

AD-A136 781

DECENTRALIZED CONTROL OF A LARGE SPACE STRUCTURE USING

1/2

DIRECT OUTPUT FEEDBACK(U) AIR FORCE INST OF TECH

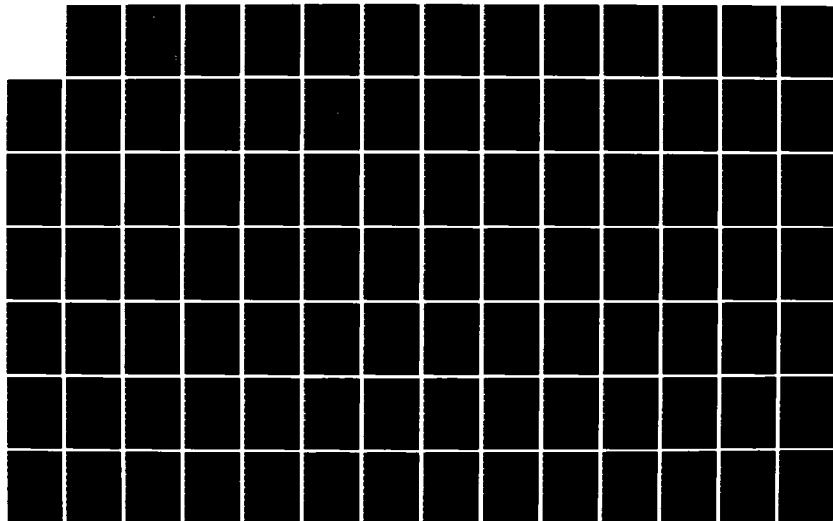
WRIGHT-PATTERSON AFB OH SCHOOL OF ENGI... D V THYFAULT

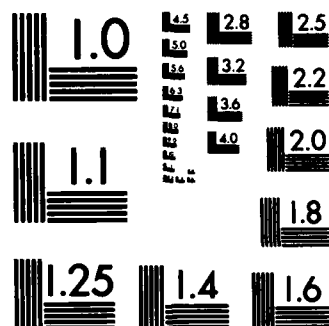
UNCLASSIFIED

DEC 83 AFIT/GA/AA/83D-8

F/G 12/1

NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

AD A136781



DECENTRALIZED CONTROL OF A LARGE SPACE
STRUCTURE USING DIRECT OUTPUT FEEDBACK

THESIS

David V. Thyfault
Captain, USAF

AFIT/GA/AA/83D-8

DTIC FILE COPY

DEPARTMENT OF THE AIR FORCE
AIR UNIVERSITY

AIR FORCE INSTITUTE OF TECHNOLOGY

DTIC
ELECTE
JAN 13 1984
S E D

Wright-Patterson Air Force Base, Ohio

This document has been approved
for public release and sale; its
distribution is unlimited.

84 01 13 015

AFIT/GA/AA/83D-8

①

DECENTRALIZED CONTROL OF A LARGE SPACE
STRUCTURE USING DIRECT OUTPUT FEEDBACK

THESIS

David V. Thyfault
Captain, USAF

AFIT/GA/AA/83D-8

Approved for public release; distribution unlimited

DTIC
UNCLASSIFIED
L

DECENTRALIZED CONTROL OF A LARGE SPACE
STRUCTURE USING DIRECT OUTPUT FEEDBACK

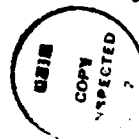
THESIS

Presented to the Faculty of the School of Engineering
of the Air Force Institute of Technology
Air University
in Partial Fulfillment of the
Requirements for the Degree of
Master of Science in Astronautical Engineering

David V. Thyfault
Captain, USAF

December 1983

Accession For	
NTIS GRA&I	☒
DTIC TAB	☐
Unannounced	☐
Justification	
By _____	
Distribution/	
Availability Codes	
Avail and/or	
Dist	Special
A-1	



Acknowledgments

This study was conducted under the sponsorship of the Air Force Flight Dynamics Laboratory, AFFDL/FIGCA.

I have many to thank for guiding me through this thesis effort. To my thesis advisor, Dr. Robert Calico, thank you for keeping me pointed in the right direction and for your contagious enthusiasm for this control method. To Jeff Hess, my ever present mentor, thanks for drilling me in the basics. I am ever grateful to Ed Aldridge, Beth Copenhaver, and Vickie Tischler for their generous help with creating and continued pest control of the computer programs. Many thanks to Mike Wallace and Sparky Olsen for their help with the finite element data files. To Debbie, my express lane typist, thanks for always being there. But above all, I thank and praise the Lord (I Timothy 1:17).

David V. Thyfault

Table of Contents

	Page
Acknowledgments	ii
List of Figures	iv
List of Tables	v
Notation	vi
Abstract	viii
I. Introduction	1
II. LSS Model Configuration	6
III. System Model	15
Equations of Motion	15
Control Model	17
Modal Control	22
Three Controllers	24
Design Feedback Gain	26
Transformed DOFB	28
Sensor/Actuator Requirements	30
IV. Transformation Technique	33
V. Computer Model	41
VI. Investigation	46
VII. Results	53
VIII. Conclusions	67
IX. Recommendations	69
Appendix A: Main Program and Subroutine Listing	70
Appendix B: NASTRAN Analysis - 12 Modes	116
Appendix C: NASTRAN Analysis - 33 Modes	119
Bibliography	125
Vita	127

List of Figures

Figure	Page
1. Decentralized Output Feedback Structure	4
2. CSDL 2 Conceptual View	7
3. Finite Element Representation	8
4. CSDL 2 Optical System	9
5. Structural Dimensions	10
6. CSDL 2 Node Placement	11
7. Computer Program Flow Chart	42

List of Tables

Table	Page
I. CSDL 2 Node Coordinates	12
II. CSDL 2 Lumped Mass Distribution	13
III. CSDL 2 Sensor/Actuator Locations and Orientations	13
IV. Key Results of NASTRAN Eigenvalue Analysis on Nominal CSDL 2 Model	14
V. Relative Angles (in Degrees) Between Modes (12)	49
VI. Relative Angles (in Degrees) Between Modes (36)	50
VII. Eigenvalue Analysis - Single Controller	59
VIII. Eigenvalue Analysis - Single Controller	60
IX. Eigenvalue Analysis - 3 Controllers	61
X. Eigenvalue Analysis - 3 Controllers (Optimal)	62
XI. Eigenvalue Analysis - 4 Controllers	63
XII. Eigenvalue Analysis - 4 Controllers	64
XIII. Eigenvalue Analysis - 3 Controllers	65
XIV. Eigenvalue Analysis - 3 Controllers	66

Notation

A	-	Plant matrix
B	-	Control matrix
C	-	Output matrix
D	-	Nodal attitude-evaluated actuator location matrix
E	-	Damping matrix
G	-	Optimal steady full-state feedback gain matrix
g	-	Generalized coordinate
I	-	Identity matrix
J	-	Quadratic performance index
K	-	Suboptimal feedback gain matrix; stiffness matrix
$\hat{K}$	-	Untransformed feedback gain matrix
$\tilde{K}$	-	Design feedback gain matrix
M	-	Mass matrix
n_a	-	Number of actuators
n_m	-	Number of modes
Q	-	State weighting matrix
R	-	Control weighting matrix
S	-	Singular value matrix; solution to steady state matrix Riccati equation
T	-	Transformation matrix
u	-	Control vector
V	-	Matrix of right singular vectors
W	-	Matrix of left singular vectors
w	-	Transformed output vector
x	-	State vector
y	-	Output vector
v	-	Transformed control vector
Γ	-	Transformation matrix
κ	-	Sum of $\hat{K}_i$'s
Λ	-	Sum of $\tilde{K}_i$'s
Φ	-	Modal matrix
Ψ	-	Structural mode amplitude matrix
ψ	-	Element of Ψ matrix
σ	-	Singular value
Σ	-	Summation when used with indices; singular value matrix
ζ	-	Damping ratio
ω	-	Natural frequency
Ω	-	ω^2

Notation (Continued)

Subscripts

- c - Critical or controlled modes
- i - Index for variable
- j - Index for variable
- m - Index for variable
- p - Partition of V matrix associated with zero singular values;
position elements
- r - Partition of W matrix associated with zero singular values;
residual modes
- s - Partition of V or W matrices associated with non-zero singular
values; suppressed modes
- um - Unmodeled modes
- v - Velocity elements
- 2N - N-1 Control matrices

Superscripts

- * - Transformed matrix
- + - Inverse of the singular value decomposition of a matrix

Abstract

Direct output feedback control methods are used to develop a multiple-input multiple-output controller. The controller is then applied to the Charles Stark Draper Laboratory 2 (CSDL 2) model. The CSDL 2 model is a sophisticated optical space structure representative of large flexible space structures. This model consists of 59 nodes and 23 lumped masses. The beam elements are fully connected and may support axial, transverse, and torsional deformations. NASTRAN is employed to generate modal approximations of the model as well as the mode shapes and frequencies of the resulting modes. Of the numerous modes available for the model, only the first 36 modes are utilized and implemented in the controller.

The control problem is formulated in state space form and direct output feedback is implemented. The state is represented as modal amplitudes and rates. System outputs are obtained by rate sensors and control is applied by point force actuators. Since the observation matrices are not of full rank, their generalized (Penrose) inverses are obtained. These and the steady full-state feedback optimal control gains are used to determine suboptimal feedback gains. In the multiple controller, individual controllers actively control the modes assigned to it, but the modes assigned to the other controllers interact with all controllers causing control and observation spillover. The spillover that appears in coupled decentralized systems is eliminated using transformation matrices obtained via modal suppression techniques. The transformation matrices are applied to the feedback gains to drive the spillover terms to zero and decouple the controllers.

Control was accomplished using four controllers without residual modes and three controllers with residual modes. Conditions for which the stability of the model is assured are developed. Arrangement of modal assignment to controllers was varied to examine modal selection and grouping. Sensitivity to uncontrolled residual modes was examined. Direct output feedback response was compared to that obtained using optimal full-state feedback. Full controller decoupling was achieved and stability of the non-rigid body modes maintained. The investigation demonstrated the feasibility of using decentralized direct output feedback in maintaining system stability.

DECENTRALIZED CONTROL OF A LARGE SPACE STRUCTURE USING DIRECT OUTPUT FEEDBACK

I. Introduction

The study of Large Space Structure (LSS) Control is a growing field of interest primarily due to the involvement by the Department of Defense (DoD) and the National Aeronautics and Space Administration (15,16). Technology and politics are leading us into space to perform scientific, exploratory, and eventually operational missions that use large structures. The future of space is going to see deployment of large space structures for a wide spectrum of missions including communications, space defense, power generation, manufacturing, and research. Future DoD large space system concepts such as High Altitude Large Optics (HALO), High Energy Laser Optics (HELO), and Millimeter (MM) may be among those deployed. These systems combine large size with stringent line-of-sight (LOS) pointing and jitter requirements. Coupled with these performance requirements are the needs to minimize spacecraft mass. In general, the constraints on large space structures result in highly flexible spacecraft for which passive methods of material selection and construction techniques do not provide adequate control. This observation has led to the idea of building control-configured spacecraft using active stability-augmentation systems integrated with the space structure.

To achieve this control-configured spacecraft, the control designer must understand the LSS control problem. Some characteristics of the LSS control problem are: (1) infinite dimensional in theory, i.e. distributed

parameter system, and very large dimensional in practice, (2) they have many low resonant frequencies and often these appear in closely-spaced "clumps", (3) their natural damping is very light, (4) prediction of their behavior in space by ground testing is quite limited, and (5) requirements for shape, orientation, alignment, vibration suppression, and pointing accuracy are extremely rigorous (6,18). The first characteristic is considered fundamental and many control engineers are developing and using rational methods to reduce theoretical models to finite dimensional ones that are representative of the actual system and enable control systems to deliver adequate performance (4,6,9,15,19,21). The LSS is modeled as a reduced order discrete system via finite element modeling that results in a finite-dimensional modal representation. The number of modeled modes in the physical model is referred to as the order of the system. When a distributed system is "discretized", the resulting physical model suffers errors that can result in overall system instability when not accounted for (1,10). From this physical model a reduced order evaluation model is selected to maintain model fidelity. Since the evaluation model may contain more modes than can be controlled, a design model is determined to include modes which if ignored would degrade system performance beyond mission requirements (18). Critical modes, denoted x_c , are those modes of the design model which are chosen to be explicitly controlled in order to assure stability and achieve performance requirements for the system. Residual modes, denoted x_r , are those modes which exist in the infinite dimensional system that are not critical in the sense defined above. Observation spillover is the contamination of the sensor outputs by modes not contained in a controller. Control spillover is the excitation of mode dynamics due to a controller that does not contain those modes. Residual modes may be subdivided

into design, evaluation, physical, and unmodeled modes. Design residual modes are those which are included in the controller design model, but not explicitly controlled. Evaluation residual modes are those modes included in the evaluation model but not in the design model. Physical residual modes are those modes which are included in the physical model, but not in the evaluation model. Unmodeled residual modes, denoted x_{um} , are those modes which exist, but are not included in any finite-dimensional model of the system under study (18).

The specific mission of the LSS forces the performance requirements which can only be met through active control. There are many appropriate control strategies for large flexible space structures (1,2,10,18,20). The choices center on the theme of constant-gain linear full state feedback control. Of the choices available, decentralized control offers several advantages. In decentralized control each controller is designed to control some of the critical modes allowing the use of low-order computer algorithms. The feedback system structure used, shown in Figure 1, provides each control channel with the same information structure. This method of decentralized control using modern (cost optimal) control for finite element models of LSS has been demonstrated by several researchers (1,2,13). They found transformation matrices applied to feedback gains proved an effective method for eliminating spillover and enhancing system performance. Optimal control methods which require estimators to reconstruct the states, however, require four states for each critical mode (mode amplitude and rate and estimated amplitude and rate). With four states per mode the computation burden rises rapidly with the number of critical modes, especially computation of the algebraic Riccati equations (8,11,20). This has motivated the use of suboptimal techniques that do not require an observer

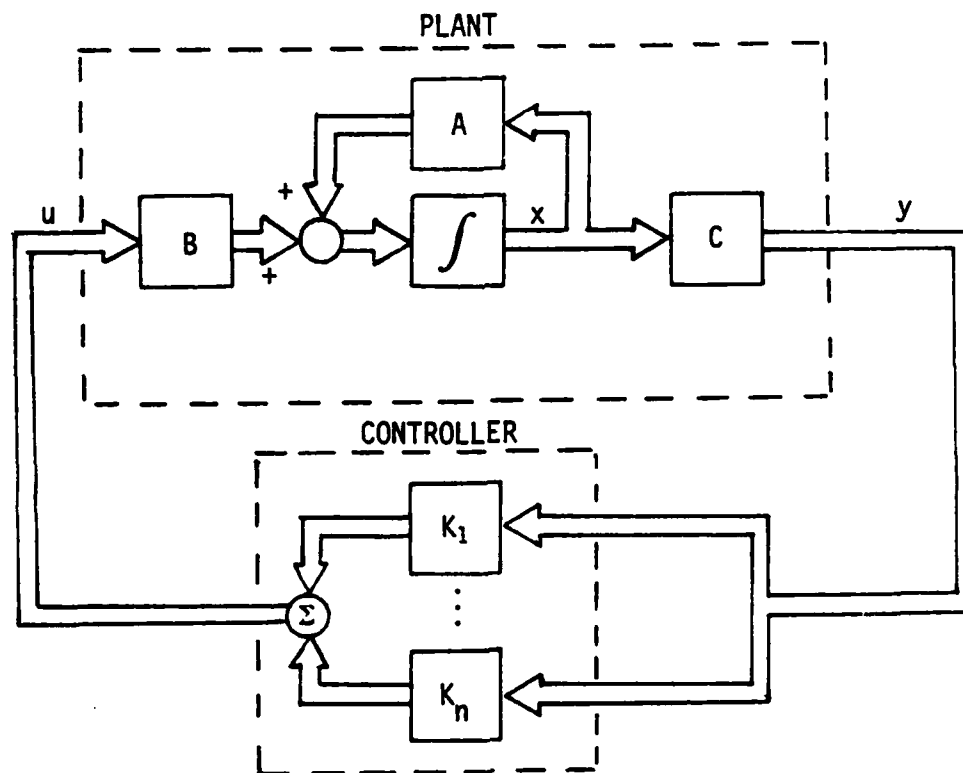


Figure 1. Decentralized Output Feedback Structure

and consequently either greatly reduces the computational burden for controlling the same number of modes or allows control of twice as many critical modes. This concept of a decentralized direct output feedback (DOFB) controller was first proposed by Calico (6) and will be discussed in detail in Section III.

Even with an expanded number of controllers made available by decentralized control, the number of modes that may be controlled is small compared to the number of modes that exist for a given structure. Consequently the selection of modes to be controlled must be made carefully. Only those modes affecting performance need be controlled. The terms "controlled" and "critical" will be used interchangeably for these modes. The remaining uncontrolled modes fall into three categories: suppressed, residual and

unmodeled. Natural damping in the structure will normally prevent instabilities arising from the higher frequency modes, so, for model simplicity, those are truncated and left as unmodeled modes. The remaining uncontrolled modes are modeled modes. Of these, some may have destabilizing effects due to spillover and therefore have to be made transparent to the controller. These are referred to as suppressed modes. The last mode group is modeled, uncontrolled and unsuppressed. These are the residual modes and may move freely when control is applied. They may become more stable, less stable or unstable due to control and observation spillover from the critical modes.

The intent of this thesis is to apply direct output feedback in developing a decentralized control system of three or four controllers. This control system will be applied to the Charles Stark Draper Laboratory, Inc. model #2 (CSDL 2). The CSDL 2 is a representative LSS for a precision optical space system. Eigenvalue analysis of the closed loop system will be used to evaluate system performance. In applying the control method, the output (observation) matrix is configured to supply position sensor, rate sensor, and position/rate sensor information. Point force actuators provide the state variable feedback control.

The following sections will detail the CSDL 2 model and its finite element representation. The control and matrix transformation methods are discussed. Finally, the computer program and results are presented.

