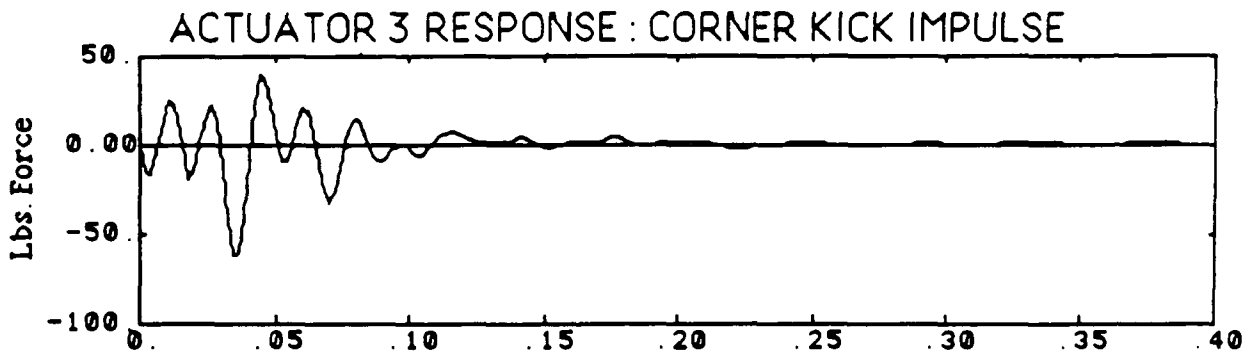
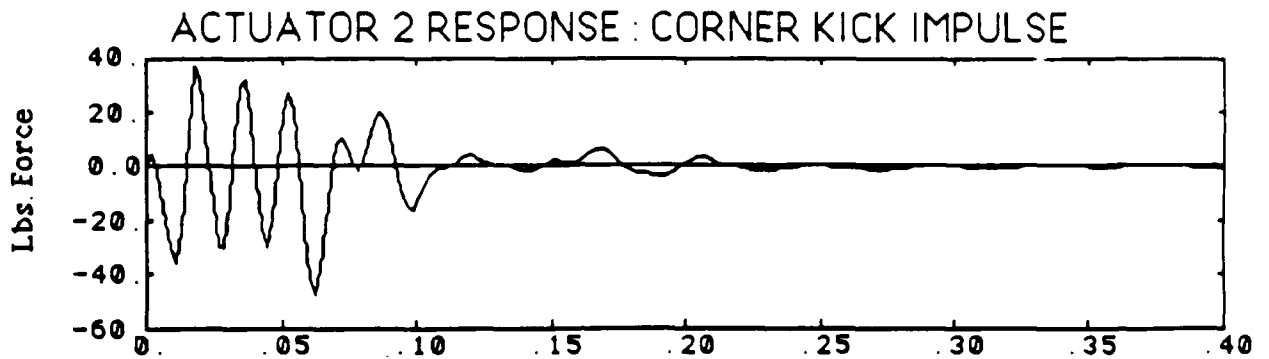
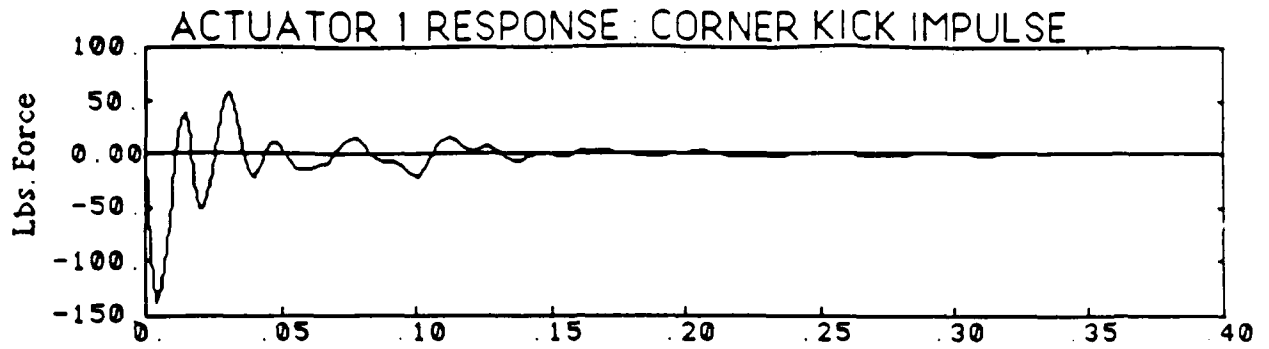


Figure 12 (a) thru (d) - <sup>Time (seconds)</sup> Corner Kick Response



**Figure 12 (e) thru (h) - Corner Kick Response**

F = Block Diagonal with i'th sub-block being

$$\begin{bmatrix} -2\rho_i\omega_i & -\omega_i^2 \\ 1 & 0 \end{bmatrix}$$

where  $\omega_i$  and  $\rho_i$  are given by

i	$\omega_i$	$\rho_i$
1	0	0
2	0	0
3	0	0
4	99.71	3e-4
5	105.4	3e-4
6	176.9	1.3e-3
7	234.7	4e-4
8	236.9	4e-4

G =

239.8000	-238.8000	237.9000	-240.8000
0.	0.	0.	0.
-188.5000	-188.7000	188.6000	188.4000
0.	0.	0.	0.
111.4000	111.2000	111.6000	111.5000
0.	0.	0.	0.
202.2000	202.2000	202.2000	202.2000
0.	0.	0.	0.
133.1000	-133.1000	-133.1000	133.1000
0.	0.	0.	0.
150.3000	150.3000	150.3000	150.3000
0.	0.	0.	0.
208.0000	-208.0000	208.0000	-208.0000
0.	0.	0.	0.
186.2000	186.2000	-186.2000	-186.2000
0.	0.	0.	0.

H transpose =

0.	0.	0.	0.
0.0547	-0.0544	0.0542	-0.0549
0.	0.	0.	0.
-0.2036	-0.2038	0.2036	0.2035
0.	0.	0.	0.
0.1015	0.1014	0.1017	0.1016
0.	0.	0.	0.
0.2065	0.2065	0.2065	0.2065
0.	0.	0.	0.
0.1222	-0.1222	-0.1222	0.1222
0.	0.	0.	0.
0.0962	0.0962	0.0962	0.0962
0.	0.	0.	0.
0.1942	-0.1942	0.1942	-0.1942
0.	0.	0.	0.
0.1679	0.1679	-0.1679	-0.1679

**Table 1 - System F,G and H Matrices**

## APPENDIX

### TRANSGEN User Manual

#### Introduction

Given the modal output of an undamped vibratory system from MSC PAL, TRANSGEN will calculate transfer functions for multi-input multi-output (MIMO) systems, the corresponding pole/residue form, and the state space form of the system equations for any selected input and output locations. The required PAL data are the modal displacements from a normal modes analysis and the modal masses from a transient response analysis (unfortunately, the PAL normal modes analysis does not have an option to print out the modal masses). The required user input is simply the number of nodes in the PAL model, the number of modes to include in calculating the desired outputs, the number of actuators and sensors, and their locations and orientations. The user also has the option of printing out any combination of output forms (i.e. pole-residue, pole-zero, and/or state space models).

Listed below are instructions for using TRANSGEN, the relevant theory behind the algorithms used in TRANSGEN, an example problem, and the FORTRAN program listing.

#### Producing PAL Modal Data

After building the PAL model file, both normal modes analysis and transient analysis (under the PAL dynamics menu) must be run in order to produce the modal displacements and mass files.

When running the normal modes analysis, the user must specify "jacobi" in the input file if the structure has any possible rigid body motions. The user may also specify how many rigid body modes will exist in this file by including the line "RIGID-BODY MODES n" where n is the number of rigid body modes expected. The addition of this line causes the frequencies of the first n modes to be exactly zero (instead of a very small number) and has little effect on the mode shapes. When prompted by PAL the user should choose the "print modal displacements" option from the output choices, and enter an appropriate file name. PAL will also ask for the number of modes to be calculated. Since dynamic analysis can have a maximum of 100 active degrees of freedom, it is best to choose less than 30 modes since mode shape accuracy degrades above that.

As mentioned above, the modal masses are also required for transfer function calculation but cannot be output from the normal modes analysis section of PAL. However, they can be obtained from transient analysis by using the following input file,

APPENDIX  
TRANSGEN User Manual

TITLE EXAMPLE  
JACOBI  
USE FIRST n MODES  
TIME RANGE 1,1,1  
PRINT MODAL EQUATIONS  
SOLVE

Here, n signifies how many modal masses are required to be output, and jACOBI has been selected to signify that rigid body modes will occur. It turns out that the number of modes used has no effect on the computation time. When the user is prompted for desired outputs he should not select any as they would then overwrite the modal equations. The modal equations are really just the elements of the diagonal modal mass and stiffness matrices. An output file name should still be entered. As the bulk of the time spent in performing a transient analysis is for recalculating the eigenvalues and eigenvectors (frequencies and mode shapes), this run will take as long as the normal modes analysis, but there is no alternative .

Running TRANSGEN

Once the modal displacement and mass files are obtained TRANSGEN can be run. The user has the option of inputting all information by hand or having it read from an input file, whose structure is shown below.

```
freebeam.mode  
freebeam.mass  
8,6  
2,2  
1,3  
6,3  
1,3  
6,3  
1,1,1
```

The first line contains the name of the modal displacements output file. The second line has the name of the modal mass file. On the third line is the number of nodes in the structure and the number of modes desired to be used in the subsequent calculations. The number of modes must obviously be less than or equal to the number of modes for which the user has obtained modal displacements and masses. On the next line is the number of actuators and the number of sensors, each of which must be  $\leq 6$  (this limit can be increased by the user by altering the parameter "ma" in the fortran source file). The following lines contain information about each actuator and sensor starting with actuator number 1 and ending with the last sensor. Each line contains two numbers; the node at which the sensor or actuator is located, and a number between 1 and 6 specifying the degree of freedom in which it is "sensitive". The number is chosen according to the following scheme,

APPENDIX  
TRANSGEN User Manual

- 1= x translation
- 2= y translation
- 3= z translation
- 4= x rotation
- 5= y rotation
- 6= z rotation.

The last line contains three switches where 1=on, 0=off, for each of the three output options, namely, residues output, transfer function zeros output, and state space form output.

TRANSGEN Output

The first section of the TRANSGEN output contains the zeros of the transfer function numerator polynomial matrix (see the section on theory below for more details about this). This matrix is size  $n_o \times n_i$  where  $n_o$ = number of sensors and  $n_i$ = number of actuators. Its form is as seen below, where  $n_{11}(s)$  is the numerator of the transfer function from actuator 1 to sensor 1,  $n_{12}(s)$  is the numerator of the transfer function from actuator 2 to sensor 1, and so on. All transfer functions have the same denominator  $d(s)$ .

$$\begin{bmatrix} y_1(s) \\ y_2(s) \\ \vdots \\ y_{n_o}(s) \end{bmatrix} = \begin{bmatrix} n_{11}(s) & n_{12}(s) & \dots \\ n_{21}(s) & n_{22}(s) & \dots \\ \vdots & & \\ \hline & & d(s) \end{bmatrix} \begin{bmatrix} u_1(s) \\ u_2(s) \\ \vdots \\ u_{n_i}(s) \end{bmatrix}$$

$$d(s) = \prod_i (s^2 + \omega_i^2)$$

$\omega_i$ =pole frequencies = normal mode frequencies

The real and imaginary parts of each root are printed since if sensor and actuator are separated (non-collocated) then zeros may exist which are not on the imaginary axis in the s-plane (s is the Laplace transform variable).

In all collocated cases all the zeros will lie on the imaginary axis and thus should have real part equal to zero. Due to numerical inaccuracies and truncation effects TRANSGEN will yield zeros with non-zero real parts, but they will be small compared to the imaginary part and can be ignored.

For non-collocated cases the user must exercise his own judgement for neglecting the real part of certain zeros since they may sometimes lie on the real axis. If a zero is found which has an imaginary part within 1 percent of a pole (normal mode) frequency the message "possible pole zero

APPENDIX  
TRANSGEN User Manual

cancellation" will appear following the zero's imaginary part. The user can then examine the pole frequencies to determine if he wishes to actually consider them cancelled.

The next section contains a list of the pole frequencies, or roots of  $d(s)$  (these are just the normal mode frequencies).

The third section contains residue matrices for the first  $n$  modes where  $n$  is the number of modes the user wanted included in the calculations. Each residue matrix is size  $n_o \times n_i$ .

The last section of the output contains the state space model  $G$  and  $H$  matrices for the state space equations,

$$\ddot{\eta} + \omega_D^2 \eta = G_m u$$

$$y = H_m \eta$$

with  $\eta$  = vector of modal deflection amplitudes. The dimension of  $G_m$  is  $n_o \times n_i$ , and the dimension of  $H_m$  is  $n_o \times n$ . The  $\omega_D^2$  matrix is the diagonal matrix of modal frequencies squared. The user should note that these are second order equations which can be transformed into first order equations by the usual methods.

APPENDIX  
TRANSGEN User Manual

Theory

In this section the algorithms for calculating the MIMO residues, transfer functions and pole-zero form from the system eigenvalues and eigenvectors (mode frequencies and shapes) are explained. These algorithms are explained in more depth by Chu and Plescia [A1].

The "residues" of a residue-pole form of a transfer function are the numerators of a partial fraction expansion of the pole-zero form. However, the residues can be calculated without first forming the pole-zero form as follows;

The system equations from a finite element analysis can be expressed as a set of coupled second-order equations:

$$\mathbf{M} \ddot{\mathbf{x}} + \mathbf{K} \mathbf{x} = \mathbf{G} \mathbf{u} ,$$

$\mathbf{x}$  =  $6n \times 1$  nodal displacements vector,  
 $n$  = number of nodes in structure,  
 $\mathbf{M}$  = mass matrix (square),  
 $\mathbf{K}$  = stiffness matrix (square),  
 $\mathbf{u}$  =  $n_i \times 1$  control input vector,  
 $\mathbf{G}$  = input distribution matrix,

Then, using the PAL generated eigenvector matrix,  $\mathbf{P}$ , the system is diagonalized to,

$$\mathbf{M}_D \ddot{\boldsymbol{\eta}} + \mathbf{K}_D \boldsymbol{\eta} = \mathbf{P}^T \mathbf{G} \mathbf{u} ,$$

using the transformation,

$$\mathbf{x} = \mathbf{P} \boldsymbol{\eta} ,$$

with

$$\mathbf{M}_D = \mathbf{P}^T \mathbf{M} \mathbf{P} ,$$

$$\mathbf{K}_D = \mathbf{P}^T \mathbf{K} \mathbf{P} , \quad \mathbf{M}_D, \mathbf{K}_D \text{ diagonal} ,$$

$\mathbf{M}_D, \mathbf{K}_D$  are modal mass and stiffness matrices

formed in PAL "transient response analysis",

$\boldsymbol{\eta}$  =  $m \times 1$  modal amplitudes vector ,

$m$  = number of modes user selects to be used .

Then with

$$\mathbf{y} = \mathbf{H} \mathbf{x} = \mathbf{H} \mathbf{P} \boldsymbol{\eta} ,$$

it is simple to form the transfer function matrix,

APPENDIX  
TRANSGEN User Manual

$$y(s) = \mathbf{H} \sum_{i=1}^m \left( \frac{\mathbf{P}_i \mathbf{P}_i^T}{m_i (s^2 + \omega_i^2)} \right) \mathbf{G} u(s),$$

where  $\mathbf{P}_i$  is the  $i$ 'th column of  $\mathbf{P}$  ( $i$ 'th mode shape),  $m_i$  is the  $i$ 'th modal mass (the  $i$ th diagonal member of the  $\mathbf{M}_D$  matrix), and  $\omega_i$  is the  $i$ 'th modal frequency =  $\sqrt{k_i/m_i}$ . Since  $\mathbf{H}$  and  $\mathbf{G}$  are constants they can be taken inside the summation to greatly reduce the size of the  $\mathbf{P}_i \mathbf{P}_i^T$  matrix since there will, in general, be fewer inputs and outputs than the number of modes ( $m$ ). The term  $\mathbf{H} \mathbf{P}_i \mathbf{P}_i^T \mathbf{G} / m_i$  is then the residue matrix for the  $i$ 'th mode (hereafter called  $\mathbf{R}_i$ ) whose frequency is  $\omega_i$ . Thus

$$y(s) = \sum_{i=1}^m \frac{\mathbf{R}_i}{(s^2 + \omega_i^2)} u(s).$$

Although  $\mathbf{P}$  is a large matrix (size  $6n \times m$ ), the computation of  $\mathbf{R}_i$  is not as difficult as it seems since the number of inputs (actuators) and outputs (sensors) is much smaller than the number of nodes in the structure so the  $\mathbf{G}$  and  $\mathbf{H}$  matrices will be sparse (i.e. have lots of zeros). This greatly reduces the task of multiplying  $\mathbf{H} \mathbf{P}_i$  (and  $\mathbf{P}_i^T \mathbf{G}$ ) since we have only to select the components of eigenvector  $i$  corresponding to the degrees of freedom in which the sensors are sensitive (or the actuators provide force or torque). Note,  $\mathbf{R}_i$  will be a matrix of dimension  $n_i \times n_o$  where " $n_i$ " is the number of inputs and " $n_o$ " is the number of outputs.

To obtain the pole-zero form, the  $m$  terms in the pole-residue form above are multiplied out to form a ratio of polynomials in  $s$  which has denominator of order  $2m$  and numerator of order  $2(m-1)$ . The numerator polynomial can then be factored to yield the zeros. The poles are already known as the natural frequencies and thus the denominator does not have to be factored.