II. LSS Model Configuration

A central issue in the active control of large space structures is the development of realistic and mathematically correct models for open and closed loop dynamic plants. NASTRAN and SPAR are currently the primary tools for generating models of sophisticated conceptual spacecraft. In order to assess the control technique of decentralized direct output feedback, as well as other control methods, a generic model is required. The Charles Stark Draper Laboratory (CSDL) developed just such a model as part of the Active Control of Space Structures (ACOSS) program sponsored by DoD's Advanced Research Projects Agency (4,5,14). The model was originally titled ACOSS Model #2, but is commonly referred to as the CSDL 2 model. CSDL 2 is a finite element representation of an optical space structure. It has incorporated the desired features of structural design based on realistic sizes and weights, a simple unclassified optical system with associated performance measures and tolerances, and a set of disturbances typical of equipment vibration and attitude control. Figure 2 shows a conceptual view of the structure and Figure 3 is a finite element representation of the model.

CSDL 2 represents a three mirror optical space telescope system. The two major components of the system are the optical support structure and the equipment section. The optical support structure consists of the upper mirror support truss, the lower mirror support truss, and metering truss. The upper mirror support truss contains the primary mirror (convex surface) and the tertiary mirror (concave surface). The lower mirror support truss contains the secondary mirror (flat surface) and the focal plane (image receiving device). The metering truss maintains the mirror separation

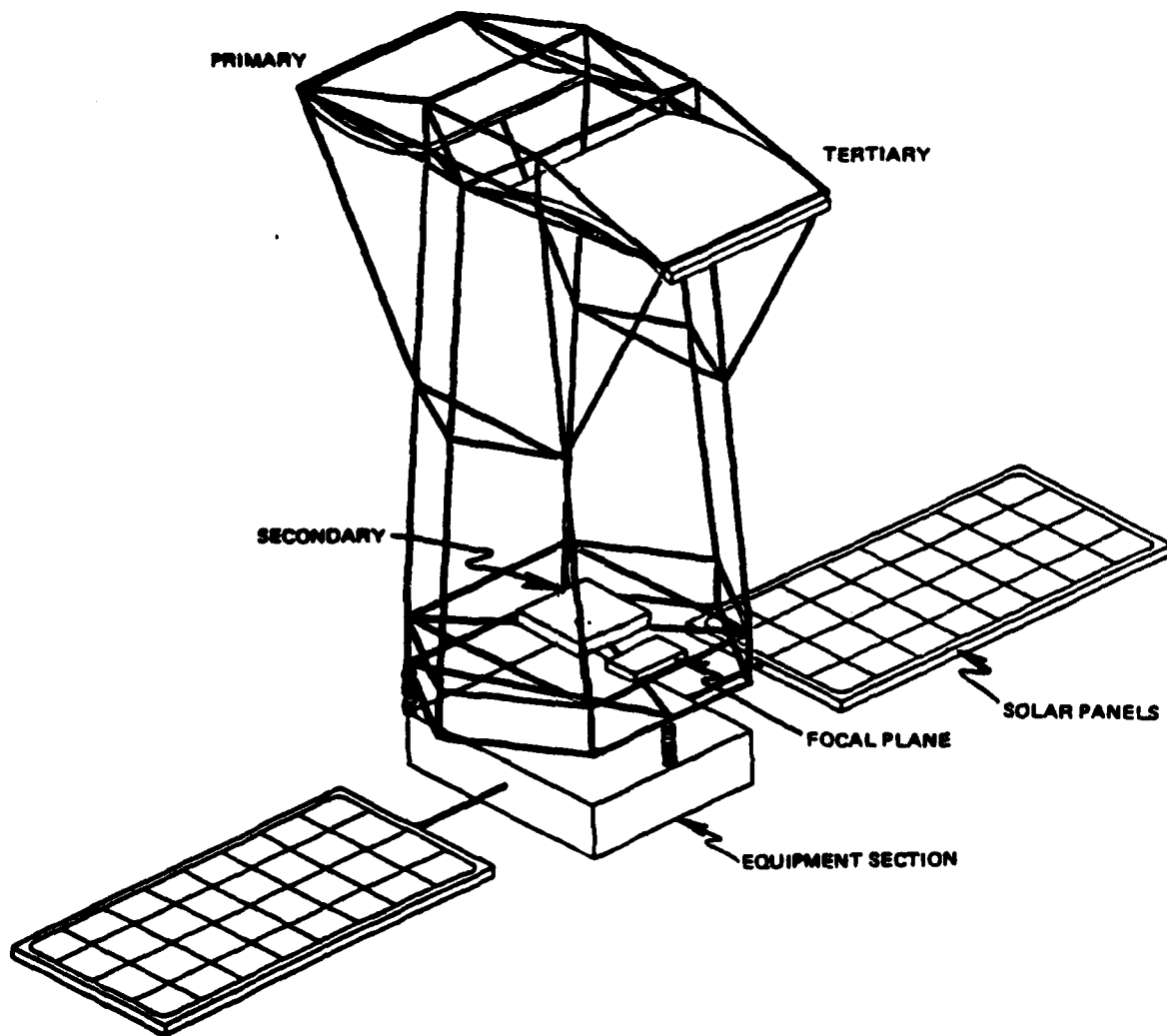


Figure 2. CSDL 2 Conceptual View

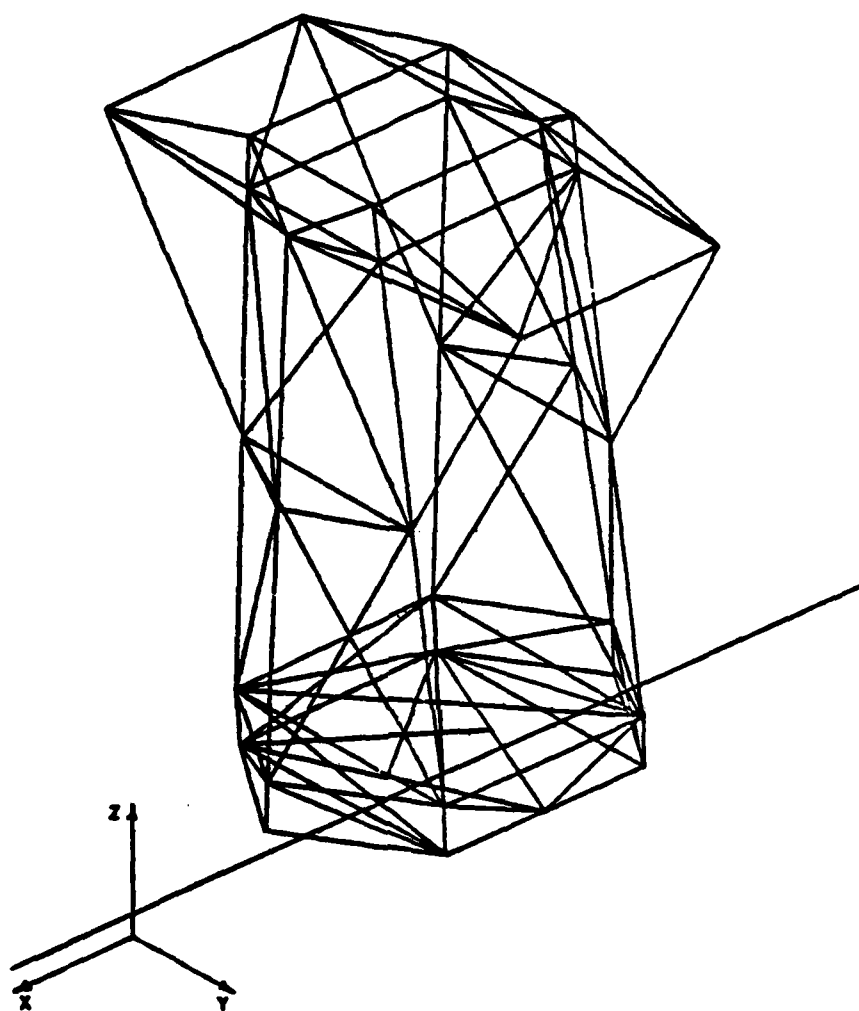


Figure 3. Finite Element Representation

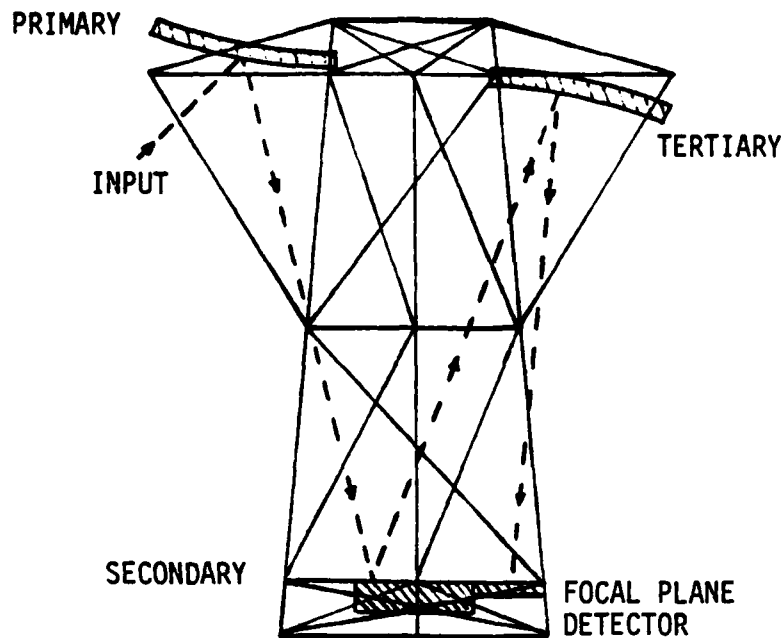


Figure 4. CSDL 2 Optical System

and is the key section when examining defocus. The optical support structure and mirror placement are shown in Figure 4. Attached to the lower side of the lower mirror support truss is the equipment section which consists of the control package, modeled as a rigid body, with two cantilevered flexible solar panels. The full structure is approximately twenty-eight meters high and has a mass of 9300 kilograms. The structural dimensions are shown in Figure 5.

The finite element model of the structure contains fifty-nine node points, but the actual structure has only fifty-one nodes. The extra nodes were added to provide more detail in the modeling of the mirrors and equipment section (5). The coordinates of the nodes are given in Table I, and the placement of the nodes in the support structure are shown in Figure 6. The truss members are fully joined so that bending and torsion are

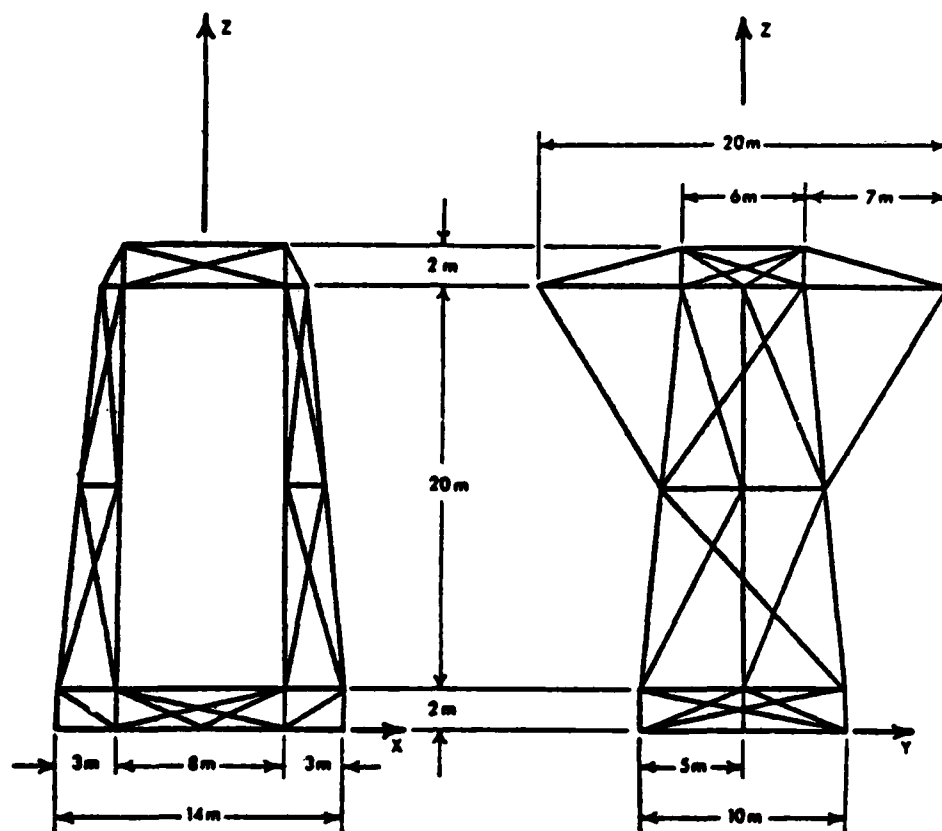


Figure 5. Structural Dimensions

allowed. The truss elements are made of graphite-epoxy and assumed to be massless. The system mass is lumped at twenty-three nodes and distributed as shown in Table II. The largest mass is located in the equipment package, as would be expected.

This model makes use of twenty-one pairs of collocated force actuators and position sensors. A list of sensor/actuator locations and orientations is provided in Table III.

The key results of an eigenvalue analysis performed on this model are listed in Table IV. The generalized mass and stiffness, as well as the natural frequency of the first thirty-six structural modes is given. Of the modes listed, those with an asterisk are considered critical for

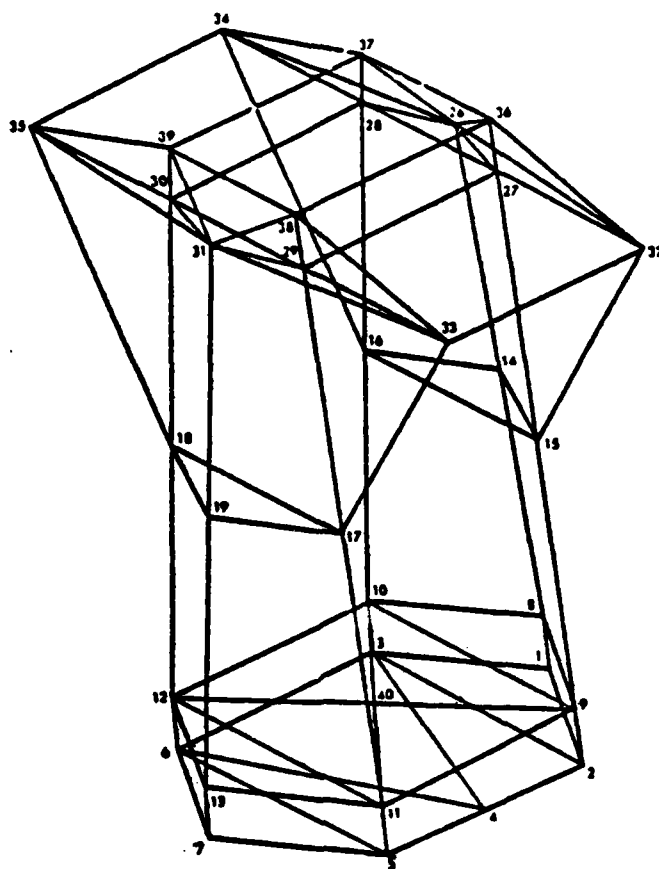


Figure 6. CSDL 2 Node Placement

LOS performance (15).

Eigenvalue analyses will provide control performance information. Line-of-sight performance, as well as defocus along the z-axis, are important analysis factors, but will not be directly addressed in the results of this investigation. Instead, the eigenvalue analysis will provide information on controller maintenance of modal stability and on controller independence (decoupling).

The controller development on which this study is based will now be presented.

TABLE I

CSDL 2 Node Coordinates

<u>Node</u>	<u>X(m)</u>	<u>Y(m)</u>	<u>Z(m)</u>	<u>Node</u>	<u>X(m)</u>	<u>Y(m)</u>	<u>Z(m)</u>
1	-7.0	0.0	0.0	37	-4.0	-3.0	24.0
2	-4.0	5.0	0.0	38	4.0	3.0	24.0
3	-4.0	-5.0	0.0	39	4.0	-3.0	24.0
4	0.0	5.0	0.0	40	0.0	2.5	2.0
5	4.0	5.0	0.0	42	0.0	5.0	-0.3
6	4.0	-5.0	0.0	43	-2.0	0.0	-1.3
7	7.0	0.0	0.0	44	0.0	-1.7	-1.3
8	-7.0	0.0	2.0	45	2.0	0.0	-1.3
9	-4.0	5.0	2.0	46	-4.0	-5.0	-0.3
10	-4.0	-5.0	2.0	47	4.0	-5.0	-0.3
11	4.0	5.0	2.0	48	-26.0	0.0	-1.3
12	4.0	-5.0	2.0	49	-21.0	0.0	-1.3
13	7.0	0.0	2.0	50	-16.0	0.0	-1.3
14	-6.0	0.0	12.0	51	-11.0	0.0	-1.3
15	-4.0	4.0	12.0	52	-6.0	0.0	-1.3
16	-4.0	-4.0	12.0	53	6.0	0.0	-1.3
17	4.0	4.0	12.0	54	11.0	0.0	-1.3
18	4.0	-4.0	12.0	55	16.0	0.0	-1.3
19	6.0	0.0	12.0	56	21.0	0.0	-1.3
26	-5.0	0.0	22.0	57	26.0	0.0	-1.3
27	-4.0	3.0	22.0	100	0.0	0.0	0.0
28	-4.0	-3.0	22.0	910	-4.0	-2.5	2.0
29	4.0	3.0	22.0	1001	0.0	-6.5	22.0
30	4.0	-3.0	22.0	1002	0.0	0.0	2.0
31	5.0	0.0	22.0	1003	0.0	6.5	22.0
32	-4.0	10.0	22.0	1004	0.0	4.0	2.0
33	4.0	10.0	22.0	1112	4.0	-2.5	2.0
34	-4.0	-10.0	22.0	2830	0.0	-3.0	22.0
35	4.0	-10.0	22.0	3233	0.0	10.0	22.0
35	-4.0	3.0	24.0				

TABLE II

CSDL 2 Lumped Mass Distribution

<u>Node</u>	<u>Mass (kg)</u>	<u>Node</u>	<u>Mass (kg)</u>
9	67.4	44	3500
10	67.4	48	81.9
11	67.4	50	1.638
12	67.4	52	73.8
27	69.5	53	73.8
28	6.74	55	163.8
29	69.5	57	81.9
30	6.74	1001	1000
32	6.74	1002	800
33	6.74	1003	1200
34	69.5	1004	600
35	69.5		

TABLE III

CSDL 2 Sensor/Actuator Locations and Orientations
(x, y, z in direction cosines)