A state space realization of the modal equations can be easily assembled using the following procedure. The matrix equation for the system can be rewritten as  $n$  scalar equations

$$\ddot{\eta}_i + \omega_i^2 \eta_i = \mathbf{P}_i^T \mathbf{G} u / m_i, \quad i=1, \dots, n.$$

The product of  $\mathbf{P}_i^T$  and  $\mathbf{G}$  is found as explained above. Defining  $\mathbf{P}_i^T \mathbf{G} / m_i = \mathbf{G}_{m_i}$  we can then transform this second order system to the first order system,

APPENDIX  
TRANSGEN User Manual

$$\dot{\eta} = \mathbf{F}\eta + \mathbf{G}_m u ,$$
$$\eta = [\eta_1, \dot{\eta}_1, \dots, \eta_n, \dot{\eta}_n]^T ,$$

with the even rows of  $\mathbf{G}_m$  being composed of  $\mathbf{G}_{m_i}$  and the odd rows of  $\mathbf{G}_m$  being rows of zeros.  $\mathbf{F}$  is a block diagonal matrix with the  $i$ 'th sub block

$$\mathbf{F}_i = \begin{bmatrix} 0 & 1 \\ -\omega_i^2 & 0 \end{bmatrix} .$$

The output distribution matrix for the state space realization is

$$y = \mathbf{H}_m \eta$$
$$\mathbf{H}_m = \mathbf{H}\mathbf{P}$$

The calculation of  $\mathbf{H}_m$  is similar to the calculation of  $\mathbf{G}_m$  but has, in general, even columns composed of all zeros since the modal displacements, and not velocities, are usually output. TRANSGEN outputs the second order state space form since this requires less space, so the user must manually perform the insertion of rows or columns of zeros in the  $\mathbf{G}_m$  and  $\mathbf{H}_m$  matrices and the expansion of the  $\mathbf{F}_i$  sub-blocks. This is easily done using common matrix manipulation programs.

APPENDIX  
TRANSGEN User Manual

Examples

Example 1. Free-Free Beam

The first example shows how TRANSGEN would be used to find the different transfer function forms for a continuous uniform beam with a force actuator perpendicular to the beam at one end and four sensors as shown below in figure A1. The beam is made of Aluminum, has a length of 10 feet and square cross section of side length 1 inch. Sensors 1 and 3 are translation sensors while 2 and 4 are rotation sensors. The PAL input file in Table A1 shows the nodal geometry and which degrees of freedom are made active, namely z translation and y rotation.

```
TITLE NINE POINT BEAM
NODAL POINT LOCATIONS 1
1 0 0 0 THROUGH 9 120 0 0
10 0 0 1
11 120 0 1
-
MATERIAL PROPERTIES 1E7 0 2.6E-4 .3
C ALUMINUM BEAM OF CROSS SECTION 1 X 1 INCH
BEAM TYPE 2 1 1
DO CONNECT 1 2 THROUGH 8 9 STEP 1 1
SPRING 1E-7
CO 1 TO 10
CO 9 TO 11
ZERO
TY ALL
TX ALL
RZ ALL
RX ALL
ALL 10,11
-
ACTIVATE
TZ ALL
RY ALL
-
END DEFINITION
```

Table A1. PAL Input File

The reader may note that there are extra springs inserted at the ends of the beam perpendicular to it's length. These very small springs were used to force PAL to obtain rigid body mode shapes which corresponded to pure z-translation and pure y-rotation about the center of the bar, and were small enough so that they would have only negligible effects on the flexible mode shapes and frequencies. Without the springs PAL returns rigid body mode shapes which are both y-rotations about points other than the center of mass of the beam.

APPENDIX  
TRANSGEN User Manual

Tables A3 and A4 show the command files for the Normal Modes and Transient Analysis sections of PAL dynamic analysis which were used to obtain the mode shape and modal mass files seen in the, Tables A6 and A7.

```
TITLE FREEBEAM.MASS
USE FIRST 10 MODES
JACOBI
TIME RANGE 1,1,1
PRINT MODAL EQUATIONS
SOLVE
```

Table A2. PAL Normal Modes Analysis Command File

```
TITLE FREEBEAM.DYN
JACOBI
RIGID-BODY MODES 2
SOLVE
```

Table A3. PAL Transient Response Analysis Command File

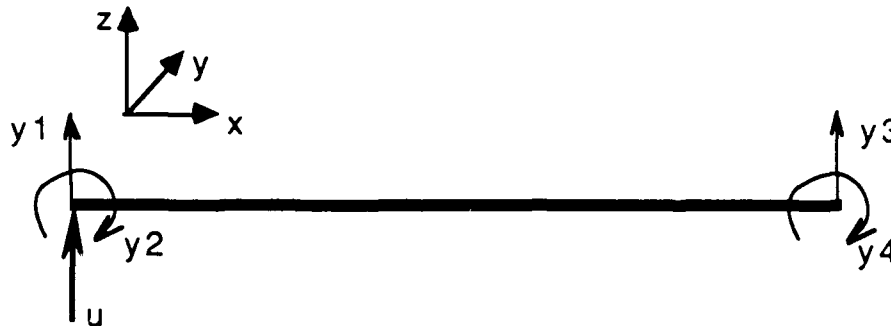


Figure A1. Uniform Beam Geometry and Sensor/Actuator Placement

Table A4 below shows the input file for TRANSGEN which will use the first eight modes (2 rigid + 6 flexible) to calculate the desired transfer functions and state space form of the modal equations. Note the correspondence of the sensor locations specified in the file with those pictured in figure A1. Also note that even though there are only nine nodes on the beam itself, 11 nodes are specified in line three since there are two extra nodes (rigid supports for the small springs). This file will call for all three types of output to be printed in the output file which the user is asked to identify during the running of TRANSGEN.

APPENDIX  
TRANSGEN User Manual

freebeam.mode  
 freebeam.mass  
 11,8  
 1,4  
 1,3  
 1,3  
 1,5  
 9,3  
 9,5  
 1,1,1

Table A4. TRANSGEN Input File

The resulting output of TRANSGEN is shown in Table A8. The first section of the output file contain an echo of the relevant input information. Next is a listing of the zeros, or roots, of the numerator polynomials for all the different sensor actuator combinations. We thus see that the transfer function from actuator 1 to sensor 1, which is a collocated combination, has the following form;

$$\frac{y_1(s)}{u_1(s)} = \frac{907.1s^2 (s^2 + 60.6^2) (s^2 + 196.8^2) (s^2 + 411.7^2) (s^2 + 707.5^2) (s^2 + 1088.3^2) \dots}{s^2 s^2 (s^2 + 87.94^2) (s^2 + 242.5^2) (s^2 + 475.9^2) (s^2 + 788^2) (s^2 + 1183^2) \dots}$$

There is a pole zero cancellation at the origin (the two zeros cancel two of the four poles at the origin). The poles and zeros then alternate with increasing frequency as expected for a collocated sensor-actuator pair. The transfer function from the input to output 2 (rotation sensor at left end of beam) is similar but with different zeros.

The transfer functions from the input to outputs at the other end of the beam show zeros which are either real or complex instead of purely imaginary, since they correspond to non-collocated sensor actuator pairs. The lower zeros, i.e. those at  $s = \pm 122.4$ ,  $s = \pm 307.3$  are quite close to the exact zeros (exact meaning roots of the exact transfer function for a simple beam) for this case as calculated by Wie [A2]. The higher complex zeros could be actual zeros or they could be the results of numerical truncation in the factoring of the numerator polynomial performed in TRANSGEN. In order to determine if they are actual zeros the user could either do another TRANSGEN run with more modes used in the calculation or he could take the state space description output from TRANSGEN and use a program like PC- MATLAB to form the transfer functions independently. In the end, it will most likely be concluded that some of these complex zeros are due to inaccuracies of the mode shapes as calculated by PAL, and that it may be best to ignore them as long as their frequencies appear to be much higher than the first few mode frequencies.

The following section shows the natural frequencies of the beam in units of radians/sec. These frequencies are really the imaginary parts of

APPENDIX  
TRANSGEN User Manual

the roots of the denominator of the transfer function, also known as poles. Each pole also has a complex conjugate which is not shown here, so for this example there will be four poles at the origin.

The next section of output data is the modal residues section. The residue for each input-output pair is shown for each mode. This is the data that is used to compute the zeros of the transfer functions as explained in the theory section above. The residues for actuator 1 to sensor 2 are combined to form the following equation which is then multiplied out and factored to get the zeros.

$$\frac{y_2}{u_1} = \frac{0}{s^2} + \frac{1.6}{s^2} + \frac{4.97}{s^2 + 88^2} + \frac{8.41}{s^2 + 242.5^2} + \frac{11.83}{s^2 + 475.9^2} + \frac{15.4}{s^2 + 788^2} + \dots$$

Note that the first mode has a residue of zero since the rigid translation mode will not contribute to the output of a rotation sensor at the end of the beam. The other residues are of the same order of magnitude here only by coincidence.

The last section of output data is the state space form of the modal equations. The **G** matrix is 8x1 since there is one input and eight modes, while the **H** matrix is 4x8 since there are four outputs. From the **G** matrix the user can note that all 8 modes are approximately equally controllable with a force actuator at one end of the beam. From **H** he can see that the first mode is unobservable with the rotation sensors (outputs 2 and 4) as expected since it is a rigid body translation. The **G** and **H** matrices can be used in a program like PC-MATLAB or similar control system design software after insertion of rows and columns of zeros as described in the theory section. Natural damping inherent to all structures can be inserted in the **F** matrix if the analyst already knows how much damping each mode contains.

A set of modal state space equations of smaller order than the set which is output can be formed either by running TRANSGEN again requesting a smaller number of modes be used, or by simply truncating the larger system so that only the desired number of modes are kept.

APPENDIX  
TRANSGEN User Manual

Table A.1. PAL Normal Modes Analysis Output

FREEBEAM.DYN

MODE SHAPE NO. 1 AT 0.00000E-01 CPS ( 0.00000E-01 RAD/SEC)

NODE	X TRANS	Y TRANS	Z TRANS	X ROT	Y ROT	Z ROT
1	0.0000E-01	0.0000E-01	3.3331E-01	0.0000E-01	-3.7939E-07	0.0000E-01
2	0.0000E-01	0.0000E-01	3.3332E-01	0.0000E-01	-3.7939E-07	0.0000E-01
3	0.0000E-01	0.0000E-01	3.3332E-01	0.0000E-01	-3.7938E-07	0.0000E-01
4	0.0000E-01	0.0000E-01	3.3333E-01	0.0000E-01	-3.7936E-07	0.0000E-01
5	0.0000E-01	0.0000E-01	3.3333E-01	0.0000E-01	-3.7934E-07	0.0000E-01
6	0.0000E-01	0.0000E-01	3.3334E-01	0.0000E-01	-3.7932E-07	0.0000E-01
7	0.0000E-01	0.0000E-01	3.3334E-01	0.0000E-01	-3.7931E-07	0.0000E-01
8	0.0000E-01	0.0000E-01	3.3335E-01	0.0000E-01	-3.7930E-07	0.0000E-01
9	0.0000E-01	0.0000E-01	3.3336E-01	0.0000E-01	-3.7929E-07	0.0000E-01
10	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01
11	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01

MODE SHAPE NO. 2 AT 0.00000E-01 CPS ( 0.00000E-01 RAD/SEC)

NODE	X TRANS	Y TRANS	Z TRANS	X ROT	Y ROT	Z ROT
1	0.0000E-01	0.0000E-01	5.1624E-01	0.0000E-01	8.6038E-03	0.0000E-01
2	0.0000E-01	0.0000E-01	3.8718E-01	0.0000E-01	8.6038E-03	0.0000E-01
3	0.0000E-01	0.0000E-01	2.5812E-01	0.0000E-01	8.6038E-03	0.0000E-01
4	0.0000E-01	0.0000E-01	1.2907E-01	0.0000E-01	8.6038E-03	0.0000E-01
5	0.0000E-01	0.0000E-01	1.1751E-05	0.0000E-01	8.6038E-03	0.0000E-01
6	0.0000E-01	0.0000E-01	-1.2904E-01	0.0000E-01	8.6038E-03	0.0000E-01
7	0.0000E-01	0.0000E-01	-2.5810E-01	0.0000E-01	8.6038E-03	0.0000E-01
8	0.0000E-01	0.0000E-01	-3.8716E-01	0.0000E-01	8.6038E-03	0.0000E-01
9	0.0000E-01	0.0000E-01	-5.1621E-01	0.0000E-01	8.6038E-03	0.0000E-01
10	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01
11	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01

MODE SHAPE NO. 3 AT 1.39966E+01 CPS ( 8.79434E+01 RAD/SEC)

NODE	X TRANS	Y TRANS	Z TRANS	X ROT	Y ROT	Z ROT
1	0.0000E-01	0.0000E-01	5.5883E-01	0.0000E-01	2.1645E-02	0.0000E-01
2	0.0000E-01	0.0000E-01	2.3668E-01	0.0000E-01	2.0994E-02	0.0000E-01
3	0.0000E-01	0.0000E-01	-5.5472E-02	0.0000E-01	1.7322E-02	0.0000E-01
4	0.0000E-01	0.0000E-01	-2.6384E-01	0.0000E-01	9.8905E-03	0.0000E-01
5	0.0000E-01	0.0000E-01	-3.3969E-01	0.0000E-01	-1.0394E-15	0.0000E-01
6	0.0000E-01	0.0000E-01	-2.6384E-01	0.0000E-01	-9.8905E-03	0.0000E-01
7	0.0000E-01	0.0000E-01	-5.5472E-02	0.0000E-01	-1.7322E-02	0.0000E-01
8	0.0000E-01	0.0000E-01	2.3668E-01	0.0000E-01	-2.0994E-02	0.0000E-01
9	0.0000E-01	0.0000E-01	5.5883E-01	0.0000E-01	-2.1645E-02	0.0000E-01
10	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01
11	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01

APPENDIX  
TRANSGEN User Manual

MODE SHAPE NO. 4 AT 3.85917E+01 CPS ( 2.42479E+02 RAD/SEC)

NODE	X TRANS	Y TRANS	Z TRANS	X ROT	Y ROT	Z ROT
1	0.0000E-01	0.0000E-01	5.4647E-01	0.0000E-01	3.5812E-02	0.0000E-01
2	0.0000E-01	0.0000E-01	2.6410E-02	0.0000E-01	3.1536E-02	0.0000E-01
3	0.0000E-01	0.0000E-01	-3.1972E-01	0.0000E-01	1.2231E-02	0.0000E-01
4	0.0000E-01	0.0000E-01	-3.0911E-01	0.0000E-01	-1.2990E-02	0.0000E-01
5	0.0000E-01	0.0000E-01	5.2976E-15	0.0000E-01	-2.4597E-02	0.0000E-01
6	0.0000E-01	0.0000E-01	3.0911E-01	0.0000E-01	-1.2990E-02	0.0000E-01
7	0.0000E-01	0.0000E-01	3.1972E-01	0.0000E-01	1.2231E-02	0.0000E-01
8	0.0000E-01	0.0000E-01	-2.6410E-02	0.0000E-01	3.1536E-02	0.0000E-01
9	0.0000E-01	0.0000E-01	-5.4647E-01	0.0000E-01	3.5812E-02	0.0000E-01
10	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01
11	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01

MODE SHAPE NO. 5 AT 7.57386E+01 CPS ( 4.75880E+02 RAD/SEC)

NODE	X TRANS	Y TRANS	Z TRANS	X ROT	Y ROT	Z ROT
1	0.0000E-01	0.0000E-01	-5.3454E-01	0.0000E-01	-4.9055E-02	0.0000E-01
2	0.0000E-01	0.0000E-01	1.4295E-01	0.0000E-01	-3.4982E-02	0.0000E-01
3	0.0000E-01	0.0000E-01	3.3218E-01	0.0000E-01	1.1731E-02	0.0000E-01
4	0.0000E-01	0.0000E-01	-7.8635E-02	0.0000E-01	3.3610E-02	0.0000E-01
5	0.0000E-01	0.0000E-01	-3.8056E-01	0.0000E-01	0.0000E-01	0.0000E-01
6	0.0000E-01	0.0000E-01	-7.8635E-02	0.0000E-01	-3.3610E-02	0.0000E-01
7	0.0000E-01	0.0000E-01	3.3218E-01	0.0000E-01	-1.1731E-02	0.0000E-01
8	0.0000E-01	0.0000E-01	1.4295E-01	0.0000E-01	3.4982E-02	0.0000E-01
9	0.0000E-01	0.0000E-01	-5.3454E-01	0.0000E-01	4.9055E-02	0.0000E-01
10	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01
11	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01

MODE SHAPE NO. 6 AT 1.25510E+02 CPS ( 7.88600E+02 RAD/SEC)

NODE	X TRANS	Y TRANS	Z TRANS	X ROT	Y ROT	Z ROT
1	0.0000E-01	0.0000E-01	-5.2213E-01	0.0000E-01	-6.1748E-02	0.0000E-01
2	0.0000E-01	0.0000E-01	2.6383E-01	0.0000E-01	-2.9315E-02	0.0000E-01
3	0.0000E-01	0.0000E-01	1.3305E-01	0.0000E-01	3.9374E-02	0.0000E-01
4	0.0000E-01	0.0000E-01	-3.6447E-01	0.0000E-01	8.2770E-03	0.0000E-01
5	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	-4.3599E-02	0.0000E-01
6	0.0000E-01	0.0000E-01	3.6447E-01	0.0000E-01	8.2770E-03	0.0000E-01
7	0.0000E-01	0.0000E-01	-1.3305E-01	0.0000E-01	3.9374E-02	0.0000E-01
8	0.0000E-01	0.0000E-01	2.6383E-01	0.0000E-01	-2.9315E-02	0.0000E-01
9	0.0000E-01	0.0000E-01	5.2213E-01	0.0000E-01	-6.1748E-02	0.0000E-01
10	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01
11	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01