<u>Pair</u>	<u>Node</u>	<u>x</u>	<u>y</u>	<u>z</u>
1	9	0	1	0
2	9	0	0	1
3	10	0	0	1
4	11	1	0	0
5	11	0	1	0
6	11	0	0	1
7	12	0	0	1
8	27	1	0	0
9	27	0	1	0
10	27	0	0	1
11	28	0	0	1
12	29	0	1	0
13	29	0	0	1
14	30	0	0	1
15	32	0	0	1
16	33	0	0	1
17	34	1	0	0
18	34	0	1	0
19	34	0	0	1
20	35	0	1	0
21	35	0	0	1

TABLE IV

Key Results of NASTRAN Eigenvalue Analysis on Nominal CSDL 2 Model

Mode	Generalized Mass	Generalized Stiffness	$\omega \frac{\text{rad}}{\text{sec}}$	$\Omega \frac{\text{rad}^2}{\text{sec}^2}$
1-6*	1.00E+00	0.0	0.0	0.0
7*	1.00E+00	5.128E-01	7.161E-01	5.128E-01
8	1.00E+00	8.521E-01	9.231E-01	8.521E-01
9	1.00E+00	8.835E-01	9.399E-01	8.835E-01
10	1.00E+00	1.212E+00	1.101E+00	1.212E+00
11	1.00E+00	8.819E+00	2.862E+00	8.189E+00
12*	1.00E+00	1.266E+01	3.502E+00	1.226E+01
13*	1.00E+00	1.403E+01	3.746E+00	1.403E+01
14	1.00E+00	1.492E+01	3.863E+00	1.492E+01
15	1.00E+00	1.599E+01	3.998E+00	1.599E+01
16	1.00E+00	1.625E+01	4.032E+00	1.625E+01
17*	1.00E+00	2.623E+01	5.122E+00	2.623E+01
18	1.00E+00	2.630E+01	5.128E+00	2.630E+01
19	1.00E+00	2.677E+01	5.174E+00	2.677E+01
20	1.00E+00	3.310E+01	5.753E+00	3.310E+01
21*	1.00E+00	3.730E+01	6.197E+00	3.730E+01
22*	1.00E+00	5.301E+01	7.281E+00	5.301E+01
23	1.00E+00	9.498E+01	9.746E+00	9.498E+01
24*	1.00E+00	1.241E+02	1.114E+01	1.241E+02
25	1.00E+00	1.999E+02	1.414E+01	1.999E+02
26	1.00E+00	2.001E+02	1.416E+01	2.001E+02
27	1.00E+00	4.654E+02	2.157E+01	4.654E+02
28*	1.00E+00	4.705E+02	2.169E+01	4.705E+02
29	1.00E+00	6.182E+02	2.468E+01	6.182E+02
30*	1.00E+00	6.275E+02	2.505E+01	6.275E+02
31	1.00E+00	6.481E+02	2.546E+01	6.481E+02
32	1.00E+00	7.428E+02	2.725E+01	7.428E+02
33	1.00E+00	1.700E+03	4.123E+01	1.700E+03
34	1.00E+00	2.568E+03	5.067E+01	2.568E+03
35	1.00E+00	2.821E+03	5.311E+01	2.821E+03
36	1.00E+00	3.095E+03	5.563E+01	3.095E+03

III. System Model

Equations of Motion

As presented by Calico and Miller (2,13), the system model may be developed from the vibrational equations of motion for a large space structure given generally as

$$M\ddot{\bar{g}} + E\dot{\bar{g}} + K\bar{g} = D\bar{u} \quad (1)$$

where M is a $n \times n$ symmetric mass matrix, E is a $n \times n$ symmetric damping matrix, K is a $n \times n$ symmetric stiffness matrix, D is a $n \times m$ matrix of nodal, attitude-evaluated actuator locations, $\bar{g}$ is a $n \times 1$ generalized coordinate vector, and $\bar{u}$ is a $m \times 1$ control input vector. Introducing the $n \times n$ modal matrix ϕ for Eq (1), such that

$$\bar{g} = \phi \bar{\eta} \quad (2)$$

where $\bar{\eta}$ is the n -vector of modal coordinates, Eq (1) may be written as

$$\begin{bmatrix} I \end{bmatrix} \ddot{\bar{\eta}} + \begin{bmatrix} 2\zeta\omega \end{bmatrix} \dot{\bar{\eta}} + \begin{bmatrix} \omega^2 \end{bmatrix} \bar{\eta} = \phi^T D \bar{u} \quad (3)$$

the ω_i and ζ_i terms being natural frequencies and damping coefficients, respectively, of the specific modes. The properties of the modal matrix ϕ are such that the coefficients of Eq (3) are given by

$$\begin{aligned} \begin{bmatrix} I \end{bmatrix} &= \phi^T M \phi \\ \begin{bmatrix} 2\zeta\omega \end{bmatrix} &= \phi^T E \phi \\ \begin{bmatrix} \omega^2 \end{bmatrix} &= \phi^T K \phi \end{aligned} \quad (4)$$

where

- $\underline{I}$ = nxn identity matrix
- $2\zeta\omega$ = nxn diagonal damping matrix
- ω^2 = nxn diagonal matrix of eigenvalues of Eq (1)

Equation (3) may now be converted into a state space representation of the system, given by

$$\dot{\bar{x}} = A\bar{x} + B\bar{u} \quad (5)$$

in which A is the nxn plant matrix, B is the nxm input matrix, $\bar{x}$ is the nx1 state vector, and $\bar{u}$ is the mx1 control input vector. These system parameters in state space are

$$A = \begin{bmatrix} 0 & \vdots & \underline{I} \\ \dots\dots\dots & & \\ -\omega^2 & \vdots & -2\zeta\omega \end{bmatrix}$$

$$B = \begin{bmatrix} 0 \\ \dots\dots\dots \\ \Phi^T D \end{bmatrix} \quad (6)$$

$$x = \begin{pmatrix} \bar{n} \\ \dot{\bar{n}} \end{pmatrix}$$

The complete state, however, is normally not available, so Eq (5) must be supplemented by an output equation. State space form gives the sensor output as

$$\bar{y} = c_p \bar{g} + c_v \dot{\bar{g}} \quad (7)$$

when both position (p subscript) and velocity (v subscript) sensors are used. Expressing Eq (7) with the state vector $\bar{x}$

$$\bar{y} = C\bar{x} \quad (8)$$

where

$$C = \begin{bmatrix} C_p \Phi & \vdots & C_v \Phi \end{bmatrix} \quad (9)$$

Equations (5) and (8) form the large space structure model available to the control designer. These equations will be further explained so they will hold more significance when being applied to modal control of flexible structures.

Control Model

The full structural model is represented by the $2n$ -dimensional state vector $\bar{x}$. As noted earlier, it is impossible to model all of the possible modes for a complex structure, and of those modeled, even fewer will be actively controlled. Assuming that multiple controllers are available, each controlling a small subset n_i of nodes, as in this investigation, the state vector may be simply represented by

$$\bar{x} = \left\{ \bar{x}_1^T, \bar{x}_2^T, \dots, \bar{x}_N^T, \bar{x}_r^T, \bar{x}_{um}^T \right\}^T \quad (10)$$

The x_i terms represent $2n_i$ -vectors of modal amplitudes and velocities as defined by the last of Eq (6) controlled by the i th controller of N controllers present. The $\bar{x}_r$ represents a $2n_r$ -vector of modeled residual modes and the $\bar{x}_{um}$ represents a $2n_{um}$ -vector of unmodeled modes.

As defined earlier, the unmodeled modes are those which exist but are beyond the number of modes in the structural model. These will no

longer appear in the derivations. The residual modes are those which are modeled but not controlled. The controlled modes are those which require active control in order to obtain satisfactory system response. The selection of the modes to be controlled and their assignment to one of the N controllers is left to the control designer.

It should be noted that so far suppressed modes have not directly been referred to even though they were defined earlier in the text. They have not been ignored, in fact they are included within the controlled modes, in the following manner: In a multiple controller design, the individual controller actively controls those modes assigned to it. But the modes assigned to the other controllers interact with this individual controller, causing control and/or observation spillover. In the process of controlling the system, each controller may contribute to the elimination of observation and control spillover in the system. Thus each controller, in effect "suppresses" the modes contained in the other controllers. In other words, the controlled modes of one controller are the suppressed modes of another controller. Therefore, the suppressed modes are contained implicitly within the controlled modes. So, like the unmodeled modes, the suppressed modes exist in the system, but will not be mentioned in the derivations since they are included implicitly in the controlled modes.

Continuing with the derivation, the notation of Eq (10) may be used to express the state equations as follows

$$\dot{\bar{x}}_i = A_i \bar{x}_i + B_i \bar{u}, \quad i = 1, 2, \dots, N \quad (11)$$

$$\dot{\bar{x}}_r = A_r \bar{x}_r + B_r \bar{u} \quad (12)$$

$$\bar{y} = \sum_{i=1}^N C_i \bar{x}_i + C_r \bar{x}_r \quad (13)$$

where the A, B, and C matrices are

$$A_j = \begin{bmatrix} 0 & \vdots & I \\ \dots & & \dots \\ -\omega_j^2 & \vdots & -2\zeta\omega_j \end{bmatrix} \quad j = 1, 2, \dots, N, r \quad (14)$$

$$B_j = \begin{bmatrix} 0 \\ \dots \\ \phi^T D_j \end{bmatrix} \quad j = 1, 2, \dots, N, r \quad (15)$$

$$C_j = \begin{bmatrix} C_{pj}\phi & \vdots & C_{vj}\phi \end{bmatrix} \quad j = 1, 2, \dots, N, r \quad (16)$$

Moreover, the lower partition of Eq (15), the $\phi^T D_i$ and $\phi^T D_r$ matrices, are of the form

$$\phi^T D_i = \psi_i \quad (17)$$

$$\phi^T D_r = \psi_r \quad (18)$$

where

$$(\psi_i)_{jk} = \bar{\phi}_j \cdot \bar{d}_{ik} \quad (19)$$

$$(\psi_r)_{jk} = \bar{\phi}_j \cdot \bar{d}_{rk} \quad (20)$$

The ϕ_j are the column vectors of the matrix ϕ and the d_{ik} and d_{rk} are the column vectors of the D_i and D_r matrices, respectively. Using the forms given in Eqs (17) to (20), the $\phi^T D_i$ and $\phi^T D_r$ matrices may be represented as

$$\Psi_i = \begin{bmatrix} (\psi_i)_{11} & (\psi_i)_{12} & \dots & (\psi_i)_{1n_a} \\ (\psi_i)_{21} & (\psi_i)_{22} & & (\psi_i)_{2n_a} \\ \vdots & & & \vdots \\ (\psi_i)_{n_i1} & (\psi_i)_{n_i2} & \dots & (\psi_i)_{n_i n_a} \end{bmatrix} \quad (21)$$

$$\Psi_r = \begin{bmatrix} (\psi_r)_{11} & (\psi_r)_{12} & \dots & (\psi_r)_{1n_a} \\ (\psi_r)_{21} & (\psi_r)_{22} & & (\psi_r)_{2n_a} \\ \vdots & & & \vdots \\ (\psi_r)_{n_r1} & (\psi_r)_{n_r2} & \dots & (\psi_r)_{n_r n_a} \end{bmatrix} \quad (22)$$

where n_i is the number of modes in the i th controller, n_r is the number of residual modes, and n_a is the number of actuators employed. This allows Eq (15) to be rewritten as

$$B_j = \begin{bmatrix} 0 \\ \dots \\ \psi_j \end{bmatrix} \quad j = 1, 2, \dots, N, r \quad (23)$$

In simpler terms, the rows of the ψ_j matrices represent the amplitude of each structural mode along the line of action of each actuator location.

The dimension of the ψ_j matrices is $n_j \times n_a$ making the dimension of the B_j matrices $2n_j \times n_a$ when the upper null partition is included. Likewise it can be seen that the $C_{pj}\Phi$ and $C_{vj}\Phi$ partitions of Eq (16) are of dimension $n_s \times n_j$ where n_s is the number of sensors employed. This makes the dimension of the C_j matrices $n_s \times 2n_j$.

Examining the C_j matrix more closely, the C_{pj} and C_{vj} terms are the position and velocity coefficient matrices, respectively, of the sensors employed, assuming that both position and velocity sensors are used. However, if only rate sensors are used, this makes the C_{pj} into zero matrices so that Eq (16) now becomes

$$C_j = \begin{bmatrix} 0 & \vdots & C_{vj}\phi \end{bmatrix} \quad j = 1, 2, \dots, N, r \quad (24)$$

Furthermore, when collocated sensor/actuators are employed with the same alignment, this simplifies even more. In this special case

$$C_{vj}\phi = \left[\phi^T D_j \right]^T \quad (25)$$

$$C_{vj}\phi = \left[\psi_j^T \right] \quad (26)$$

so that

$$C_j = \begin{bmatrix} 0 & \vdots & \psi_j^T \end{bmatrix} \quad j = 1, 2, \dots, N, r \quad (27)$$

This simplicity is the prime advantage of using either rate or position sensors only. As pointed out in the B_j matrix, the columns of the ψ_j^T matrix in C_j represent the rates of each structural mode at each sensor location along the line of the sensor.

The equations thus derived are very general in form and are independent of structural complexity. Only the matrix dimensions will vary depending on the number of sensors, actuators, and modes studied. This general development can lead one to understand the wide variety of structures to which the following analysis may be applied.

Modal Control

The controller design for N controllers will be based upon the model given by Eqs (11), (12), and (13) which, restated, are

$$\dot{\bar{x}}_i = A_i \bar{x}_i + B_i \bar{u} \quad (11)$$

$$\dot{\bar{x}}_r = A_r \bar{x}_r + B_r \bar{u} \quad (12)$$

$$\bar{y} = \sum_{i=1}^N C_i \bar{x}_i + C_r \bar{x}_r \quad (13)$$

The term $B_r \bar{u}$ in Eq (12) represents control spillover from the residual control matrix. Similarly, the term $C_r \bar{x}_r$ is called observation spillover. These spillovers can destabilize the residual modes. In decentralized control systems another set of spillovers appears. If the states are available to the control designer, the feedback controller may be designed such that

$$\bar{u} = \sum_{i=1}^N \hat{K}_i \bar{x}_i \quad (28)$$

where $\hat{K}_i$ are the control gain matrices. The development of the $\hat{K}_i$ matrices will be presented shortly. Now the residual state equation becomes

$$\dot{\bar{x}}_r = A_r \bar{x}_r + B_r \sum_{i=1}^N \hat{K}_i \bar{x}_i \quad (29)$$

which is spillover from the controlled modes to the residuals. The controlled state equation is

$$\dot{\bar{x}}_i = A_i \bar{x}_i + B_i \sum_{j=1}^N \hat{K}_j \bar{x}_j \quad (30)$$

and shows the control spillover from controller to controller.

Now for direct output feedback the control vector is

$$\bar{u} = \sum_{i=1}^N \hat{K}_i \bar{y} \quad (31)$$

$$\bar{u} = \sum_{i=1}^N \hat{K}_i \left(\sum_{j=1}^N C_j \bar{x}_j + C_r \bar{x}_r \right) \quad (32)$$

The closed loop state equation becomes

$$\dot{\bar{x}}_i = A_i \bar{x}_i + B_i \sum_{j=1}^N \hat{K}_j \left(\sum_{k=1}^N C_k \bar{x}_k + C_r \bar{x}_r \right) \quad (33)$$

and in matrix form for the N controller with residuals

$$\dot{\bar{x}} = \begin{bmatrix} A_1 + B_1 \kappa C_1 & B_1 \kappa C_2 & \cdots & B_1 \kappa C_N & B_1 \kappa C_r \\ B_2 \kappa C_1 & A_2 + B_2 \kappa C_2 & \cdots & B_2 \kappa C_N & B_2 \kappa C_r \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ B_N \kappa C_1 & B_N \kappa C_2 & \cdots & A_N + B_N \kappa C_N & B_N \kappa C_r \\ B_r \kappa C_1 & B_r \kappa C_2 & \cdots & B_r \kappa C_N & A_r + B_r \kappa C_r \end{bmatrix} \bar{x} \quad (34)$$

where $\kappa = \sum_{i=1}^N (\hat{K}_i)$. The last column in this plant matrix represents observation spillover; the last row, control spillover. The upper off-diagonal terms are the controller to controller observation spillover. The lower off-diagonal terms the controller to controller control spillover terms. To assure stability of the system based on the stability in each controller, the system matrix must be within a similarity transformation of either upper or lower block triangular. Lower block triangular form requires

$$B_i \hat{K}_j = 0 \text{ and } \hat{K}_i C_j = 0 \quad i = 1, 2, \dots, N-1; j = i + 1, \dots, N \quad (35)$$

and upper block triangular form requires

$$B_i \hat{K}_j = 0 \text{ and } \hat{K}_i C_j = 0 \quad i = j + 1, \dots, N; j = 1, 2, \dots, N-1 \quad (36)$$

These conditions are identical to those required by the decentralized optimal controller as found by Calico and Miller (2).

The block structure of Eq (34) can be more easily seen in a specific example. The three controller case will now be examined.

Three Controllers

Setting N equal to 3 and following the form given in the previous development for N controllers, the state equations for a three controller system are given as

$$\dot{\bar{x}}_1 = A_1 \bar{x}_1 + B_1 \bar{u} \quad (37)$$

$$\dot{\bar{x}}_2 = A_2 \bar{x}_2 + B_2 \bar{u} \quad (38)$$

$$\dot{\bar{x}}_3 = A_3 \bar{x}_3 + B_3 \bar{u} \quad (39)$$

$$\dot{\bar{x}}_r = A_r \bar{x}_r + B_r \bar{u} \quad (40)$$

The control applied is

$$\bar{u} = \sum_{i=1}^3 \hat{K}_i \bar{y} = \sum_{i=1}^3 \hat{K}_i \left(\sum_{m=1}^3 C_m \bar{x}_m + C_r \bar{x}_r \right) \quad (41)$$

$$\bar{u} = (\hat{K}_1 + \hat{K}_2 + \hat{K}_3)(C_1 \bar{x}_1 + C_2 \bar{x}_2 + C_3 \bar{x}_3 + C_r \bar{x}_r) \quad (42)$$

Substituting Eq (42) into (37) through (40), the system equations are

$$\begin{aligned} \dot{\bar{x}}_1 &= A_1 \bar{x}_1 + B_1 (\hat{K}_1 + \hat{K}_2 + \hat{K}_3)(C_1 \bar{x}_1 + C_2 \bar{x}_2 + C_3 \bar{x}_3 + C_r \bar{x}_r) \\ &= (A_1 + B_1 \hat{K}_1 C_1) \bar{x}_1 + B_1 \hat{K}_1 C_2 \bar{x}_2 + B_1 \hat{K}_1 C_3 \bar{x}_3 + B_1 \hat{K}_1 C_r \bar{x}_r \end{aligned} \quad (43)$$

$$\begin{aligned}\dot{\bar{x}}_2 &= A_2\bar{x}_2 + B_2(\hat{K}_1 + \hat{K}_2 + \hat{K}_3)(C_1\bar{x}_1 + C_2\bar{x}_2 + C_3\bar{x}_3 + C_r\bar{x}_r) \\ &= B_2KC_1\bar{x}_1 + (A_2 + B_2KC_2)\bar{x}_2 + B_2KC_3\bar{x}_3 + B_2KC_r\bar{x}_r\end{aligned}\quad (44)$$

$$\begin{aligned}\dot{\bar{x}}_3 &= A_3\bar{x}_3 + B_3(\hat{K}_1 + \hat{K}_2 + \hat{K}_3)(C_1\bar{x}_1 + C_2\bar{x}_2 + C_3\bar{x}_3 + C_r\bar{x}_r) \\ &= B_3KC_1\bar{x}_1 + B_3KC_2\bar{x}_2 + (A_3 + B_3KC_3)\bar{x}_3 + B_3KC_r\bar{x}_r\end{aligned}\quad (45)$$

$$\begin{aligned}\dot{\bar{x}}_r &= A_r\bar{x}_r + B_r(\hat{K}_1 + \hat{K}_2 + \hat{K}_3)(C_1\bar{x}_1 + C_2\bar{x}_2 + C_3\bar{x}_3 + C_r\bar{x}_r) \\ &= B_rKC_1\bar{x}_1 + B_rKC_2\bar{x}_2 + B_rKC_3\bar{x}_3 + (A_r + B_rKC_r)\bar{x}_r\end{aligned}\quad (46)$$

Now combining these system equations with an augmented state vector $\bar{x}$

$$\bar{x} = \{ \bar{x}_1^T, \bar{x}_2^T, \bar{x}_3^T, \bar{x}_r^T \}^T \quad (47)$$

the closed loop system equation may be given as

$$\dot{\bar{x}} = \begin{bmatrix} A_1 + B_1KC_1 & B_1KC_2 & B_1KC_3 & B_1KC_r \\ B_2KC_1 & A_2 + B_2KC_2 & B_2KC_3 & B_2KC_r \\ B_3KC_1 & B_3KC_2 & A_3 + B_3KC_3 & B_3KC_r \\ B_rKC_1 & B_rKC_2 & B_rKC_3 & A_r + B_rKC_r \end{bmatrix} \bar{x} \quad (48)$$

Inspection of Eq (48) shows that individual controller stability cannot guarantee overall system stability. Block triangularization may be used to remove controller to controller spillover, but the spillover from the residuals will remain. Triangularization requires the satisfaction of Eq (35) conditions. These conditions may partially be met by using selective sensor and actuator placements, but in general their satisfaction requires careful consideration.

The approach, used in this study, proposes that transformation matrices T_j and Γ_i be found such that $B_i T_j = 0$ and $\Gamma_i C_j = 0$, $j > i$ for lower block triangular form, $j < i$ for upper block triangular form. It will be shown shortly how the T_j and Γ_i may be used in eliminating destabilizing spillovers.

Design Feedback Gain

The suboptimal output feedback gains for this study are obtained using an extension of the Kosut approximation (18). The state, output, and control equations are

$$\dot{\bar{x}} = A\bar{x} + B\bar{u} \quad (49)$$

$$\bar{y} = C\bar{x} \quad (50)$$

$$\bar{u} = \hat{K}\bar{y} = \hat{K}C\bar{x} \quad (51)$$

The direct output feedback allows us to rewrite Eq (49) as

$$\dot{\bar{x}} = (A + B\hat{K}C)\bar{x} \quad (52)$$

It is desirable for the closed loop system matrix, $A + B\hat{K}C$, to equal the closed loop system matrix whose control gain matrix is the optimal steady full-state gain, denoted as G . This gain, G , is determined using optimal control (3). The control gain matrix G is derived by defining a quadratic performance index J such that

$$J = \frac{1}{2} \int_0^{\infty} (\bar{x}^T Q \bar{x} + \bar{u}^T R \bar{u}) dt \quad (53)$$

where Q is an $n \times n$ positive semidefinite weighting matrix and R is an $m \times m$ positive semidefinite weighting matrix. It is desired to minimize this index subject to Eq (11). Then the optimal solution to the minimization

problem is

$$G = -R^{-1}B^T S \quad (54)$$

where S is the solution to the steady state matrix Riccati equation

$$SA + A^T S - SBR^{-1}B^T S + Q = 0 \quad (55)$$

The control applied with optimal feedback gains becomes

$$\bar{u} = G\bar{x} \quad (56)$$

for which the closed loop system is

$$\dot{\bar{x}} = (A + BG)\bar{x} \quad (57)$$

The two systems are equated and the suboptimal gain matrix $\hat{K}$ is solved for

$$(A + B\hat{K}C)\bar{x} = (A + BG)\bar{x} \quad (58)$$

$$A + B\hat{K}C = A + BG \quad (59)$$

$$B\hat{K}C = BG \quad (60)$$

$$\hat{K}C = G \quad (61)$$

In general, the output matrix is not square and at this point the Kosut approximation solves for $\hat{K}$ using a pseudo-inverse method

$$\hat{K}CC^T = GC^T \quad (62)$$

$$\hat{K}(CC^T)(CC^T)^{-1} = GC^T(CC^T)^{-1} \quad (63)$$

$$\hat{K} = GC^T(CC^T)^{-1} \quad (64)$$

This solution is valid if the output matrix is of full rank making CC^T non-singular and invertible. For the case where C is not of full rank, $\hat{K}$ can

still be found by using the inverse of the singular value decomposition of C , denoted as C^+ (22). This inverse is also known as the generalized or Penrose inverse. The method of singular value decomposition is discussed in the next chapter. Armed with this generalized inverse and solving for $\hat{K}$ from Eq (61)

$$\hat{K}CC^+ = GC^+ \quad (65)$$

$$\hat{K} = GC^+ \quad (66)$$

This relatively simple procedure allows us to determine the suboptimal gains for an output matrix of any rank.

These equations can be generalized for decentralized controllers and allow us to complete the development of the transformed DOFB.

Transformed DOFB

Consider an output transformed by Γ_i

$$\bar{w}_i = \Gamma_i \bar{y}_i \quad (67)$$

and a control law

$$\bar{u} = \sum_{i=1}^N T_i \bar{v}_i \quad (68)$$

where $\bar{v}_i = K^* \bar{w}_i$. When Γ_i and T_i are such that the conditions $B_i T_j = 0$ and $\Gamma_i C_j = 0$ for $j > i$ or $i < j$ are met, destabilizing controller spillover is eliminated. Either of these conditions will lead to a closed loop state system where the controller matrices on the diagonal are decoupled and are the result of solving the following open control problems (6):

$$\dot{\bar{x}}_i = A \bar{x}_i + B_i^* \bar{v}_i \quad (69)$$

$$\bar{w}_i = \Gamma_i \bar{y} \quad (70)$$

$$B_i^* = B_i T_i \quad (71)$$

$$C_i^* = \Gamma_i C_i \quad (72)$$

$$\bar{v}_i = K_i^* \bar{w}_i \quad (73)$$

The gain K_i^* is obtained by solving for a transformed optimal gain G_i^* . G_i^* is found by substituting B_i^* into Eqs (54) and (55) and finding the generalized inverse of C_i^* . Hence, the transformed output gain is

$$K^* = G^* C^{*+} \quad (74)$$

the design feedback control gain matrix $\tilde{K}$ is given by

$$\tilde{K}_i = T_i K_i^* \Gamma_i \quad (75)$$

and the closed loop equation becomes

$$\dot{\bar{x}}_i = A_i + B_i \left(\sum_{j=1}^N \tilde{K}_j \right) C_i \quad (76)$$

In matrix form Eq (76) is

$$\dot{\bar{x}} = \begin{bmatrix} A_1 + B_1 \tilde{K}_1 C_1 & 0 & 0 & \dots & 0 & B_1 \Delta C_r \\ B_2 \tilde{K}_1 C_1 & A_2 + B_2 \tilde{K}_2 C_2 & 0 & \dots & 0 & B_2 \Delta C_r \\ B_3 (\tilde{K}_1 + \tilde{K}_2) C_1 & B_3 \tilde{K}_2 C_2 & \ddots & & \vdots & \vdots \\ \vdots & \vdots & & & 0 & \vdots \\ B_N \left(\sum_{j=1}^{N-1} \tilde{K}_j \right) C_1 & B_N \left(\sum_{j=1}^{N-2} \tilde{K}_j \right) C_2 & \dots & A_N + B_N \tilde{K}_N C_N & B_N \Delta C_r \\ B_r \Delta C_1 & B_r \Delta C_2 & \dots & B_r \Delta C_N & A_r + B_r \Delta C_r \end{bmatrix} \bar{x} \quad (77)$$

where $\Lambda = \sum_{i=1}^N \tilde{K}_i$. For a three controller system with residuals, which this study implements, Eq (77) becomes

$$\dot{\bar{x}} = \begin{bmatrix} A_1 + B_1 \tilde{K}_1 C_1 & 0 & 0 & B_1 \Lambda C_r \\ B_2 \tilde{K}_1 C_1 & A_2 + B_2 \tilde{K}_2 C_2 & 0 & B_2 \Lambda C_r \\ B_3 (\tilde{K}_1 + \tilde{K}_2) C_1 & B_3 \tilde{K}_2 C_2 & A_3 + B_3 \tilde{K}_3 C_3 & B_3 \Lambda C_r \\ B_r \Lambda C_1 & B_r \Lambda C_2 & B_r \Lambda C_3 & A_r + B_r \Lambda C_r \end{bmatrix} \bar{x} \quad (78)$$

A four controller system without residuals is also examined and its closed loop system equation with design gains should look like

$$\dot{\bar{x}} = \begin{bmatrix} A_1 + B_1 \tilde{K}_1 C_1 & 0 & 0 & 0 \\ B_2 \tilde{K}_1 C_1 & A_2 + B_2 \tilde{K}_2 C_2 & 0 & 0 \\ B_3 (\tilde{K}_1 + \tilde{K}_2) C_1 & B_3 \tilde{K}_2 C_2 & A_3 + B_3 \tilde{K}_3 C_3 & 0 \\ B_4 (\tilde{K}_1 + \tilde{K}_2 + \tilde{K}_3) C_1 & B_4 (\tilde{K}_2 + \tilde{K}_3) C_2 & B_4 \tilde{K}_3 C_3 & A_4 + B_4 \tilde{K}_4 C_4 \end{bmatrix} \bar{x} \quad (79)$$

It is quite obvious from comparing Eqs (78) and (79) that the residual terms can have a significant effect on system eigenvalues and hence system stability, while the system eigenvalues of Eq (79) will always be those of the individual controllers along the diagonal. With the control developed, we need to establish how many modes may be controlled in a given controller, i.e. define the sensor/actuator requirements.

Sensor/Actuator Requirements

As mentioned for three controllers, in order to perform spillover suppression, one or more gain matrices must be made orthogonal to N-1 B or C matrices. For example, from Eq (35), to satisfy the expression for $B_i \hat{K}_j$

the columns of $\hat{K}_1$ must be simultaneously orthogonal to the rows of B_2 through B_N . In other words, the columns of $\hat{K}_1$ must be in the null space of the matrix B_{2N} , where B_{2N} is defined as

$$B_{2N} = \begin{bmatrix} B_2 \\ \dots \\ B_3 \\ \dots \\ B_N \end{bmatrix} \quad (80)$$

The null space of B_{2N} has dimension P_{2N} given as

$$P_{2N} = (n_a - r_{2N}) \quad (81)$$

where

n_a = number of actuators

r_{2N} = rank of $B_{2N} \leq \min (n_2 + n_3 + \dots + n_N, n_a)$

Therefore, $\hat{K}_1$ has P_{2N} columns.

The number of actuators must exceed the rank of B_{2N} in order for $B_{2N}\hat{K}_1 = 0$. Otherwise the system is overspecified and no transformation matrix exists which will drive the $B_{2N}\hat{K}_1$ to zero. If rows of B_{2N} are linearly independent, then the number of actuators needed is given as

$$n_a > \sum_{i=2}^{N-1} n_i \quad (82)$$

and if the rows are not linearly independent, $n_a > r_{2N}$. It can be seen that the other control gain matrices will have a sufficient number of actuators if the inequality in Eq (82) is met. A similar study shows that the number

of sensors needed is, for C_{2N} with linearly independent columns

$$n_s > \sum_{i=1}^{N-1} n_i \quad (83)$$

and for columns that are not linearly independent, $n_s > r_{2N}$ here is the rank of C_{2N} .

Likewise, for the lower block triangular conditions given in Eq (51), and actuator and sensor requirements can be shown to be

$$n_a > \sum_{i=1}^{N-1} n_i \quad (84)$$

$$n_s > \sum_{i=2}^N n_i \quad (85)$$

for linearly independent rows and columns of the B_{2N} and C_{2N} matrices, respectively.

It should be pointed out that satisfying the inequalities given in Eqs (82) through (85) may actually require more actuators and sensors than indicated. As an example, consider a thirty mode model. Using three controllers, each with ten modes, the above inequalities require at least twenty sensors and twenty actuators to assure decoupled system stability. However, to control and observe the system, at least one more sensor and actuator would be required. These conditions must be met in order to generate the transformation matrices and to implement the controllers. With these requirements and control model developed, the technique to obtain the transformation matrices T_i and Γ_i are presented.

IV. Transformation Technique

The transformation technique used in this study is the same one applied by Aldridge and Miller (1,13). It has been mentioned several times that the closed loop state equations, Eqs (11) and (33), will be put into block triangular form by the selective elimination of control and observation spillover terms. However, the exact details of this spillover elimination have been neglected until now. The following will describe the generation of the transformation matrices, which were referred to specifically as T and Γ in the previous section. The T matrix is a transformation matrix for the control spillover and the Γ matrix is a transformation matrix for the observation spillover.

In a single controller case, it can be seen that the spillover terms which, when eliminated, will assure system stability are $B_s K$ or $K C_s$, where the s subscript designates modes to be suppressed. An immediately obvious solution to this is $K = 0$. But this solution will make the respective terms $B_c K$ and $K C_c$ equal to zero also. Therefore, this solution is unsatisfactory. The transformation method generates a solution which for a single controller, is subject to the conditions

$$B_s K = 0 \quad (86)$$

$$K C_s = 0 \quad (87)$$

while maintaining

$$B_c K \neq 0 \quad (88)$$

$$K C_c \neq 0 \quad (89)$$

It would also be desirable to apply the additional constraint to the residual modes:

$$B_r K = 0 \quad (90)$$

$$K C_r = 0 \quad (91)$$

however, due to the large number of structural modes present in the model, this constraint is not realistic and will be ignored in this development. The effects of the residual spillovers may be minimized by the careful selection of modes designated as residual or suppressed, so as to create a frequency separation between the residuals and the bandwidth of the controller.

For a multiple controller, the conditions given in Eqs (86) and (87) apply, but now the B_s and C_s matrices may take on the form illustrated by Eq (80) and will be referred to as the B_{iN} and C_{iN} matrices. Instead of discussing all of the possible combinations of B_{iN} and C_{iN} for N controllers, take as an example the first condition given in Eq (35), that is

$$B_i K_j = 0 \quad j = 1, 2, \dots, N-1; \quad i = j+1, \dots, N \quad (92)$$

Given N controllers, K_i will have to be made orthogonal to $N-1$ B_i matrices. The $N-1$ B_i matrices may be combined into a single matrix such that

$$B_{2N} = \begin{bmatrix} B_2 \\ \dots\dots\dots \\ B_3 \\ \dots\dots\dots \\ \vdots \\ \dots\dots\dots \\ B_N \end{bmatrix} \quad (93)$$

Therefore, one of the conditions to be met is $B_{2N}K_1 = 0$. In other words, the K_1 matrix must be transformed such that its columns are orthogonal to the rows of B_{2N} , or as seen in the previous section, K_1 must be in the null space of B_{2N} . This is the most difficult case for N controllers and is presented only to describe how the multiple matrix is set up. The remainder of the derivation will be in terms of a generic B_s matrix which represents B_{2N} and B_2 alike.

The transformation matrix sought will be referred to as T and will be such that

$$B_s T = 0 \quad (94)$$

B_s has the row dimension of n_m (the number of modes to be suppressed) and the column dimension of n_a (the number of actuators). T , therefore, has dimensions of n_a by $n_a - n_m$. If there are fewer actuators than linearly independent modes, then no solution matrix T exists. The system is over-specified in this case, meaning there are more equations than unknowns. If the number of modes and actuators are equivalent, then the system is stable but uncontrollable (or in the case of the KC matrices, the system is stable but unobservable assuming an equal number of sensors and actuators). The actuators (sensors) are saturated with maintaining stability alone. Simply stated, the conditions given in Eq (82) and (83) must be met in order to generate a transformation matrix.

To illustrate the result of applying the transformation matrix described above, consider the following system of controlled and suppressed modes:

$$\dot{\bar{x}}_c = A_c \bar{x}_c + B_c \bar{u} \quad (95)$$

$$\dot{\bar{x}}_S = A_S \bar{x}_S + B_S \bar{u} \quad (96)$$

where

$$\bar{u} = G_C \bar{x}_C \quad (97)$$

The $B_S \bar{u}$ term of Eq (91) is a control spillover term which may be adversely affected by the control applied to Eq (96). The elimination of this term requires the use of a transformation matrix T such that

$$B_S T = 0 \quad (98)$$

while maintaining

$$B_C T \neq 0 \quad (99)$$

This transformation matrix may be used to define a new control $\bar{v}$ as

$$\bar{u} = T \bar{v} \quad (100)$$

Inserting this expression into the state equations given in Eqs (91) and (92) yields

$$\dot{\bar{x}}_C = A_C \bar{x}_C + B_C T \bar{v} \quad (101)$$

$$\dot{\bar{x}}_S = A_S \bar{x}_S + B_S T \bar{v} \quad (102)$$

Letting $B_C T = B^*$ and knowing that $B_S T = 0$ the new system is described by

$$\dot{\bar{x}}_C = A_C \bar{x}_C + B^* \bar{v} \quad (103)$$

$$\dot{\bar{x}}_S = A_S \bar{x}_S \quad (104)$$

in which no controller spillover exists. The new control vector will be shown to be

$$\bar{v} = G^* \bar{x}_c \quad (105)$$

With this general overview of the transformation process and its results, the development of the matrix T will now be discussed.