MODE SHAPE NO. 7 AT 1.88251E+02 CPS ( 1.18282E+03 RAD/SEC)

NODE	X TRANS	Y TRANS	Z TRANS	X ROT	Y ROT	Z ROT
------	---------	---------	---------	-------	-------	-------

APPENDIX  
TRANSGEN User Manual

1	0.0000E-01	0.0000E-01	-5.0768E-01	0.0000E-01	-7.3752E-02	0.0000E-01
2	0.0000E-01	0.0000E-01	3.2585E-01	0.0000E-01	-1.3803E-02	0.0000E-01
3	0.0000E-01	0.0000E-01	-1.4433E-01	0.0000E-01	4.7186E-02	0.0000E-01
4	0.0000E-01	0.0000E-01	-2.0030E-01	0.0000E-01	-4.3204E-02	0.0000E-01
5	0.0000E-01	0.0000E-01	3.6168E-01	0.0000E-01	0.0000E-01	0.0000E-01
6	0.0000E-01	0.0000E-01	-2.0030E-01	0.0000E-01	4.3204E-02	0.0000E-01
7	0.0000E-01	0.0000E-01	-1.4433E-01	0.0000E-01	-4.7186E-02	0.0000E-01
8	0.0000E-01	0.0000E-01	3.2585E-01	0.0000E-01	1.3803E-02	0.0000E-01
9	0.0000E-01	0.0000E-01	-5.0768E-01	0.0000E-01	7.3752E-02	0.0000E-01
10	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01
11	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01

MODE SHAPE NO. 8 AT 2.64102E+02 CPS ( 1.65940E+03 RAD/SEC)

NODE	X TRANS	Y TRANS	Z TRANS	X ROT	Y ROT	Z ROT
1	0.0000E-01	0.0000E-01	4.8558E-01	0.0000E-01	8.4162E-02	0.0000E-01
2	0.0000E-01	0.0000E-01	-3.2352E-01	0.0000E-01	-9.7318E-03	0.0000E-01
3	0.0000E-01	0.0000E-01	3.2950E-01	0.0000E-01	-2.0942E-02	0.0000E-01
4	0.0000E-01	0.0000E-01	-1.9839E-01	0.0000E-01	4.8086E-02	0.0000E-01
5	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	-5.8190E-02	0.0000E-01
6	0.0000E-01	0.0000E-01	1.9839E-01	0.0000E-01	4.8086E-02	0.0000E-01
7	0.0000E-01	0.0000E-01	-3.2950E-01	0.0000E-01	-2.0942E-02	0.0000E-01
8	0.0000E-01	0.0000E-01	3.2352E-01	0.0000E-01	-9.7318E-03	0.0000E-01
9	0.0000E-01	0.0000E-01	-4.8558E-01	0.0000E-01	8.4162E-02	0.0000E-01
10	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01
11	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01

MODE SHAPE NO. 9 AT 3.49371E+02 CPS ( 2.19516E+03 RAD/SEC)

NODE	X TRANS	Y TRANS	Z TRANS	X ROT	Y ROT	Z ROT
1	0.0000E-01	0.0000E-01	4.1905E-01	0.0000E-01	8.4679E-02	0.0000E-01
2	0.0000E-01	0.0000E-01	-2.4673E-01	0.0000E-01	-3.2592E-02	0.0000E-01
3	0.0000E-01	0.0000E-01	3.0171E-01	0.0000E-01	2.4485E-02	0.0000E-01
4	0.0000E-01	0.0000E-01	-3.2716E-01	0.0000E-01	-1.2500E-02	0.0000E-01
5	0.0000E-01	0.0000E-01	3.3608E-01	0.0000E-01	0.0000E-01	0.0000E-01
6	0.0000E-01	0.0000E-01	-3.2716E-01	0.0000E-01	1.2500E-02	0.0000E-01
7	0.0000E-01	0.0000E-01	3.0171E-01	0.0000E-01	-2.4485E-02	0.0000E-01
8	0.0000E-01	0.0000E-01	-2.4673E-01	0.0000E-01	3.2592E-02	0.0000E-01
9	0.0000E-01	0.0000E-01	4.1905E-01	0.0000E-01	-8.4679E-02	0.0000E-01
10	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01
11	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01

MODE SHAPE NO. 10 AT 4.86785E+02 CPS ( 3.05856E+03 RAD/SEC)

NODE	X TRANS	Y TRANS	Z TRANS	X ROT	Y ROT	Z ROT
1	0.0000E-01	0.0000E-01	-5.9611E-01	0.0000E-01	-1.4691E-01	0.0000E-01
2	0.0000E-01	0.0000E-01	2.1177E-01	0.0000E-01	9.7273E-02	0.0000E-01
3	0.0000E-01	0.0000E-01	-1.5635E-01	0.0000E-01	-1.1626E-01	0.0000E-01
4	0.0000E-01	0.0000E-01	8.0076E-02	0.0000E-01	1.2634E-01	0.0000E-01
5	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	-1.2980E-01	0.0000E-01

APPENDIX  
TRANSGEN User Manual

6	0.0000E-01	0.0000E-01	-8.0076E-02	0.0000E-01	1.2634E-01	0.0000E-01
7	0.0000E-01	0.0000E-01	1.5635E-01	0.0000E-01	-1.1626E-01	0.0000E-01
8	0.0000E-01	0.0000E-01	-2.1177E-01	0.0000E-01	9.7273E-02	0.0000E-01
9	0.0000E-01	0.0000E-01	5.9611E-01	0.0000E-01	-1.4691E-01	0.0000E-01
10	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01
11	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01	0.0000E-01

Table A.2. PAL Transient Analysis Output

FREEBEAM.MASS

MODAL VALUES FOR M MATRIX

3.467E-03	2.772E-03	2.437E-03	2.327E-03	2.217E-03	2.096E-03
1.958E-03	1.788E-03	1.569E-03	3.082E-03		

MODAL VALUES FOR C MATRIX

0.000E-01	0.000E-01	0.000E-01	0.000E-01	0.000E-01	0.000E-01
0.000E-01	0.000E-01	0.000E-01	0.000E-01		

MODAL VALUES FOR K MATRIX

2.222E-08	5.330E-08	1.884E+01	1.368E+02	5.021E+02	1.304E+03
2.739E+03	4.923E+03	7.560E+03	2.883E+04		

Table A.3. TRANSGEN Output File

The following is data for the PAL input files

freebeam.mode            freebeam.mass

There are 1 actuators and 4 sensors.

Actuators location,orientation are

1,3

Sensors location,orientation are

1,3    1,5    9,3    9,5

Zeros of element 1 1 of the transfer function numerator matrix

numerator multiplying factor is 907.1

0.000000E+00+    0.000000E+00\*j    possible pole zero cancellation

-0.286830E-36+    0.000000E+00\*j    possible pole zero cancellation

0.405669E-15+    0.606458E+02\*j

0.405669E-15+    -0.606458E+02\*j

## APPENDIX

### TRANSGEN User Manual

```

-0.748325E-10+ 0.196769E+03*j
-0.748325E-10+ -0.196769E+03*j
 0.423430E-07+ 0.411693E+03*j
 0.423430E-07+ -0.411693E+03*j
-0.156208E-05+ 0.707490E+03*j
-0.156208E-05+ -0.707490E+03*j
 0.991068E-05+ 0.108826E+04*j
 0.991068E-05+ -0.108826E+04*j
-0.133909E-04+ 0.156100E+04*j
-0.133909E-04+ -0.156100E+04*j

```

Zeros of element 2 1 of the transfer function numerator matrix  
numerator multiplying factor is 84.16

```

 0.000000E+00+ 0.000000E+00*j possible pole zero cancellation
-0.212906E-35+ 0.000000E+00*j possible pole zero cancellation
 0.148755E-16+ 0.391307E+02*j
 0.148755E-16+ -0.391307E+02*j
-0.225091E-10+ 0.157639E+03*j
-0.225091E-10+ -0.157639E+03*j
 0.239905E-07+ 0.357889E+03*j
 0.239905E-07+ -0.357889E+03*j
-0.115668E-05+ 0.643106E+03*j
-0.115668E-05+ -0.643106E+03*j
 0.838468E-05+ 0.101919E+04*j
 0.838468E-05+ -0.101919E+04*j
-0.122520E-04+ 0.149776E+04*j
-0.122520E-04+ -0.149776E+04*j