The major tool used obtaining this result is called the Singular Value Decomposition (22). The matrix to be decomposed is B_s which has dimensions $n_m \times n_a$ and can be described by

$$B_s = W \Sigma V^T \quad (106)$$

where W is an $(n_m \times n_m)$ orthogonal matrix of left singular vectors, V is an $(n_a \times n_a)$ orthogonal matrix of right singular vectors, and Σ is an $(n_m \times n_a)$ matrix with the s singular values of B_s in the first s entries along the main diagonal and zeroes in all other positions:

$$\Sigma = \begin{bmatrix} s & \vdots & 0 \\ \cdots & \vdots & \cdots \\ 0 & \vdots & 0 \end{bmatrix} \quad (107)$$

$n_m \times n_a$

such that

$$S = \begin{bmatrix} \sigma_1 & 0 & \cdots & 0 \\ 0 & \sigma_2 & & \\ \vdots & & \ddots & \\ 0 & & & \sigma_s \end{bmatrix} \quad (108)$$

The total number of singular values present is equal to the rank of the B_s matrix, and they are all non-negative. Assuming B_s is of full rank, the dimension of $S = \min(n_a, n_m)$.

By partitioning, the W matrix can be defined by

$$W = \begin{bmatrix} W_s & \vdots & W_r \end{bmatrix} \quad (109)$$

where W_s is an $n_m \times s$ matrix of left singular vectors associated with the non-zero singular values, W_r is an $n_m \times r$ matrix of left singular vectors associated with the zero singular values, and

$$s + r = n_m \quad (110)$$

Similarly, by partitioning, the V matrix can be defined by

$$V = \begin{bmatrix} V_s & \vdots & V_p \end{bmatrix} \quad (111)$$

where V_s is an $s \times n_a$ matrix of right singular vectors associated with the non-zero singular values, V_p is a $p \times n_a$ matrix of right singular vectors associated with the zero singular values, and

$$s + p = n_a \quad (112)$$

Remembering the V matrix is orthogonal and noting then that

$$V_s^T V_p = 0 \quad (113)$$

the decomposed matrix may be written as

$$B_s^T = W_s^T S V_s^T V_p = B_s V_p = 0 \quad (114)$$

which leads to the conclusion that the transformation matrix desired is given in the matrix of right singular values associated with the zero singular values:

$$T = V_p \quad (115)$$

where $T \neq 0$.

Once the transformation matrix is found, implementation is relatively simple. Equation (100) defined a $\bar{v}_i$ as the new control input vector. This is now given as

$$\bar{v}_i = G_i^* \bar{x}_i \quad (116)$$

The G_i^* matrix is found in the same manner as in the Modal Control subsection of Section III in which

$$G_i^* = -R_i^{*-1} B_i^{*T} S_i \quad (117)$$

where

$$R_i^* = T_i^T R_i T_i \quad (118)$$

R_i is the positive definite weighting matrix in Eq (53)

$$B_i^* = B_i T_i \quad (119)$$

and S_i is the solution to the matrix Riccati equation

$$S_i A_i + A_i^T S_i - S_i B_i^* R_i^{*-1} B_i^{*T} S_i + Q_i = 0 \quad (120)$$

Simple manipulation of Eq (117) will show that the transformed gain matrix is finally given by

$$G_i^* = T_i G_i \quad (121)$$

Substitution of this back into the state equations yields a closed loop system which is block triangular with no control spillover.

This technique may be paralleled to obtain a Γ transformation matrix for the observation gain to eliminate observation spillover. Substituting C_s^T for B_s , and Γ for T will give the same results with Γ equal to V_p . Otherwise, a new derivation using C_s will give the result that the observation transformation matrix Γ is equal to the transpose of the matrix W_r of left singular vectors associated with the zero singular values of C_s . Here, as alluded to earlier, the number of sensors must be greater than the number of modes to be suppressed.

V. Computer Model

Simplicity and flexibility were considered in the development of the computer program. The general format given by Aldridge (1) was followed as it provided a simple progression of logic. The program was modified to calculate the suboptimal gains and accommodate a design model of 33 modes (66 states). A program flow chart is shown in Figure 7.

The program provides decoupled control by reducing the closed loop state equation matrix, given in form by Eq (34), to the lower block triangular form. It is unnecessary to form the upper block system since the eigenvalue results for both systems are identical. In fact, by renumbering the controllers in the spillover elimination portion of this program, the opposite transformation is achieved. For this reason, only a main program for the lower block triangular controlled system is presented. This is listed in Appendix A. The subroutines which support this program are also in Appendix A. Several other subroutines are called, but not listed. These are provided by the International Mathematical and Statistical Library (IMSL).

Since program flexibility is desired, once the modal data is read, the program is designed to make any number of runs with different parameters for each run. The parameters that may be varied by the operator include: using a three or a four controller system, which modes are assigned to each controller, what control and observer weighting values to assign to each mode, and what initial system damping ratio is applied.

The program may read the input data from initialization assignments within the program or from a permanent data file. In either case, the

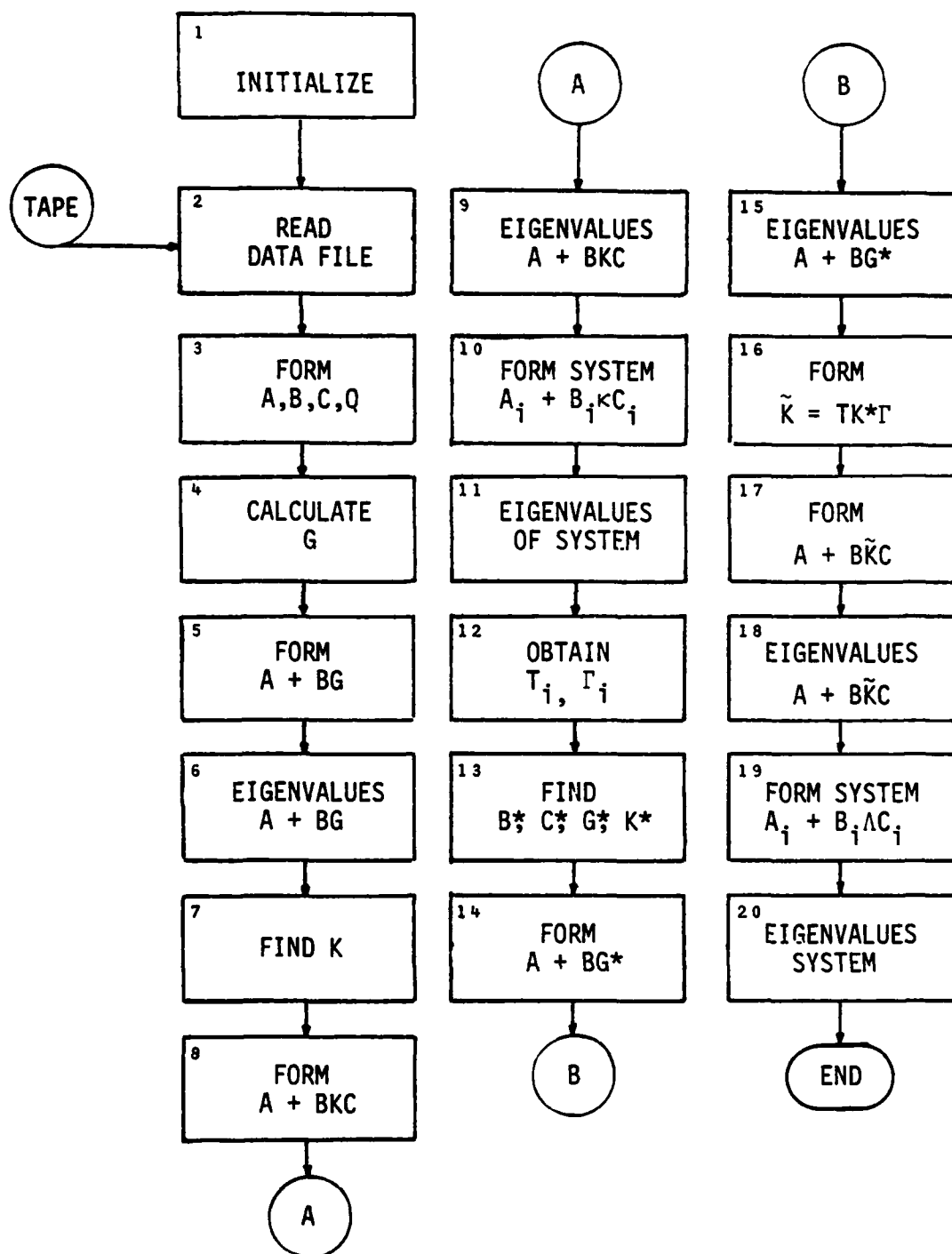


Figure 7. Computer Program Flow Chart

program operates as if it were interactive by prompting for input and then echoing the data read in. This makes the output very easy to interpret by allowing the user to trace the computer's progress through the execution of the program.

The program is initialized by inputting the number of controllers desired and then the number of modes in each controller. If three controllers are used, the number of residual modes will be requested, otherwise the fourth controller system is run without residuals. Next, the number of actuators and sensors are input, along with the damping ratio ζ . CSDL 2 was tested with twenty-one actuators, twenty-one sensors, and a damping ratio of 0.01 for each mode (15). The program will then read from a permanent file the matrix $(\Phi^T D)$ of modal data at each actuator location, followed by the transpose of the matrix of modal data at each sensor location. In this study, these two matrices are identical since colocated pairs of actuators and sensors are employed. The sensor modal matrix is input in transposed form so that the matrix for a colocated system may be copied directly from the actuator modal data matrix. However, these are left as separate entries in the event the actuators and sensors are not colocated. Finally, the modal frequencies are read in from the data file. After this preload of data, the desired run is made by specifying which modes are to be controlled by each controller and which modes are to be left as residuals, along with the desired control and observer weighting values for each mode. Different configuration runs may be made changing mode assignments, weighting values or controller configurations.

Program execution actually begins with the formation of the A, B, C, and weighting (Q) matrices for each controller. This is conveniently done by subroutines which read the required data for the modes specified. These

subroutines allow the operator to change the size of the controllers as needed simply by specifying the number of modes to be placed in each controller.

Once the initial matrices are formed, the optimal control gain matrices, G_i , are determined using a series of subroutines which generate a numerical solution to the matrix Riccati equation. These sophisticated routines were created by Kleinman (7) and so are known as the Kleinman routines. The G_i matrices, along with the parameter matrices, A , B , and C , are then combined to form the optimal closed loop system matrix. This particular program develops the three controller system with residuals, but does not include residual terms in the four controller systems.

Eigenvalue analysis of the individual optimal controllers is performed next, making use of the ISML routine EIGRF which determines the eigenvalues of real non-symmetric matrices. These eigenvalues are the desired values of the individual controllers and are used to compare the suboptimal controllers eigenvalues against. The optimal gains and generalized inverses of the output matrices are used to calculate the single controller DOFB gains, $\hat{K}$. These suboptimal gains are then used to form closed loop single controller systems whose eigenvalues are compared with the desired optimal values. The multiple controller system is formed with the summed gains, κ , and the system eigenvalues are calculated to evaluate coupling and use for comparison to the transformed system. Next the decoupling transformation matrices, T and Γ , are obtained and used to find B^* , C^* , G^* , K^* , and to form the control design gains, $\tilde{K}$.

Spillover elimination is the next step in the control algorithm. This varies with controller configuration. The 3 or 4 controller variation is accommodated by the program. The spillover elimination is a rather

lengthy portion of the program. The modes to be suppressed are formed into non-zero B_s and C_s^T matrices of the form given in Eq (80). The IMSL routine LSVDF is used to perform a singular value decomposition on these matrices. By using the left singular vectors associated with the zero singular values of B_s and C_s^T , the transformation matrices T_i and Γ_i are formed. The program then applies the transformation matrices and as discussed in Section IV to create new matrices, B_i^* , C_i^* , G_i^* , K_i^* , and $\tilde{K}_i$. The new closed loop systems are generated and the eigenvalue analysis is repeated.

This program demonstrates the flexibility of the control method applied. The only change that needs to be made to adapt the method for another structure is the basic matrix dimensioning in the program. Nothing else has to be altered, as long as the system model can be defined by

$$\dot{\bar{x}} = A\bar{x} + B\bar{u} \quad (5)$$

and

$$\bar{y} = C\bar{x} \quad (8)$$

as discussed in Section III. Thus, the ease of application of the control method described can be seen. Now, the performance of control method will be examined using the programs' eigenvalue analysis results.

VI. Investigation

Applying the decentralized control method to a large design model is a complicated task, so a systematic, building block approach was used to conduct the study. It was necessary to establish the method of feedback gain calculation, examine sensor information configurations (rate vs. position), and verify the results on a "perfect" (no residuals) model. The results are compared against those for steady full-state feedback optimal control. Once this foundation was laid, the model could be expanded to the multiple controller case with and without residual modes.

In the initial phase, a single controller was developed to establish the method of feedback gain calculation. The controller was assigned four flexible modes for this analysis. The four modes were arbitrarily selected from the twelve modes considered critical in the ACOSS Five report (15), which are identified in Table IV. The control weighting matrix, Q , was fixed at 1000 I. This program was modified to allow four configurations of the output matrix: position sensors only, $C = [C_{pj}\phi \vdots 0]$, rate sensors only, $C = [0 \vdots C_{vj}\phi]$, a mix of position and rate sensors, $C = [C_{p1}\phi \ C_{p2}\phi \ 0 \ 0 \vdots 0 \ 0 \ C_{v3}\phi \ C_{v4}\phi]$, and both position and rate sensors. These single controller responses were compared with those for the desired optimal controllers and to each other.

Next, the system was expanded to the multiple controller case to actively control twelve modes. No residuals were included in this phase of the study. The twelve modes selected are the same twelve identified in Table IV. Three and four controller cases were examined using low control weighting for the rigid body modes and higher (5,000 to 50,000) weighting

for the flexible modes as suggested by Aldridge (1). These controllers were operated using either position sensors only or rate sensors only. Mode assignments were rearranged to study mode grouping, and control weighting was varied to see the effect this had on mode controllability.

Finally the model was run using 33 modes (4-36) in three controllers with residuals and four controllers without residuals. For this phase, only rate sensor data was used. In this expanded model, modal selection and grouping were examined as well as the effects of biased weighting of lower frequency modes. The results of this outline will be presented shortly.

One of the more difficult tasks in controlling a large space structure is the determination of which structural modes are to be actively controlled and which are left as residuals. Factors which affect this selection process include sensor/actuator placement and alignment, controller bandwidths, and control constraints such as line-of-sight tolerances.

For multiple controllers, an additional step has to be taken. This is the assignment of the modes to be controlled to minimize the control effort. Organizing modes into compatible groups can be done by a simple examination of the angles between the vectors of modal amplitudes (the angles between the rows of the $\Phi^T D$ matrix). Defining these vectors as $\bar{\psi}_i$, the angles may be found from the equation for the dot product of two vectors, given by

$$\bar{\psi}_i \cdot \bar{\psi}_j = |\bar{\psi}_i| |\bar{\psi}_j| \cos \theta_{ij} \quad (123)$$

and

$$\theta_{ij} = \cos^{-1} \frac{\bar{\psi}_i \cdot \bar{\psi}_j}{|\bar{\psi}_i| |\bar{\psi}_j|} \quad (124)$$

The modes are then grouped so no two orthogonal modes are in the same controller. In some cases the grouping of orthogonal modes in the same controller may be unavoidable.

The selection of modes based on angles will vary with the number of modes in the design model, since the modal vectors, ψ , are calculated based upon the evaluation model's (all modeled modes) modal matrix. Applying the relationship given in Eq (124), the twelve mode model relative angles are given in Table V and for the 36 mode model the relative angles are given in Table VI.

Another consideration in modal grouping is unpredictable ill conditioning of matrices. Aldridge found that when rigid and flexible body modes were assigned to the same controller, loss of symmetry could result and the feedback gains tended to infinity. All of his twelve mode controllers required a separate controller for the rigid body modes. The actual groupings selected for this study will be presented in the following section.

TABLE V
Relative Angles (in Degrees) Between Modes (12)

Mode	4	5	6	7	12	13	17	21	22	24	28	30
4	0.00	89.09	90.00	90.01	90.09	123.20	104.48	90.05	90.01	90.01	39.16	81.12
5	89.09	0.00	67.47	175.39	116.85	90.05	89.97	106.57	116.23	90.16	87.89	87.84
6	90.00	67.47	0.00	115.31	0.00	89.99	90.04	116.99	91.57	80.13	89.41	89.27
7	90.01	175.39	115.31	0.00	60.67	89.69	90.02	73.19	66.49	90.68	92.45	92.35
12	90.09	116.85	104.01	60.67	0.00	89.69	90.01	102.77	68.57	111.87	96.61	95.63
13	123.20	90.05	89.99	39.95	89.69	0.00	61.22	90.07	89.87	90.15	104.90	73.29
17	107.48	89.97	90.04	90.02	90.02	61.22	0.00	89.91	90.15	89.96	80.56	127.80
21	90.05	106.57	116.99	73.19	102.77	90.07	89.91	0.00	93.88	38.05	93.50	93.10
22	90.01	116.23	91.57	66.49	68.57	89.87	90.15	93.88	0.00	103.22	90.51	90.74
24	90.01	90.16	80.13	90.68	111.87	90.15	89.96	38.05	103.22	0.00	94.15	93.50
28	39.16	87.89	89.41	92.45	96.61	104.90	80.56	93.50	90.51	94.15	0.00	89.82
30	81.12	87.94	89.27	92.35	95.63	73.29	127.80	93.10	90.74	93.50	89.82	0.00

TABLE VI

Relative Angles (in degrees) Between Modes (36)

<u>Mode</u>	<u>4</u>	<u>5</u>	<u>6</u>	<u>7</u>	<u>8</u>	<u>9</u>	<u>10</u>	<u>11</u>
4	0							
5	102	0						
6	111	43	0					
7	98	82	84	0				
8	53	116	109	100	0			
9	75	149	146	106	71	0		
10	94	102	111	116	103	70	0	
11	85	115	107	81	63	73	130	0
12	103	60	61	76	98	129	107	88
13	83	43	53	81	96	133	131	87
14	98	142	135	91	86	41	73	70
15	94	58	43	80	93	140	112	96
16	80	63	86	95	114	100	60	135
17	87	87	72	108	77	98	64	123
18	98	123	134	85	87	47	74	76
19	108	82	98	62	96	91	106	72
20	73	64	60	87	77	118	131	79
21	99	127	121	96	103	61	59	88
22	105	121	105	71	83	75	97	59
23	96	95	80	76	85	100	108	71
24	105	121	108	101	92	68	64	95
25	103	27	43	68	106	157	117	99
26	107	80	82	74	100	96	100	93
27	95	120	112	77	75	58	85	81
28	91	64	73	89	108	118	68	122
29	101	96	78	78	63	94	104	78
30	101	101	99	79	102	80	85	88
31	92	74	69	82	95	107	77	121
32	96	89	97	78	76	81	91	89
33	92	76	74	78	91	112	101	91
34	110	90	94	63	106	92	95	87
35	91	121	115	84	69	62	101	52
36	92	64	65	86	85	119	106	90

TABLE VI (Continued)

<u>Mode</u>	<u>12</u>	<u>13</u>	<u>14</u>	<u>15</u>	<u>16</u>	<u>17</u>	<u>18</u>	<u>19</u>
12	0							
13	63	0						
14	122	140	0					
15	33	56	134	0				
16	84	82	115	87	0			
17	82	104	107	68	80	0		
18	117	135	47	139	99	105	0	
19	88	94	81	106	101	113	51	0
20	54	37	141	46	86	90	128	101
21	100	135	48	107	91	91	59	93
22	76	119	59	85	117	91	64	74
23	58	90	95	57	103	78	90	83
24	108	128	67	108	97	79	59	91
25	52	40	137	51	81	92	124	77
26	68	80	107	76	83	89	91	77
27	96	122	69	106	96	89	55	76
28	71	89	107	71	65	67	103	93
29	81	101	90	78	124	62	86	73
30	76	101	77	87	87	97	90	95
31	57	90	113	54	65	49	109	105
32	103	104	85	113	96	89	52	46
33	78	80	114	77	79	93	96	82
34	106	100	75	108	101	104	59	51
35	96	112	53	105	116	106	63	77
36	68	67	107	63	85	86	115	98

TABLE VI (Continued)

<u>Mode</u>	<u>20</u>	<u>21</u>	<u>22</u>	<u>23</u>	<u>24</u>	<u>25</u>	<u>26</u>	<u>27</u>
20	0							
21	125	0						
22	100	51	0					
23	71	70	40	0				
24	115	46	65	71	0			
25	60	128	108	85	119	0		
26	72	105	86	80	100	81	0	
27	100	81	63	85	76	122	65	0
28	96	80	95	83	88	65	115	120
29	90	106	70	75	91	84	77	72
30	94	83	78	94	105	103	50	63
31	76	95	84	74	100	82	65	73
32	103	103	88	99	83	90	89	61
33	72	100	93	82	90	70	79	93
34	114	76	74	79	64	85	99	90
35	101	76	54	78	85	119	91	52
36	72	111	93	84	110	69	98	93

TABLE VI (Continued)

<u>Mode</u>	<u>28</u>	<u>29</u>	<u>30</u>	<u>31</u>	<u>32</u>	<u>33</u>	<u>34</u>	<u>35</u>	<u>36</u>
28	0								
29	94	0							
30	113	91	0						
31	74	77	67	0					
32	93	63	108	94	0				
33	82	87	88	92	90	0			
34	85	87	115	109	63	95	0		
35	124	82	76	93	76	110	83	0	
36	88	92	95	71	91	97	94	68	0

VII. Results

As in most research efforts, much more data was produced than will be presented in this section. The eigenvalue results that do appear should be considered representative of the controller system configurations studied. Results are presented in order of system magnitude, i.e. single controller, twelve mode multiple controller, and 33 mode multiple controller.

The first controller studied was a four mode single controller. The open loop damping ratio applied was 0.01 and a control weighting matrix of $Q = 1000 \underline{I}$ was used for modes 12, 17, 22, and 24. The response of this controller using position sensors only, rate sensors only, a mix of position and rate sensors, and both position and rate sensors (full C matrix) is given in Table VII. Next the rigid body modes were placed in a single controller with position sensors and again with rate sensors. The control weighting matrix was set at $Q = 2 \underline{I}$. The results for these controllers are given in Table VIII. The performance of the single controllers revealed that rate sensors provided the desired response for flexible body modes, but the response of the rigid body modes was undamped for position sensor information and overdamped for rate sensor information. The rigid body modes, however, were retained in the 12 and 33 mode design models, and for these multiple controller runs, only rate sensors were used.

The twelve mode model was examined with three and four controllers. The modes selected for this system are the 12 critical modes identified in the ACOSS Five report (15), i.e. modes 4, 5, 6, 7, 12, 13, 17, 21, 22, 24, 28, 30. The initial mode grouping was identical to Aldridge's three controller (1). The groupings are:

Controller 1: 4, 5, 6

Controller 2: 7, 13, 17, 21, 30

Controller 3: 12, 22, 24, 28

The groupings in these controllers appear undesirable since modes pairs 4-5, 4-6, 7-13, 7-17, 7-30, 13-21, 17-21, 21-30, 12-28, 22-28, and 24-28 are essentially orthogonal (see Table V). The control weighting matrix for the flexible modes was set at $Q = 5000 \underline{I}$ and remained at $Q = 2 \underline{I}$ for the rigid body modes. The eigenvalues for the closed loop system appear in Table IX. While the untransformed system shows improved damping for six of the nine flexible modes, the controller coupling has driven mode 13 (3.74 rad s^{-1}) unstable. After the transformation, the flexible modes remain stable with improved closed loop damping. Modes 28 and 30 remain unchanged though. These two nearly orthogonal modes were subsequently given a higher control weighting and reassigned to other mode groupings:

Controller 1: 4, 5, 6

Controller 2: 7, 13, 17, 21

Controller 3: 12, 22, 24, 28, 30

and

Controller 1: 4, 5, 6

Controller 2: 7, 13, 17, 21

Controller 3: 12, 22, 24

Controller 4: 28, 30

However, the closed loop damping of modes 28 and 30 remained unchanged from the open loop damping of .010. These two modes appear weakly controllable with the given actuator configuration. Examination of the modal matrix (Appendix B) shows the elements for modes 28 and 30 are zero or "small"

compared to the elements for the other modes, which supports the condition of weakly controllable/observable. Also of note is that orthogonal mode pairs like 7-13 not only remained controllable, but both show nice improvements of closed loop damping. The optimal control performance before and after transformation is also of interest and is compared for this system in Table X.

With the multiple controller program operating, the next step was to apply it to the 33 mode design model. Aldridge's results showed ill conditioning of matrices when the rigid body modes were grouped with flexible modes. So for the first run the rigid modes were grouped in a separate controller and the system modes were grouped as:

Controller 1: 4, 5, 6

Controller 2: 7, 8, 9, 10, 11, 12, 13, 14, 15, 16

Controller 3: 17, 18, 19, 20, 21, 22, 23, 24, 25, 26

Controller 4: 27, 28, 29, 30, 31, 32, 33, 34, 35, 36

The eigenvalue results for this system are given in Table XI. Here, as with the 12 mode case, the untransformed system shows coupling that destabilizes modes 8, 18, 20, 21, and 30, with mode 8 driven unstable. The transformed system remains completely stable with only one mode (11) suffering some damping loss. Most modes closed loop damping improves. Especially note controller 3 (modes 17-26) whose closed loop damping for all modes has at least doubled. Five of the higher frequency modes in controller 4 improved, including mode 30 which was uncontrollable in the 12 mode model. The modes whose closed loop damping remained equal to the passive .010 value may again be weakly observable/controllable with the given actuator/sensor configuration.

At this point there were two system conditions to investigate. First was the ill conditioning of matrices that was seen when grouping rigid and flexible modes in one controller. And second, was the initial 33 mode grouping denying better control of the higher frequencies. To check for ill conditioning, the rigid body modes were mixed with flexible modes in four different configurations as follows:

Grouping 1: 4, 5, 6, 7, 9, 10, 13

Grouping 2: 4, 5, 6, 15, 16, 17, 18, 19

Grouping 3: 4, 5, 6, 23, 24, 25, 26, 27

Grouping 4: 4, 5, 6, 31, 32, 33, 34, 35, 36

Mixing with groupings 1, 2, and 3 resulted in completely stable decoupled controllers, but when grouped with the higher frequency modes (Grouping 4), the controller gains became large and the eigenvalue analysis was stifled with infinite values. To see if this was not just a problem of mixing rigid modes with the higher frequencies, the lower frequency modes were mixed with the higher ones as well:

Controller 1: 4, 5, 6, 15, 16, 17, 18, 19

Controller 2: 7, 8, 9, 23, 24, 33, 34, 35

Controller 3: 10, 11, 12, 25, 26, 31, 32, 36

Controller 4: 13, 14, 20, 21, 22, 27, 28, 29, 30

This configuration was "nice" and eigenvalue results after transformation are given in Table XII. Mode 28 remained the only mode whose closed loop damping could not be improved via modal grouping. Unfortunately this mode is considered critical to LOS performance.

Finally the 33 mode model was run in the three controller configuration with residuals. Most of the configurations that contained more than five residuals drove one or more of the transformed system modes unstable.

However, two modal groupings that were successful, i.e. remained stable after transformation, were one with three residual modes and the other with seven residuals. The first system retained weighting matrices of $Q_{\text{rigid}} = 2 \underline{I}$ and $Q_{\text{flexible}} = 5000 \underline{I}$ and was grouped as:

Controller 1: 4, 5, 6, 9, 10, 11, 28, 29

Controller 2: 8, 12, 14, 15, 16, 17, 18, 19, 30, 31

Controller 3: 20, 21, 22, 23, 24, 25, 26, 27, 32, 33

Residuals: 34, 35, 36

The system eigenvalue results for this configuration are given in Table XIII. For this configuration, mode 28 grudgingly improved and most of the actively controlled modes show improved closed loop damping.

The last controller system to examine has the modal groupings:

Controller 1: 4, 5, 6, 9, 10, 11, 13

Controller 2: 8, 12, 14, 15, 16, 17, 18, 19, 20

Controller 3: 21, 22, 23, 24, 26, 28, 29, 30, 31

Residuals: 25, 27, 32, 33, 34, 35, 36

The control weighting matrix on modes 7-20 was raised to $Q = 50,000 \underline{I}$ and lowered on modes 21-36 to create high control of mode frequencies below 6 rad s^{-1} , and a "dead zone" effect for the frequencies above that cutoff. The eigenvalue results for this system are given in Table XIV. Again the closed loop damping is remarkably improved for modes 20 and below and only the residual mode 35 loses some damping.

In general, the untransformed system shows the destabilizing effect of spillover. After transformation, though, system performance still is highly dependent upon modal grouping and control weighting. The condition and degree of controllability/observability is absolutely dependent

on the actuator/sensor configuration and modal matrix values. This is easily seen in following the response of mode 28 through it various controller assignments.

TABLE VII
Eigenvalue Analysis - Single Controller
Modes: 12, 17, 22, 24

<u>OPTIMAL FULL STATE FEEDBACK</u>	<u>ζ</u>
$-.6978118 + 11.1193805i$	.063
$-.1950110 + 7.2780866i$	.027
$-.0613474 + 5.1212778i$	.012
$-.1814128 + 3.4979155i$	.052
 <u>DOFB POSITION SENSORS</u>	
$-.111409 + 11.1441i$	.010
$-.072806 + 7.2804i$	.010
$-.051216 + 5.1214i$	.010
$-.035193 + 3.5013i$	.010
 <u>DOFB RATE SENSORS</u>	
$-.697751 + 11.115567i$	.063
$-.194998 + 7.277893i$	.027
$-.061347 + 5.121270i$	.012
$-.181487 + 3.498374i$	.052
 <u>POSITION (12,17) AND RATE (22,24)</u>	
$-.697816 + 11.117928i$	.063
$-.194983 + 7.278735i$	.027
$-.051216 + 5.121389i$	.010
$-.035056 + 3.501293i$	.010
 <u>POSITION AND RATE SENSORS</u>	
$-.425969 + 11.160118i$	.038
$-.134633 + 7.287732i$	.018
$-.056303 + 5.122321i$	.011
$-.107221 + 3.521215i$	.030

TABLE VIII
Eigenvalue Analysis - Single Controller
Rigid Body Modes: 4, 5, 6

<u>OPTIMAL</u>	<u>ξ</u>
$-0.16093 + .15691i$	.716
$-0.14062 + .13792i$	.714
$-0.11454 + .11307i$	.712

POSITION SENSORS

$0.0 + .2248i$	0.0
$0.0 + .1970i$	0.0
$0.0 + .1609i$	0.0

RATE SENSORS

$-.55906 + 0.0i$	}	>>1
$-.00018 + 0.0i$		
$-.17413 + 0.0i$	}	>>1
$-.00057 + 0.0i$		
$-.04497 + 0.0i$	}	>>1
$-.00222 + 0.0i$		

TABLE IX
Eigenvalue Analysis - 3 Controllers

$Q_{\text{rigid}} = 2$ $Q_{\text{flexible}} = 5000$

Mode Assignments

Controller 1: 4, 5, 6 Controller 3: 12, 22, 24, 28
Controller 2: 7, 13, 17, 21, 30 Residuals: None

Overall System Eigenvalues

Mode	Before Transformation	ζ	After Transformation	ζ
30	-0.2505 + 25.0483i	.010	-0.2505 + 25.0483i	.010
28	-0.2169 + 21.6903i	.010	-0.2169 + 21.6903i	.010
24	-2.7170 + 10.5290i	.250	-0.7316 + 11.1106i	.066
22	-0.3863 + 7.1399i	.054	-0.2549 + 7.2653i	.035
21	-0.2484 + 6.1062i	.041	-0.2987 + 6.0988i	.049
17	-0.0679 + 5.1249i	.013	-0.0853 + 5.1191i	.017
13	+0.3818 + 4.1764i	unstable	-0.9443 + 3.6271i	.252
12	-0.1040 + 3.3917i	.031	-0.2371 + 3.5011i	.068
7	-0.0191 + 0.6893i	.028	-0.0427 + 3.7148i	.060
6	-7.1082 + 0.0i } 0.1 E-8 + 0.0i }	>>1	-0.3219 + 0.0i } 0.0002 + 0.0i }	>>1
5	-2.1548 + 0.0i } 0.2 E-7 + 0.0i }	>>1	-0.2535 + 0.0i } 0.0002 + 0.0i }	>>1
4	-0.4526 + 0.0i } 0.0001 + 0.0i }	>>1	0.1414 + 0.0i } 0.0023 + 0.0i }	>>1

TABLE X

Eigenvalue Analysis - 3 Controllers (Optimal)

$$Q_{\text{rigid}} = 2 \quad Q_{\text{flexible}} = 5000$$

Mode Assignments

Controller 1: 4, 5, 6 Controller 3: 12, 22, 24, 28
 Controller 2: 7, 13, 17, 21, 30 Residuals: None

Controller Eigenvalues

<u>Before Transformation (BG)</u>	<u>ζ</u>	<u>After Transformation (B*G*)</u>	<u>ζ</u>
Controller 1			
-0.16093 + 0.15691i	.716	-0.16093 + 0.15691i	.716
-0.14062 + 0.13792i	.714	-0.14062 + 0.13792i	.714
-0.11454 + 0.11307i	.712	-0.11454 + 0.11307i	.712
Controller 2			
-0.25050 + 25.04834i	.010	-0.25050 + 25.04834i	.010
-0.34700 + 6.09807i	.057	-0.29920 + 6.10041i	.049
-0.08782 + 5.12071i	.017	-0.08661 + 5.12082i	.017
-1.12614 + 3.59620i	.299	-0.94251 + 3.64174i	.251
-0.59706 + 0.75548i	.620	-0.04268 + 0.71644i	.059
Controller 3			
-0.21692 + 21.69033i	.010	-0.21691 + 21.69033i	.010
-1.54350 + 11.03465i	.139	-0.73205 + 11.11716i	.066
-0.40996 + 7.26962i	.056	-0.25523 + 7.27624i	.035
-0.39925 + 3.48263i	.114	-0.23632 + 3.49517i	.067