```

Zeros of element 3 1 of the transfer function numerator matrix  
numerator multiplying factor is -65.70

```

 0.000000E+00+ 0.000000E+00*j possible pole zero cancellation
-0.415497E-37+ 0.000000E+00*j possible pole zero cancellation
 0.122458E+03+ 0.000000E+00*j possible pole zero cancellation
-0.122458E+03+ 0.000000E+00*j possible pole zero cancellation
 0.307296E+03+ 0.000000E+00*j possible pole zero cancellation
-0.307296E+03+ 0.000000E+00*j possible pole zero cancellation
 0.425389E+03+ 0.359202E+03*j
 0.425389E+03+ -0.359202E+03*j
-0.425389E+03+ 0.359202E+03*j
-0.425389E+03+ -0.359202E+03*j
 0.497451E+03+ 0.104043E+04*j
 0.497451E+03+ -0.104043E+04*j
-0.497451E+03+ 0.104043E+04*j
-0.497451E+03+ -0.104043E+04*j

```

Zeros of element 4 1 of the transfer function numerator matrix  
numerator multiplying factor is 12.34

```

 0.000000E+00+ 0.000000E+00*j possible pole zero cancellation
-0.312078E-36+ 0.000000E+00*j possible pole zero cancellation
 0.787205E+02+ 0.000000E+00*j possible pole zero cancellation
-0.787205E+02+ 0.000000E+00*j possible pole zero cancellation
 0.238805E+03+ 0.000000E+00*j possible pole zero cancellation
-0.238805E+03+ 0.000000E+00*j possible pole zero cancellation

```

APPENDIX  
TRANSGEN User Manual

0.361175E+03+	0.338212E+03*j
0.361175E+03+	-0.338212E+03*j
-0.361175E+03+	0.338212E+03*j
-0.361175E+03+	-0.338212E+03*j
0.454174E+03+	0.101174E+04*j
0.454174E+03+	-0.101174E+04*j
-0.454174E+03+	0.101174E+04*j
-0.454174E+03+	-0.101174E+04*j

THE DENOMINTOR ROOTS ARE (NATURAL FREQUENCIES rad/sec)

.0000  
.0000  
87.94  
242.5  
475.9  
788.6  
1183.  
1659.

residue matrix for mode 1 is...

0.3204E+02  
-0.3647E-04  
0.3205E+02  
-0.3646E-04

residue matrix for mode 2 is...

0.9614E+02  
0.1602E+01  
-0.9614E+02  
0.1602E+01

residue matrix for mode 3 is...

0.1281E+03  
0.4963E+01  
0.1281E+03  
-0.4963E+01

residue matrix for mode 4 is...

0.1283E+03  
0.8410E+01  
-0.1283E+03  
0.8410E+01

residue matrix for mode 5 is...

0.1289E+03  
0.1183E+02  
0.1289E+03  
-0.1183E+02

residue matrix for mode 6 is...

0.1301E+03  
0.1538E+02

APPENDIX  
TRANSGEN User Manual

-0.1301E+03  
0.1538E+02

residue matrix for mode 7 is...

0.1316E+03  
0.1912E+02  
0.1316E+03  
-0.1912E+02

residue matrix for mode 8 is...

0.1319E+03  
0.2286E+02  
-0.1319E+03  
0.2286E+02

G matrix is

0.9614E+02  
0.1862E+03  
0.2293E+03  
0.2348E+03  
-0.2411E+03  
-0.2491E+03  
-0.2593E+03  
0.2716E+03

H matrix TRANSPOSED is

0.3333E+00 -0.3794E-06 0.3334E+00 -0.3793E-06  
0.5162E+00 0.8604E-02 -0.5162E+00 0.8604E-02  
0.5588E+00 0.2164E-01 0.5588E+00 -0.2164E-01  
0.5465E+00 0.3581E-01 -0.5465E+00 0.3581E-01  
-0.5345E+00 -0.4905E-01 -0.5345E+00 0.4905E-01  
-0.5221E+00 -0.6175E-01 0.5221E+00 -0.6175E-01  
-0.5077E+00 -0.7375E-01 -0.5077E+00 0.7375E-01  
0.4856E+00 0.8416E-01 -0.4856E+00 0.8416E-01

# APPENDIX

## TRANSGEN User Manual

Thu Jul 14, 1993

```
c
c -----
c      TRANSGEN
c      This is a program to read  output files from MSC PAL which
c      contains eigenvectors for each mode (nodal displacements).
c      The information is reordered and transfer functions and residues
c      are found for an input sensor-actuator set.
c      Other information needed is a modal mass and stiffness file name
c      and the locations and orientations of the sensors  and
c      actuators .
c      This version can handle a max of 30 modes for a model with
c      any number of node points.
c      This program is freeware and is not meant to be sold or exchanged
c      for any value
c
c      Written by Fred Wiesinger, Stanford
c      University
c -----
```

```
INTEGER MX, resnum, FLG, resout, transout, ghout, actnum, sennum
PARAMETER (MX=30)
parameter (ma=6)
DOUBLE PRECISION D(0:MX), R(MX), IMG(MX)
real*8 freq(MX), hold(100, 6), num(ma, ma, MX), xmass(MX), xstif(MX)
REAL*8 C(MX), freq2(MX), specres(ma, ma, MX), ASUM, reshold(MX)
integer actnods(ma), actdofs(ma), sennods(ma), sendofs(ma)
real*8 g(MX, ma), h(ma, MX)
character*20 fname, namest, mkname, namepal, atest, alpha, outfile,
1 echofile
integer CK

write(*, 1313)
1313 FORMAT(1X, 'THE REQUIRED INPUTS ARE:', /, 'NAME OF PAL MODAL OUTPUT
1 FILE', /, 'NAME OF PAL MODAL MASS AND STIFFNESS FILE (FROM TRANS. ANAL
2 .)', /, 'NUMBER OF NODES IN THE MODEL', /, 'NUMBER OF MODES TO BE USED
3 ', /, ' ACTUATOR AND SENSOR NODAL PLACEMENT AND ORIENTATION', /,
4 ' ENTER A "1" IF YOU WOULD LIKE TO SPECIFY THESE BY HAND ', /,
5 ' ENTER A "2" IF THEY SHOULD BE READ FROM A FILE')
READ(*, *) PARM

IF (PARM=1) THEN
WRITE(*, *) 'ENTER PAL MODES FILE NAME IN QUOTES'
READ(*, *) namepal
write(*, *) 'ENTER PAL MODAL MASS FILE NAME IN QUOTES'
READ(*, *) mkname
1 write(*, *) 'ENTER NUMBER OF NODES IN STRUCTURE AND NUMBER
1 OF MODES TO BE USED'
READ(*, *) nodnum, modenum
WRITE(*, *) 'ENTER NUMBER OF ACTUATORS, NUMBER OF SENSORS '
READ(*, *) actnum, sennum
do (i=1, actnum)
write(*, *) 'ENTER ACTUATOR # ', i, '- NODE NUMBER AND
1 D.O.F. (1-> 6)'
READ(*, *) actnods(i), actdofs(i)
repeat
do (i=1, sennum)
```

## APPENDIX

### TRANSGEN User Manual

```

write(*,*) 'ENTER SENSOR # ',i,'- NODE NUMBER AND D.O.F. (1-> 6) '
READ(*,*) sennods(i),sendofs(i)
repeat
write(*,*) 'Enter 1="yes",0="no" for the following output
1 options'
write(*,*) 'Residues output,zeros output,g & h matrices output'
read(*,*) resout,transout,ghout

elseif(PARM=2)THEN
WRITE(*,*) 'ENTER INPUT FILE NAME'
read(*,*) fname
open(unit=12,file=fname)
c read name of PAL output file
read(12,1000) namepal
1000 format(A20)
read(12,1009) mkname
1009 format(a20)
c read number of Nodes, and number of Modes to be used.****
read(12,*) nodnum,modenum
c read number of actuators and sensors ****
read(12,*) actnum,senum
do(i=1,actnum)
c read actuator nodes and d.o.f.'s.
READ(12,*) actnodes(i),actdofs(i)
repeat
c read sensor nodes and d.o.f.'s.
do(i=1,senum)
READ(12,*) sennods(i),sendofs(i)
repeat
c Enter 1="yes",0="no" for the following output options
c Residues output,zeros output,g & h matrices output
READ(12,*) resout,transout,ghout

close(unit=12)

else
GO TO 999 ! go to end
endif

c Input modal mass and stiffness from PAL output file for each mode.
c Modal mass and stiffness are generated by PAL's transient
c analysis when the "print modal equations" command is given.
write(*,*) mkname,' IS MASS MATRIX FILE'
open(unit=14,file=mkname)
read(14,1010)
1010 format(////////)
c READ MODAL MASS FOR EACH MODE
if(modenum.lt.6) go to 41
do 40 i=1,modenum/6
40 read(14,1011) (xmass(i*6+j-6),j=1,6)
1011 format(1x,6e12.4)
41 l=modenum-modenum/6*6
if(l.eq.0) go to 48
lk=modenum-1
read(14,1011) (xmass(i),i=lk+1,modenum)

```

# APPENDIX

## TRANSGEN User Manual

```

48 continue
C
C   FIND START OF STIFFNESS MATRIX
C
      do 50 i=1,200
        read(14,1012) alpha
1012 format(a3)
        atest=' M'                                !searches for the "r"
in modal
        if(atest.eq.alpha) go to 70                !third one is modal
stiffness
        GO TO 50                                    !section
70 alphtes=1+alphtes
        if(alphtes.ge.2)go to 59
50 continue
C
C   READ MODAL STIFFNESS FOR EACH MODE
59 read(14,*)
      if(modenum.lt.6) go to 61
      do 60 i=1,modenum/6
60 read(14,1011) (xstif(i*6+j-6),j=1,6)
61 l=modenum-modenum/6*6
      if(l.eq.0) go to 68
      lk=modenum-l
      read(14,1011) (xstif(i),i=lk+1,modenum)
68 continue
      close(unit=14)

```

---

```

      write(*,*) 'enter an output file name '
      read(*,*) outfile
      open(unit=15,file=outfile,status='new')
      write(15,1111) namepal,mkname
      write(15,1112)actnum, sennum
      write(15,1113) (actnods(i),actdofs(i),i=1,actnum)
      write(15,1114) (sennods(i),sendofs(i),i=1,sennum)
1111 format(1x,'The following is data for the PAL input files',/
1,4X,a20,a20)
1112 format(1x,'There are ',I1,' actuators and ',I1,' sensors.')
1113 format(2x,'Actuators location,orientation are',/,3x,
1i3,',',',i1,2x,i3,',',',i1,2x,i3,',',',i1,2x,i3,',',',i1,2x,i3,',',',
2i1,2x,i3,',',',i1,2x,i3,',',',i1,2x,i3,',',',i1,2x)
1114 format(2x,'Sensors location,orientation are',/,3x,
1i3,',',',i1,2x,i3,',',',i1,2x,i3,',',',i1,2x,i3,',',',i1,2x,i3,',',',
2i1,2x,i3,',',',i1,2x,i3,',',',i1,2x,i3,',',',i1,2x)

```

---

```

C
      open(unit=13,file=namepal)

      do 900 n=1,sennum
      do 899 m=1,actnum
          actnod=actnods(m)
          actdof=actdofs(m)

```

## APPENDIX

### TRANSGEN User Manual

```

sennod=sennods(n)
sendof=sendofs(n)
c


---


c **** begin the data read from PAL output file.*****
      rewind(unit=13)
      read(13,1007)
1007 format(//)
c For each mode read the frequency and the values of the modal
displacements
c in the corresponding degree of freedom directions and use the
displacements
c to calculate the residue for the given sensor and actuator transfer
c function.
      do 42 i=1,modenum
        read(13,1001) freq(i)
1001 format(/,45x,e11.5,///)
          do 5 j=1,nodnum
            if(j=actnod.or.j=sennod) then
              read(13,1002) (hold(j,k),k=1,6)
            else
              read(13,*)
            endif
1002 format(1x,6x,e11.4,1x,e11.4,1x,e11.4,1x,e11.4,1x,e11.4,1x,e11.4)
          5 continue
c *** calculate the residue for each mode *****
          num(n,m,i)=(hold(actnod,actdof))*(hold(sennod,sendof))
c ***** NOTE: the g and h matrices which are output are the Gm and Hm
c ***** matrices of the theory section of the TRANSGEN user manual.
          g(i,m)=hold(actnod,actdof)/xmass(i)
          h(n,i)=hold(sennod,sendof)
        42 continue

          if(resout=0.and.transout=0) go to 899
c


---


C FORM SPECIFIC RESIDUES (residue divided by modal mass for each mode)
C If a specific residue is equal to zero THAT WHOLE MODE WILL BE
C OMITTED FROM THE TRANSFER FUNCTION CALCULATION.
      lowres=0
      do (i=1,modenum)
        specres(n,m,i)=num(n,m,i)/xmass(i)
        if(specres(n,m,i)=0.) lowres=lowres+1
      repeat
        if(lowres=modenum)then
          write(15,*) 'all residues are 0,sensor ',n,' input is not
associated with actuator ',m,'output'
          go to 899
        endif
        if(transout=0)go to 899
c


---


vector      do (i=1,modenum)                                !define freq2 and reshold
      reshold(i)=specres(n,m,i)
      freq2(i)= -freq(i)**2

```

## APPENDIX

### TRANSGEN User Manual

```

repeat
c
-----
    resnum=modenum-lowres
    if(resnum=1) then
        write(*,*) 'only one residue is nonzero so t.f. is trivial'
    go to 899
    endif
c
-----
c  compute products of partial fractions          (i.e. multiply out the
partial
c  fractions.
    call pf(modenum, resnum, reshold, freq2, C, ASUM, FLG)
c
-----
C NUMERATOR COEFFICIENTS OUTPUT FROM pf ARE ORDERED AS BELOW...
C ASUM*{S^N+ C(N)*S^N-1+ C(N-1)*S^N-2+ ... +C(2)*S+ C(1)}
C WHERE N=modenum-1,ASUM=numerator multiplying factor
C
C REORDER COEFFICIENTS FOR BAIRSTOW ALGO.
C ODD COEFFICIENTS ARE ZERO SINCE SYSTEM IS UNDAMPED,N=2*(MODENUM-1)
150 DO (I=0,MX)
    D(I)=0.
    REPEAT
    IF((FLG=1).AND.(resnum.eq.1)) then
        write(*,*) 'numerator is a constant = ',ASUM
        go to 899
    endif
    N=2*(resnum-1)
    D(0)=1
    DO (I=1,N)
        CK=I/2
        IF(2*CK-I.LT.0)THEN
            D(I)=.00001                                !BAIRSTOW ALGO. NEEDS ALL
COEFFICIENTS ≠ 0
        ELSE
            D(I)=C((N-I+2)/2)
        ENDIF
    REPEAT
c
-----
c  call subroutine to factor transfer function numerator
C MASROOT FINDS ROOTS OF NUMERATOR POLYNOMIAL
    call MASROOT(N,D,R,IMG)
c
-----

    write(15,*)
    write(15,*) 'Zeros of element ',n,' ',m,
1  ' of the transfer function numerator matrix'
    write(15,*) 'numerator multiplying factor is ',ASUM
    do(i=1,N)
c    if(IMG(i)=-IMG(i+1).and.i.ne.N) go to 1876
        flag1=0
        do (j=1,modenum)

```

## APPENDIX

### TRANSGEN User Manual

```

        if(abs((abs(IMG(i))-freq(j))/(freq(j)+1e-7)).lt.1e-2) flag1=1
        repeat
        if(flag1=1)then
        write(15,4041) R(i),IMG(i)
        else
        write(15,4040) R(i),IMG(i)
        endif
1876 repeat
4040 format(4x,e14.6,'+ ',e14.6,'*j')
4041 format(4x,e14.6,'+ ',e14.6,'*j' possible pole
1 zero cancellation ')

899 continue
900 continue
      close(unit=13)
c          END OF MAIN LOOPS

c


---


      write(15,*)
      WRITE(15,*) 'THE DENOMINTOR ROOTS ARE (NATURAL FREQUENC
1IES rad/sec)'
      DO (I=1,modenum)
      WRITE(15,*) freq(I)
      REPEAT
c


---


      if(resout=0) go to 1400
c  print residue matrices to the output file in mode order
      do(i=1,modenum)
      write(15,1555) i
1555  format(/,1x,'residue matrix for mode ',i2,' is...')
      do(n=1,sennum)
      write(15,1395) (specres(n,m,i),m=1,actnum)
      repeat
      repeat
      write(15,1398)
1395 format(1x,6(e12.4))
1398 format(///)
c


---


1400 if(ghout=0) go to 999
      write(15,*) 'G matrix is'
      do (i=1,modenum)
      write(15,1450) (g(i,j),j=1,actnum)
      repeat
      write(15,*)
      write(15,*) 'H matrix TRANSPOSED is'
      do (i=1,modenum)
      write(15,1450) (h(j,i),j=1,sennum)
      repeat
1450 format(2x,6(e12.4))
c
c