TABLE XI

Eigenvalue Analysis - 4 Controllers

Mode Assignments

Controller 1: 4-6
Controller 2: 7-16

Controller 3: 17-26
Controller 4: 27-36

Mode	Before Transformation	ζ	After Transformation	ζ
36	-0.5563 + 55.6271i	.010	-0.5618 + 55.6269i	.010
5	-0.5311 + 53.1101i	.010	-0.5316 + 53.1101i	.010
4	-0.5066 + 50.6708i	.010	-0.5064 + 50.6716i	.010
3	-0.4952 + 41.2112i	.012	-0.4371 + 41.2318i	.011
2	-2.1740 + 26.8590i	.081	-0.3024 + 27.2506i	.011
1	-0.3027 + 25.5128i	.012	-0.2789 + 25.4481i	.011
30	-0.2300 + 25.0623i	.009	-0.4011 + 25.0463i	.016
29	-0.2486 + 24.8632i	.010	-0.2617 + 24.8747i	.011
28	-0.2170 + 21.6915i	.010	-0.2181 + 21.6904i	.010
27	-0.2274 + 21.5769i	.011	-0.2158 + 21.5727i	.010
26	-0.1372 + 14.1619i	.010	-0.4941 + 14.1297i	.035
25	-0.2164 + 14.1263i	.015	-0.2765 + 14.1218i	.020
24	-0.1203 + 11.1374i	.011	-0.3710 + 11.1009i	.033
23	-0.1085 + 9.7419i	.011	-0.3310 + 9.7432i	.034
22	-0.1823 + 7.2325i	.025	-0.2890 + 7.2147i	.040
21	-0.0529 + 6.1024i	.009	-0.3465 + 6.0180i	.057
20	-0.0463 + 5.6700i	.008	-0.6631 + 5.6971i	.116
19	-0.0535 + 5.1690i	.010	-0.5306 + 5.2201i	.101
18	-0.0093 + 5.1363i	.002	-0.3346 + 5.1874i	.064
17	-0.0987 + 5.1204i	.018	-0.1277 + 5.1317i	.025
16	-0.0511 + 4.0070i	.013	-0.0423 + 4.0074i	.011
15	-0.0409 + 3.9980i	.010	-0.0543 + 3.9704i	.014
14	-0.0643 + 3.8315i	.017	-0.3137 + 3.7563i	.083
13	-0.0855 + 3.7221i	.023	-0.0408 + 3.7520i	.011
12	-0.1230 + 3.4744i	.035	-0.0353 + 3.5018i	.010
11	-0.1786 + 2.8124i	.063	-0.2616 + 2.8557i	.091
10	-0.0247 + 1.0029i	.025	-0.0293 + 1.0764i	.027
9	-0.0100 + 0.9235i	.011	-0.0097 + 0.9376i	.010
8	+0.0242 + 0.7261i	unstable	-0.0550 + 0.8444i	.065
7	-0.4161 + 0.6663i	.530	-0.2689 + 0.7831i	.325
6	-0.9692 + 0.0i } -0.0002 + 0.0i }	>>1	-0.0001 + 0.0099i	.010
5	-0.1755 + 0.0i } -0.0006 + 0.0i }	>>1	-0.0001 + 0.0099i	.010
4	-0.0502 + 0.0i } -0.0022 + 0.0i }	>>1	-0.3002 + 0.0i } -0.0003 + 0.0i }	>>1

TABLE XII

Eigenvalue Analysis - 4 Controllers

Mode Assignments

Controller 1: 4-6, 15-19 Controller 3: 10-12, 25, 26, 31, 32, 36
 Controller 2: 7-9, 23, 23, 33, 34, 35 Controller 4: 13, 14, 20-22, 27-30

After Transformation

<u>Controller 1</u>	<u>ζ</u>
-0.0590 + 5.1721i	.011
-0.0513 + 5.1248i	.010
-0.4740 + 4.9394i	.096
-0.1130 + 4.0306i	.028
-0.0406 + 3.9992i	.010
Rigid Body	>>1

<u>Controller 2</u>	<u>ζ</u>
-0.6904 + 53.1049i	.013
-0.5774 + 50.6682i	.011
-0.4449 + 41.2314i	.011
-0.3913 + 11.0811i	.035
-0.3550 + 9.7809i	.036
-0.3861 + 0.8432i	.416
-0.4367 + 0.6173i	.578
-0.7272 + 0.5072i	.820

<u>Controller 3</u>	<u>ζ</u>
-0.5927 + 55.6253i	.011
-0.4866 + 27.2171i	.018
-0.5558 + 25.4804i	.022
-0.1890 + 14.1351i	.013
-0.9301 + 14.0928i	.066
-0.1337 + 3.4590i	.039
-0.3617 + 2.8683i	.125
-0.5185 + 0.9766i	.469

<u>Controller 4</u>	<u>ζ</u>
-0.2513 + 25.0480i	.010
-0.3108 + 24.8594i	.013
-0.2210 + 21.6878i	.010
-0.2834 + 21.5732i	.013
-0.1345 + 7.2705i	.018
-0.0616 + 6.1070i	.010
-0.1522 + 5.7430i	.026
-0.0564 + 3.8072i	.015
-0.2251 + 3.8044i	.059

TABLE XIII

Eigenvalue Analysis - 3 Controller

Mode Assignments

Controller 1: 4-6, 9-11, 28, 29 Controller 3: 20-27, 32, 33
 Controller 2: 8, 12, 14-19, 30, 31 Residuals: 34-36

<u>Mode</u>	<u>After Transformation</u>	<u>ζ</u>
36	-0.5562 + 55.6270i	.010
35	-0.5313 + 53.1227i	.010
34	-0.5067 + 50.6687i	.010
33	-0.4124 + 41.2324i	.010
32	-0.2730 + 27.2535i	.010
31	-0.6159 + 25.4194i	.024
30	-0.6987 + 25.0474i	.028
29	-0.2496 + 24.8631i	.010
28	-0.2484 + 21.6868i	.011
27	-0.2164 + 21.5727i	.010
26	-0.2615 + 14.1483i	.018
25	-0.1416 + 14.1373i	.010
24	-0.2567 + 11.1296i	.023
23	-0.1845 + 9.7514i	.019
22	-0.0785 + 7.2801i	.011
21	-0.0864 + 6.1025i	.014
20	-0.1328 + 5.7604i	.023
19	-0.6081 + 5.0370i	.120
18	-0.4594 + 5.0346i	.091
17	-0.2799 + 5.0143i	.056
16	-0.7299 + 4.0903i	.176
15	-0.2282 + 3.9903i	.057
14	-0.7534 + 3.7874i	.195
13	-0.0503 + 3.7379i	.013
12	-0.2638 + 3.5546i	.074
11	-0.0310 + 2.8596i	.011
10	-0.0149 + 1.0883i	.014
9	-0.0364 + 0.8697i	.042
8	-0.0183 + 0.6945i	.026
7	-0.7512 + 0.5480i	.808
4-5-6		>>1

TABLE XIV
Eigenvalue Analysis - 3 Controllers

Mode Assignments

Controller 1: 4-6, 9-11, 13 Controller 3: 21-24, 26, 28-31
Controller 2: 8, 12, 14-20 Residuals: 25, 27, 32-36

<u>Mode</u>	<u>After Transformation</u>	<u>ζ</u>
36	-0.5488 + 55.6505i	.010
35	-0.3455 + 53.1892i	.006
34	-0.5839 + 50.6627i	.012
33	-0.5755 + 41.0474i	.014
32	-2.0730 + 26.9359i	.077
31	-0.2586 + 25.4573i	.010
30	-0.2524 + 25.0479i	.010
29	-0.2507 + 24.8623i	.010
28	-0.2172 + 21.6903i	.010
27	-0.4040 + 21.5123i	.019
26	-0.1450 + 14.1625i	.010
25	-0.2526 + 14.1139i	.018
24	-0.1137 + 11.1404i	.010
23	-0.0989 + 9.7452i	.010
22	-0.0740 + 7.2802i	.010
21	-0.0643 + 6.1075i	.011
20	-0.3642 + 4.9781i	.073
19	-0.3872 + 4.8587i	.079
18	-1.2217 + 4.8231i	.246
17	-1.9930 + 4.2648i	.423
16	-2.2082 + 3.9841i	.485
15	-0.9154 + 3.9237i	.227
14	-0.4845 + 3.7452i	.128
13	-0.1403 + 3.5873i	.039
12	-2.6692 + 3.3207i	.627
11	-3.0419 + 2.6371i	.756
10	-0.2659 + 2.3818i	.111
9	-0.0274 + 0.8565i	.032
8	-0.1131 + 0.2524i	.409
7	-0.1027 + 0.0386i	.936
4-5-6		>>1

VIII. Conclusions

This investigation demonstrated the feasibility and merits of the decentralized direct output feedback control method. This technique can be used to design feedback gains for multiple controllers that decouple the individual controllers by eliminating controller to controller observation (or control) spillover. While system stability was maintained for design models that did not include residual modes, system stability cannot be guaranteed for design models that include residual modes. Modal grouping played a significant part in the system stability achieved.

Suitable modal grouping may have to be accomplished by "trial and error" unless a more technical approach can be devised. The use of angles between modal amplitude or rate vectors is a convenient method for initial grouping of modes. However, orthogonal modal pairs may be satisfactorily controlled even though they are in the same controller. The rank of the B and C matrices should be examined closely. If they are not of full rank, the modal groups may not be fully compatible as indicated by zero entries in the non-zero singular values of the singular value decomposition. Since the B and C matrices are a function of actuator and sensor placement, this model evaluation technique is a valuable design tool for examining performance of actuator and sensor configurations. The $\Phi^T D$ matrix can be checked for determining mode observability and controllability. This step should be considered part of the control-configured system approach.

A major shortcoming of direct output feedback using rate sensors was the loss of control of the rigid body modes. This is certainly not a fatal flaw for this control method. The rigid body modes may be controlled by a

separate optimal control system or additional states (e.g., rigid body angles) may be added to the state vector.

Overall closed loop damping was improved for actively controlled modes and system stability maintained for both the 12- and 33-mode design models. This study has demonstrated a control technique that eliminated the requirement for the observer states of modern control and allows either control of twice as many modes or dramatically reduces the computation burden to control the same number of modes.

IX. Recommendations

While this study has demonstrated the successful application of the decentralized DOFB method, it is not an end in itself. Before a control method is applied to a system, a control-configuration approach should be developed to establish desirable mode selection and grouping. This should involve examination of techniques to orient sensors and actuators specifically for the critical modes. This approach should include examination of the changes in the $\Phi^T D$ matrices with variations of the number of modes in the design model and reconfiguration of sensors/actuators. It may be possible to develop a method to establish optimal modal grouping.

System robustness is a desirable characteristic for control systems. Just how sensitive this control method is to parameter variations and component failure should be established. Also, analysis to predict ill conditioning of matrices would certainly be useful to a control designer.

The systems studied have provided a take off point for calculations of the line of sight (LOS) error for both forced and unforced responses. The systems should also be modified to provide proper control of the rigid body modes if they are included in the design model. This may be easily achieved by addition of body angles to the state vector.

One further item of interest is the computer burden comparisons for modern (cost optimal) and DOFB calculations. Both systems should be written using efficient programming techniques.

Appendix A

Main Program and Subroutine Listing

```

C      PROGRAM DOFB33(INPUT, OUTPUT, TAPE8, TAPE6)
C
C      PFN:  DOFB335
C
C      THIS PROGRAM GENERATES A LOWER TRIANGULAR TRANSFORMATION
C      THE THREE OR FOUR CONTROLLER SOLUTION MAY INCLUDE RESIDUALS.
C
      REAL A1(21,21),A2(21,21),A3(21,21),A4(21,21)
      REAL B1(21,21),B2(21,21),B3(21,21),B4(21,21)
      REAL BSTAR2(21,21),BSTAR3(21,21),BSTAR4(21,21)
      REAL CSTAR1(21,21),CSTAR2(21,21),CSTAR3(21,21)
      REAL C1(21,21),C2(21,21),C3(21,21),C4(21,21)
      REAL CTCC1(21,21),CTCC2(21,21),CTCC3(21,21)
      REAL GT1(21,21),GT2(21,21),GT3(21,21)
      REAL SI(24),KSTAR4(21,21),GSTAR4(21,21)
      REAL UT(21,21),VP(21,21),QPLUS(21,21)
      REAL QUT(21,21),CPLUS(21,21),KHAT4(21,21)
      REAL KHAT1(21,21),KHAT2(21,21),KHAT3(21,21)
      REAL KSTAR1(21,21),KSTAR2(21,21),KSTAR3(21,21)
      REAL SAT(21,21),SAT2(21,21),SAT3(21,21),SAT4(21,21)
      REAL AKC(21,21),ACT(21,21),BCG(21,21),KCC(21,21)
      REAL P(21,21),S(21,21)
      REAL ABK1(21,21),ABK2(21,21),ABK3(21,21)
      REAL GSTAR1(21,21),GSTAR2(21,21),GSTAR3(21,21)
      REAL TKG1(21,21),TKG2(21,21),TKG3(21,21)
      REAL QA1(21,21),QA2(21,21),QA3(21,21),QA4(21,21)
      REAL ABG1(21,21),ABG2(21,21),ABG3(21,21),ABG4(21,21)
      REAL ABG5(21,21),ABG6(21,21),ABG7(21,21),ABG8(21,21)
      REAL GAIN1(21,21),GAIN2(21,21),GAIN3(21,21),GAIN4(21,21)
      REAL KT1(21,21),KT2(21,21),KT3(21,21),KT4(21,21)
      REAL KOB1(21,21),KOB2(21,21),KOB3(21,21),KOB4(21,21)
      REAL GAMMA1(21,21),GAMMA2(21,21),GAMMA3(21,21)
      REAL T2(21,21),T3(21,21),T4(21,21)
      REAL TRT(21,21),TEN(21,21),CT(21,21),V(21,21)
      REAL RK(21,21),RK1(21,21),RK2(21,21),RK3(21,21)
      REAL RG2(21,21),RG3(21,21),RG4(21,21)
      REAL MAJM(66,66),D(33),X0(66),W(33),TOL,DT
      REAL ZETA,AA(33),BB(33),SING(21),XTR(21,21),X1(66)
      REAL EAT(1,1),EAT2(1,1),WORK(66,66),STOR(21,21)
      REAL PHIA(33,21),PHIS(33,21),MODE(2,21),INIT(4,21)
      INTEGER N,N2,NC1,NC2,NC3,NC12,NC22,NC32,NR,NR2
      INTEGER IC1(21),IC2(21),IC3(21),I,J,K,L,M,KK,LL,MM
      INTEGER DEC,Q,NACT,NSEN,IR(21),IER,SKIP,NCOL,NCOL1
      INTEGER NDA,NDIM,NDA1,NDIM1,ZZ,E2,E3,E4,P1,P2,P3,P5
      COMPLEX Z(24),W1(12)
      COMMON/MAINA/NDA,NDA1
      COMMON/MAINB/NCOL,NCOL1
      COMMON/MAIN1/NDIM,NDIM1,TEN,X(3364)
      COMMON/MAIN2/STOR
      COMMON/MAIN3/XTR
      COMMON/SAVE/T(100),TS(100)
      COMMON/INOUT/KOUT,TAPE
      COMMON/NUM/IC1,IC2,IC3,IR,NC1,NC2,NC3,NR

```

```

C
C
C
C
C
INITIALIZATIONS

```

```

NDIM = 33
NDIM1 = 34
NCOL = 21
NCOL1 = 22
NDA = 66
NDA1 = 67
KOUT = 6
TAPE = 9
Q = 0
IER = 0
ZZ = 0

```

```

C
C

```

```

PRINT' (////)'
PRINT*, '*****'
PRINT*, '*****'
PRINT*, '***** DIRECT OUTPUT FEEDBACK *****'
PRINT*, '*****'
PRINT*, '***** 3 4 L O W E R - R E S I D U A L *****'
PRINT*, '***** B L O C K *****'
PRINT*, '***** C S D L I I *****'
PRINT*, '*****'
PRINT' (//)'
PRINT*, ' THIS PROGRAM GENERATES A SOLUTION',
* ' USING A LOWER TRIANGULAR TRANSFORMATION '
PRINT' (////)'

```

```

C
C
C
C
C
C
C
C
C
C

```

```

INITIAL SELECTION FOR THREE OR FOUR CONTROLLERS

```

```

PRINT*, ' FOR A THREE CONTROLLER RUN, ENTER 3, OR, '
PRINT*, ' FOR A FOUR CONTROLLER RUN, ENTER 4 >'
READ(8,*) DEC

```

```

C
C
C
C

```

```

DEC DEFAULT SWITCH

```

```

IF (DEC.NE.4) DEC = 3
PRINT*, ' '
PRINT*, ' THIS IS A ',DEC,' CONTROLLER RUN '

```

```

C
C

```

```

C PHI MATRICES AND CONTROLLER ENTRIES
C
C
PRINT' (//)'
IF (DEC.EQ.3) THEN
PRINT*, ' ENTER NC1,NC2,NC3,NR,NACT,NSEN,ZETA >'
ELSE
PRINT*, ' ENTER NC1,NC2,NC3,NC4,NACT,NSEN,ZETA >'
ENDIF
READ(8,*) NC1,NC2,NC3,NR,NACT,NSEN,ZETA
PRINT*, NC1,NC2,NC3,NR,NACT,NSEN,ZETA
PRINT*, ' '
PRINT*, ' ENTER THE ',NACT,' ELEMENTS FOR EACH PHIA '
PRINT*, ' '
N = NC1 + NC2 + NC3 + NR
DO 1 I=1,N
PRINT*, 'ENTER PHIA ',I,' >'
READ(8,*) (PHIA(I,J),J=1,NACT)
PRINT*, ' ',(PHIA(I,J),J=1,NACT)
1 CONTINUE
PRINT' (//)'
PRINT*, ' ENTER THE ',NSEN,' ELEMENTS FOR EACH PHIS '
PRINT*, ' '
DO 2 I=1,N
PRINT*, 'ENTER PHIS ',I,' >'
READ(8,*) (PHIS(I,J),J=1,NSEN)
PRINT*, ' ',(PHIS(I,J),J=1,NSEN)
2 CONTINUE
PRINT' (//)'
C
C
C OMEGAS
C
C
PRINT*, ' ENTER THE VALUE FOR EACH OMEGA '
PRINT*, ' '
DO 3 I=1,N
PRINT*, 'ENTER OMEGA ',I,' >'
READ(8,*) W(I)
PRINT*, ' ',W(I)
D(I) = -2. * ZETA * W(I)
3 CONTINUE
C
C
20 CONTINUE
C
C SECONDARY SELECTION FOR THREE OR FOUR CONTROLLERS, TO
C BE USED FOR RUNS AFTER THE FIRST JOB
C
C
IF (Q.EQ.2) THEN
PRINT*, ' FOR A THREE CONTROLLER RUN, ENTER 3, OR, '