```

# APPENDIX

## TRANSGEN User Manual

```

999 write(*,*) 'Program complete. Please hit return.'
      pause
      stop
      end

C *****
C ***** SUBROUTINES FOLLOW *****
C *****

C ***** P.F. TO TRANS ***** 1/13/88 FW
C THIS IS A SUBROUTINE WHICH WILL FIND THE TRANSFER FUNCTION POLE/ZERO
C FORM FROM A POLE-RESIDUE FORM.
C THE INPUTS ARE N (THE NUMBER OF MODES), Q (THE NUMBER OF NONZERO
C RESIDUES, THE N-VECTOR A (SPECIFIC RESIDUES),
C AND THE N-VECTOR SIG (NEG OF MODAL FREQUENCIES SQUARED).
C THIS SUBROUTINE CALLS THE SUBROUTINE POLYMUL TO OBTAIN NUMERATOR
C POLYNOMIAL COEFFICIENTS FROM NUMERATOR ROOTS.
C *****
C INPUTS: N=modenum, Q=resnum, A=specres, SIG=freq2
C OUTPUTS: TCR, ASUM, FLG, (and Q could be updated)
C          subroutine pf(N, Q, A, SIG, TCR, ASUM, FLG)
C          INTEGER Q, FLG
C          PARAMETER (MX=30)
C          REAL*8 A (MX) , SIG (MX) , RR (MX) , CR (MX) , TCR (MX) , ASUM

C
C MULTIPLY OUT THE PARTIAL FRACTIONS, THEN ADD POWERS OF S COEFFICIENTS
C AND DIVIDE EACH COEFFICIENT BY THE COEFFICIENT OF THE HIGHEST POWER OF S.
C ASUM=0.
C DO 10 I=1, N
C   L=0
C   DO 20 J=1, N
C     IF (J.EQ.I) GO TO 20
C     IF (A(J)=0.) GO TO 20
C     L=L+1
C     RR(L)=SIG(J)
C 20 CONTINUE
C INPUT TO POLYMUL IS VECTOR RR, AND # OF FACTORS TO BE MULTIPLIED OUT.
C OUTPUT IS VECTOR CR WHICH CONTAINS POLYNOMIAL COEFFICIENTS.
C CALL POLYMUL(RR, Q-1, CR)
C DO 30 K=1, Q-1
C   IF (I.EQ.1) TCR(K)=0
C ADD UP ALL THE COEF.'S OF LIKE POWERS OF S ...
C TCR(K)=TCR(K)+CR(K)*A(I)
C 30 CONTINUE
C ASUM=A(I)+ASUM
C 10 CONTINUE

C
C ***** OUTPUT *****
C IF (ASUM=0.) THEN !IF ASUM=0 SET ASUM=NEXT COEFFICIENT AND
C   FLG=1 ! PROCEED
C   ASUM=TCR(Q-1)
C   Q=Q-1
C   ENDIF
C DO 40 I=1, Q-1

```

## APPENDIX

### TRANSGEN User Manual

```

TCR(I)=TCR(I)/ASUM          !DIVIDE EACH COEFFICIENT BY LEADING
COEFFICIENT
40 CONTINUE
    close(unit=12)

    return
END

C
C
C *****
C THIS IS A SUBROUTINE WHICH WILL RETURN THE COEFFICIENTS OF A
C POLYNOMIAL WHOSE ROOTS ARE SENT TO IT.
C THE INPUTS ARE THE ROOTS (I.E. -5 FOR (S+5) ),AND THE NUMBER OF
C ROOTS, N. THE OUTPUT IS AN N VECTOR CR CONTAINING THE POLYNOMIAL
C COEFFICIENTS AS SHOWN BELOW:
C   P(S)=S^N +CR(N)*S^(N-1) +....+C(2)*S +C(1)
C *****
C SUBROUTINE POLYMUL(RR,N,CR)
C REAL*8 CR(100),RR(100),DR(100)
C INTEGER N
C IF(N.NE.1) GO TO 3
C CR(2)=1
C CR(1)=-RR(1)
2 RETURN
3 DO 10 I=2,N
  CR(I)=0
10 CONTINUE
  CR(2)=1
  CR(1)=-RR(1)
  DO 20 I=3,N+1
    DO 30 J=1,I-1
      DO 40 L=1,I
        DR(L)=CR(L)
40 CONTINUE
        CR(I-J+1)=DR(I-J)-DR(I-J+1)*RR(I-1)
30 CONTINUE
        DR(1)=CR(1)
        CR(1)=-DR(1)*RR(I-1)
20 CONTINUE
C   DO 90 I=1,N
C   WRITE(*,1000) CR(I)
C 1000 FORMAT(1X,'CR"S= ',8F12.8)
C   90 CONTINUE
C   RETURN
C   END

C ***** D(I)= COEFFICIENT OF X**(N-I) *****
C Subroutine: Finds roots of polynomial using Bairstow algorithm.
C Bairstow works fine for non-repeated roots and non-zero coefficients.
C N is order of polynomial
C D must be of dimension N+1; best to divide through by D(0), the coef
C of x**n before call to ROOTS
C Outputs: IMG is imag part of roots, R is real part of roots
C IERR=1 if there is a error in converging

```

# APPENDIX

## TRANSGEN User Manual

```

SUBROUTINE MASROOT(N,D,R,IMG)
IMPLICIT NONE
INTEGER I,J,J2,N,I1,L,MX,IERR
PARAMETER(MX=30)
DOUBLE PRECISION PI(MX),PR(MX),D(0:MX)      ! inputs
DOUBLE PRECISION R(MX),IMG(MX)             ! outputs
DOUBLE PRECISION Q(0:MX),P(0:MX),B(0:MX)    ! work vars
DOUBLE PRECISION E,R2,S2,R3,S3,R4,S4,R5,S5,D6,R6,S6
DOUBLE PRECISION RA,RB,IA,IB
IERR=0
DO (I=1,MX)
  R(I)=0.
  IMG(I)=0.
REPEAT
DO (I=1,N)
  PR(I)=0.
  PI(I)=0.
REPEAT
J=1
J2=0
E=.0000001
680 L=0
  IF (J.NE.N) THEN
    R2=-PR(J)-PR(J+1)
    S2=PR(J)*PR(J+1)+PI(J)**2
    I1=N-2*J2
    IF (I1.EQ.2) GOTO 720
    IF (I1.GT.2) GOTO 730
  ENDIF
  R(J)=-D(1)
  GOTO 910
720 R2=D(1)
  S2=D(2)
  CALL QUAD(R2,S2,J,RA,RB,IA,IB)
  R(J)=RA
  R(J+1)=RB
  IMG(J)=IA
  IMG(J+1)=IB
  GOTO 910
730 B(0)=1
  B(1)=D(1)-R2
  DO (I=2,I1)
    B(I)=D(I)-R2*B(I-1)-S2*B(I-2)
  REPEAT
  P(0)=0
  P(1)=-1
  DO (I=2,I1)
    P(I)=-B(I-1)-R2*P(I-1)-S2*P(I-2)
  REPEAT
  Q(0)=0
  Q(1)=0
  DO (I=2,I1)
    Q(I)=-B(I-2)-R2*Q(I-1)-S2*Q(I-2)
  REPEAT
  R3=B(I1-1)
  S3=B(I1)+R2*B(I1-1)

```

APPENDIX  
TRANSGEN User Manual

```

R4=P (I1-1)
S4=P (I1)+R2*P (I1-1)+B (I1-1)
R5=Q (I1-1)
S5=Q (I1)+R2*Q (I1-1)
D6=R4*S5-R5*S4
IF (ABS (D6) .LT. E) THEN
  GOTO 810
ELSE
  R6=(S3*R5-S5*R3)/D6
ENDIF
S6=(R3*S4-R4*S3)/D6
R2=R2+R6
S2=S2+S6
IF ((ABS (R6) .GT. E*ABS (R2)) .OR. (ABS (S6) .GT. E*ABS (S2))) GOTO 820
810 CALL QUAD (R2, S2, J, RA, RB, IA, IB)
  R(J)=RA
  R(J+1)=RB
  IMG (J)=IA
  IMG (J+1)=IB
  GOTO 850
820 IF (L.GE.N*50) THEN
  WRITE (*,*) ' Polynomial solver did not converge'
  IERR=1
  RETURN
ENDIF
L=L+1
GOTO 730
850 J2=J2+1
  J=J+2
  DO (I=1, I1)
    D (I)=B (I)
  REPEAT
  GOTO 680
910 WRITE (*,*)
  RETURN
END

```

```

* Solves quadratic equation
SUBROUTINE QUAD (R2, S2, J, RA, RB, IA, IB)
IMPLICIT NONE
INTEGER J
DOUBLE PRECISION DI, R2, S2, RA, RB, IA, IB
R2=R2/2.
DI=R2**2-S2
IF (DI.GE.0.) THEN
  DI=SQRT (DI)
  RA=DI-R2
  RB=-DI-R2
  IA=0.
  IB=0.
ELSE
  DI=SQRT (-DI)
  RA=-R2
  RB=-R2
  IA=DI
  IB=-DI

```

APPENDIX  
TRANSGEN User Manual

ENDIF  
RETURN  
END

APPENDIX  
TRANSGEN User Manual

**References**

- [1] Plescia, C. and Chu, P., "Adaptive Rigid Body Control for an Evolving Space Station, Preliminary Analysis Report", Ford Aerospace document WDL-TR 10882, Nov. 1986.
- [2] Wie, B., "On the Modelling and Control of Flexible Space Structures", Stanford University Dept. of Aeronautics and Astronautics Report #525, June 1981.