```

```

PRINT*, ' FOR A FOUR CONTROLLER RUN, ENTER 4 >'
READ(8,*) DEC
C
C DEC DEFAULT SWITCH
C
IF (DEC.NE.4) DEC = 3
PRINT*, ' '
PRINT*, ' THIS IS A ',DEC,' CONTROLLER RUN '
PRINT' (//)'
C
C
IF (DEC.EQ.3) THEN
PRINT*, ' ENTER THE VALUES OF NC1,NC2,NC3,NR >'
ELSE
PRINT*, ' ENTER THE VALUES OF NC1,NC2,NC3,NC4 >'
ENDIF
READ(8,*) NC1,NC2,NC3,NR
PRINT*,NC1,NC2,NC3,NR
PRINT' (//)'
ENDIF
C
C
C MODE ASSIGNMENT TO CONTROLLERS
C
PRINT' (///)'
PRINT*, ' THE FOLLOWING MODES ARE ENTERED ACCORDING TO THE '
PRINT*, ' ORDER IN WHICH THEY ARE ENTERED IN THE DATA FILE '
PRINT*, ' AND NOT ACCORDING TO THEIR ACTUAL MODE NUMBER. '
PRINT' (//)'
PRINT*, ' ENTER THE ',NC1,' CONTROLLER 1 MODES >'
READ(8,*) (IC1(I),I=1,NC1)
PRINT*, ' ',(IC1(I),I=1,NC1)
PRINT*, ' '
PRINT*, ' ENTER THE ',NC2,' CONTROLLER 2 MODES >'
READ(8,*) (IC2(I),I=1,NC2)
PRINT*, ' ',(IC2(I),I=1,NC2)
PRINT*, ' '
PRINT*, ' ENTER THE ',NC3,' CONTROLLER 3 MODES >'
READ(8,*) (IC3(I),I=1,NC3)
PRINT*, ' ',(IC3(I),I=1,NC3)
PRINT*, ' '
IF (DEC.EQ.3) THEN
PRINT*, ' ENTER THE ',NR,' RESIDUAL MODES >'
ELSE
PRINT*, ' ENTER THE ',NR,' CONTROLLER 4 MODES >'
ENDIF
READ(8,*) (IR(I),I=1,NR)
PRINT*, ' ',(IR(I),I=1,NR)
PRINT*, ' '
C
C
NC12 = 2 * NC1

```

```

NC22 = 2 * NC2
NC32 = 2 * NC3
N2 = 2 * N
NR2 = 2 * NR
M = NC12 + NC22 + NC32 + NR2
NDA = M
NDA1 = M + 1
NDIM = N
NDIM1 = N + 1
100 CONTINUE
PRINT*, ' TO PRINT ALL OF THE MATRICES ENTER 1, ELSE ENTER 0 >'
READ(8,*) Q
PRINT' (///)'

```

C
C
C
C
C
C

READ IN THE WEIGHTING MATRIX DIAGONAL
VALUE FOR EACH MODE

```

PRINT*, ' ENTER THE DIAGONAL VALUES, IN MODE INPUT '
PRINT*, ' ORDER, FOR THE CONTROL WEIGHTING MATRIX >'
READ(8,*) (AA(I), I=1, N)
PRINT*, ' '
PRINT*, (AA(I), I=1, N)
PRINT' (///)'
PRINT' (///)'

```

C
C
C
C
C
C

FORMING THE A, B, C AND WEIGHTING MATRICES

```

CALL FORMA(A1, D, W, NC1, NC12, IC1)
CALL FORMB(B1, PHIA, NC1, NC12, NACT, IC1)
CALL FORMC(C1, PHIS, NC1, NC12, NSEN, IC1)
CALL FORMA(A2, D, W, NC2, NC22, IC2)
CALL FORMB(B2, PHIA, NC2, NC22, NACT, IC2)
CALL FORMC(C2, PHIS, NC2, NC22, NSEN, IC2)
CALL FORMA(A3, D, W, NC3, NC32, IC3)
CALL FORMB(B3, PHIA, NC3, NC32, NACT, IC3)
CALL FORMC(C3, PHIS, NC3, NC32, NSEN, IC3)
CALL FORMA(A4, D, W, NR, NR2, IR)
CALL FORMB(B4, PHIA, NR, NR2, NACT, IR)
CALL FORMC(C4, PHIS, NR, NR2, NSEN, IR)
CALL FORMQ(QA1, AA, NC1, IC1)
CALL FORMQ(QA2, AA, NC2, IC2)
CALL FORMQ(QA3, AA, NC3, IC3)
IF (DEC. EQ. 4) THEN
CALL FORMQ(QA4, AA, NR, IR)
ENDIF

```

C
C
C
C
C

PRINTING THE A, B, C AND WEIGHTING MATRICES

C

```
IF (Q.EQ.1) THEN
PRINT*, ' THE CONTROLLER 1 A MATRIX IS '
CALL PRNT(A1,NC12,NC12)
PRINT*, ' THE CONTROLLER 1 B MATRIX IS '
CALL PRNT(B1,NC12,NACT)
PRINT*, ' THE CONTROLLER 1 C MATRIX IS '
CALL PRNT(C1,NSEN,NC12)
PRINT*, ' THE C1 CONTROL WEIGHTING MATRIX IS '
CALL PRNT(QA1,NC12,NC12)
PRINT*, ' THE CONTROLLER 2 A MATRIX IS '
CALL PRNT(A2,NC22,NC22)
PRINT*, ' THE CONTROLLER 2 B MATRIX IS '
CALL PRNT(B2,NC22,NACT)
PRINT*, ' THE CONTROLLER 2 C MATRIX IS '
CALL PRNT(C2,NSEN,NC22)
PRINT*, ' THE C2 CONTROL WEIGHTING MATRIX IS '
CALL PRNT(QA2,NC22,NC22)
PRINT*, ' THE CONTROLLER 3 A MATRIX IS '
CALL PRNT(A3,NC32,NC32)
PRINT*, ' THE CONTROLLER 3 B MATRIX IS '
CALL PRNT(B3,NC32,NACT)
PRINT*, ' THE CONTROLLER 3 C MATRIX IS '
CALL PRNT(C3,NSEN,NC32)
PRINT*, ' THE C3 CONTROL WEIGHTING MATRIX IS '
CALL PRNT(QA3,NC32,NC32)
```

C

```
IF (NR.EQ.0) THEN
PRINT*, ' NO RESIDUAL TERMS '
GOTO 115
ENDIF
```

C

```
IF (DEC.EQ.3) THEN
PRINT*, ' THE A RESIDUAL MATRIX IS '
CALL PRNT(A4,NR2,NR2)
PRINT*, ' THE B RESIDUAL MATRIX IS '
CALL PRNT(B4,NR2,NACT)
PRINT*, ' THE C RESIDUAL MATRIX IS '
CALL PRNT(C4,NSEN,NR2)
ELSE
PRINT*, ' THE CONTROLLER 4 A MATRIX IS '
CALL PRNT(A4,NR2,NR2)
PRINT*, ' THE CONTROLLER 4 B MATRIX IS '
CALL PRNT(B4,NR2,NACT)
PRINT*, ' THE CONTROLLER 4 C MATRIX IS '
CALL PRNT(C4,NSEN,NR2)
PRINT*, ' THE C4 CONTROL WEIGHTING MATRIX IS '
CALL PRNT(QA4,NR2,NR2)
ENDIF
```

C

ENDIF

C

115 CONTINUE

```

C
C
C THIS SECTION GENERATES THE RICCATI SOLUTIONS
C AND THE GAIN MATRICES OF EACH CONTROLLER
C
C
120 CALL VMULFP(B1,B1,NC12,NACT,NC12,NCOL,NCOL,SAT,NCOL,IER)
CONTINUE
IER = 0
TOL = 0.001
PRINT*, ' THE FOLLOWING ARE THE MRIC A+BG 1 INPUTS '
PRINT'(/\)'
PRINT*, ' THE MATRIX A1 IS '
CALL PRNT(A1,NC12,NC12)
PRINT*, ' THE MATRIX SAT (B1*B1T) IS '
CALL PRNT(SAT,NC12,NC12)
PRINT*, ' THE MATRIX QA1 IS '
CALL PRNT(QA1,NC12,NC12)
PRINT*, ' NC12 = ',NC12
PRINT'(/\)'
CALL MRIC(NC12,A1,SAT,QA1,S,ABG1,TOL,IER)
C
C ABG1 = A1 + B1G1
C
PRINT*, ' THE EIGENVALUES IF A1 + B1G1 '
PRINT'(/\)'
CALL EIGRF(ABG1,NC12,NCOL,0,W1,TEN,NCOL,STOR,IER)
DO 629 I=1,NC12
629 PRINT*, ' ',W1(I)
PRINT'(/\)'
PRINT*, ' THE RICCATI SOLUTION OF AC + BCG #1 IS '
CALL PRNT(S,NC12,NC12)
CALL VMULFM(B1,S,NC12,NACT,NC12,NCOL,NCOL,GAIN1,NCOL,IER)
PRINT*, ' THE G1 GAIN MATRIX IS '
CALL PRNT(GAIN1,NACT,NC12)
CALL FINDK(C1,NSEN,NC12,WORK,GAIN1,NSEN,KHAT1)
PRINT*, ' THE FEEDBACK MATRIX KHAT1 IS '
CALL PRNT(KHAT1,NACT,NSEN)
CALL ABGC(A1,B1,C1,NC12,KHAT1,NACT,NSEN,ABK1)
CALL EIGRF(ABK1,NC12,NCOL,0,W1,TEN,NCOL,STOR,IER)
PRINT*, ' EIGENVALUES OF A1 + B1K1C1 ARE '
PRINT'(/\)'
DO 632 I=1,NC12
632 PRINT*, ' ',W1(I)
PRINT'(/\)'
125 CONTINUE
CALL VMULFP(B2,B2,NC22,NACT,NC22,NCOL,NCOL,SAT2,NCOL,IER)
140 CONTINUE
IER = 0
TOL = 0.001
PRINT*, ' THE FOLLOWING ARE THE MRIC A+BG 2 INPUTS '
PRINT'(/\)'
PRINT*, ' THE MATRIX A2 IS '

```

```

CALL PRNT(A2,NC22,NC22)
PRINT*, ' THE MATRIX SAT2 (B2*B2T) IS '
CALL PRNT(SAT2,NC22,NC22)
PRINT*, ' THE MATRIX QA2 IS '
CALL PRNT(QA2,NC22,NC22)
PRINT*, ' NC22 = ',NC22
PRINT' (//)'
CALL MRIC(NC22,A2,SAT2,QA2,S,ABG2,TOL,IER)
PRINT*, ' THE EIGENVALUES OF A2 + B2G2 '
CALL EIGRF(ABG2,NC22,NCOL,0,W1,TEN,NCOL,STOR,IER)
PRINT' (/)'
DO 630 I=1,NC22
630 PRINT*, ' ',W1(I)
PRINT' (//)'
PRINT*, ' THE RICCATI SOLUTION OF AC + BCG #2 IS '
CALL PRNT(S,NC22,NC22)
CALL VMULFM(B2,S,NC22,NACT,NC22,NCOL,NCOL,GAIN2,NCOL,IER)
PRINT*, ' THE G2 GAIN MATRIX IS '
CALL PRNT(GAIN2,NACT,NC22)
CALL FINDK(C2,NSEN,NC22,WORK,GAIN2,NSEN,KHAT2)
PRINT*, ' THE FEEDBACK MATRIX KHAT2 IS '
CALL PRNT(KHAT2,NACT,NSEN)
CALL ABGC(A2,B2,C2,NC22,KHAT2,NACT,NSEN,ABK2)
CALL EIGRF(ABK2,NC22,NCOL,0,W1,TEN,NCOL,STOR,IER)
PRINT' (/)'
PRINT*, ' THE EIGENVALUES OF A2 + B2K2C2 ARE '
DO 633 I=1,NC22
633 PRINT*, ' ',W1(I)
PRINT' (//)'
145 CONTINUE
CALL VMULFP(B3,B3,NC32,NACT,NC32,NCOL,NCOL,SAT3,NCOL,IER)
150 CONTINUE
IER = 0
TOL = 0.001
PRINT*, ' THE FOLLOWING ARE THE MRIC A+BG 3 INPUTS '
PRINT' (//)'
PRINT*, ' THE MATRIX A3 IS '
CALL PRNT(A3,NC32,NC32)
PRINT*, ' THE MATRIX SAT3 (B3*B3T) IS '
CALL PRNT(SAT3,NC32,NC32)
PRINT*, ' THE MATRIX QA3 IS '
CALL PRNT(QA3,NC32,NC32)
PRINT*, ' NC32 = ',NC32
PRINT' (//)'
CALL MRIC(NC32,A3,SAT3,QA3,S,ABG3,TOL,IER)
PRINT*, ' THE EIGENVALUES OF A3 + B3G3 '
CALL EIGRF(ABG3,NC32,NCOL,0,W1,TEN,NCOL,STOR,IER)
PRINT' (/)'
DO 631 I=1,NC32
631 PRINT*, ' ',W1(I)
PRINT' (//)'
PRINT*, ' THE RICCATI SOLUTION OF AC + BCG #3 IS '
CALL PRNT(S,NC32,NC32)

```

```

CALL VMULFM(B3, S, NC32, NACT, NC32, NCOL, NCOL, GAIN3, NCOL, IER)
PRINT*, ' THE G3 GAIN MATRIX IS'
CALL PRNT(GAIN3, NACT, NC32)
CALL FINDK(C3, NSEN, NC32, WORK, GAIN3, NSEN, KHAT3)
PRINT*, 'THE FEEDBACK MATRIX KHAT3 IS'
CALL PRNT(KHAT3, NSEN, NSEN)
CALL ABGC(A3, B3, C3, NC32, KHAT3, NACT, NSEN, ABK3)
CALL EIGRF(ABK3, NC32, NCOL, 0, W1, TEN, NCOL, STOR, IER)
PRINT*, ' THE EIGENVALUES OF A3 + B3K3C3 '
PRINT'(/)'
DO 634 I=1, NC32
634 PRINT*, ' ', W1(I)
PRINT'(/)'
155 CONTINUE
IF (DEC.EQ.4) THEN
CALL VMULFP(B4, B4, NR2, NACT, NR2, NCOL, NCOL, SAT4, NCOL, IER)
160 CONTINUE
IER = 0
TOL = 0.001
CALL MRIC(NR2, A4, SAT4, QA4, S, ABG4, TOL, IER)
PRINT*, ' THE RICCATI SOLUTION OF AC + BCG #4 IS '
CALL PRNT(S, NR2, NR2)
CALL VMULFM(B4, S, NR2, NACT, NR2, NCOL, NCOL, GAIN4, NCOL, IER)
PRINT*, ' THE G4 GAIN MATRIX IS '
CALL PRNT(GAIN4, NACT, NR2)
CALL FINDK(C4, NSEN, NR2, WORK, GAIN4, NSEN, KHAT4)
PRINT*, 'THE FEEDBACK MATRIX KHAT4 IS'
CALL PRNT(KHAT4, NSEN, NSEN)
ENDIF
165 CONTINUE
C
C CALCULATE SUMMATION OF KHATS
C
CALL MADD(KHAT1, KHAT2, NACT, NACT, KCC)
CALL MADD(KCC, KHAT3, NACT, NACT, BCG)
IF(DEC.EQ.3)THEN
DO 166 I=1, NACT
DO 166 J=1, NACT
166 KCC(I, J)=BCG(I, J)
ELSE
CALL MADD(BCG, KHAT4, NACT, NACT, KCC)
ENDIF
C
C SUMMATION OF KHAT IS NOW IN KCC
C
C THIS SECTION GENERATES THE BLOCK SEGMENTS
C OF MAJM AND PUTS THEM INTO THE MAJM MATRIX
C
C THE FOUR CONTROLLER MATRIX WILL CONTAIN
C RESIDUAL TERMS (SEE DIAGRAM BELOW).
C THE FOUR CONTROLLER MATRIX DOES NOT IN-
C CLUDE RESIDUALS (YET).
C

```

C THE FOUR CONTROLLER MATRIX (MAJM) WITH
C RESIDUAL TERMS WILL LOOK LIKE:

```

C *****
C *
C * A1+B1KIC1      B1KIC2      B1KIC3      B1KIC4      B1KICR *
C *
C * B2KIC1      A2+B2KIC2      B2KIC3      B2KIC4      B2KICR *
C *
C * B3KIC1      B3KIC2      A3+B3KIC3      B3KIC4      B3KICR *
C *
C * B4KIC1      B4KIC2      B4KIC3      A4+B4KIC4      B4KICR *
C *
C * BRKIC1      BRKIC2      BRKIC3      BRKIC4      AR+BRKICR *
C *
C *****

```

```

C
C      K=NC12
C      L=NC22 + K
C      P5=NC32 + L
C      MM = NC12 + NC22 + NC32 + NR2

```

```

C
C      DO 200 I=1,MM
C      DO 200 J=1,MM
200  MAJM(I,J) = 0.0
C      CALL ABGC(A1,B1,C1,NC12,KCC,NACT,NSEN,ABG1)
C      DO 201 I=1,NC12
C      DO 201 J=1,NC12
201  MAJM(I,J) = ABG1(I,J)
C      CALL ABGC(A2,B2,C2,NC22,KCC,NACT,NSEN,ABG2)
C      DO 202 I=1,NC22
C      DO 202 J=1,NC22
202  MAJM(I+K,J+K) = ABG2(I,J)
C      CALL ABGC(A3,B3,C3,NC32,KCC,NACT,NSEN,ABG3)
C      DO 203 I=1,NC32
C      DO 203 J=1,NC32
203  MAJM(I+L,J+L) = ABG3(I,J)
C      CALL MMUL(B1,KCC,NC12,NACT,NACT,TEN)
C      CALL MMUL(TEN,C2,NC12,NACT,NC22,BCG)
C      DO 208 I=1,NC12
C      DO 208 J=1,NC22
208  MAJM(I,J+K) = BCG(I,J)
C      CALL MMUL(B1,KCC,NC12,NACT,NACT,TEN)
C      CALL MMUL(TEN,C3,NC12,NACT,NC32,BCG)
C      DO 209 I=1,NC12
C      DO 209 J=1,NC32
209  MAJM(I,J+L) = BCG(I,J)
C      CALL MMUL(B2,KCC,NC22,NACT,NACT,TEN)
C      CALL MMUL(TEN,C1,NC22,NACT,NC12,BCG)
C      DO 210 I=1,NC22
C      DO 210 J=1,NC12
210  MAJM(I+K,J) = BCG(I,J)

```

```

CALL MMUL(B2,KCC,NC22,NACT,NACT,TEN)
CALL MMUL(TEN,C3,NC22,NACT,NC32,BCG)
DO 212 I=1,NC22
DO 212 J=1,NC32
212 MAJM(I+K,J+L) = BCG(I,J)
CALL MMUL(B3,KCC,NC32,NACT,NACT,TEN)
CALL MMUL(TEN,C1,NC32,NACT,NC12,BCG)
DO 213 I=1,NC32
DO 213 J=1,NC12
213 MAJM(I+L,J) = BCG(I,J)
CALL MMUL(B3,KCC,NC32,NACT,NACT,TEN)
CALL MMUL(TEN,C2,NC32,NACT,NC22,BCG)
DO 214 I=1,NC32
DO 214 J=1,NC22
214 MAJM(I+L,J+K) = BCG(I,J)
CALL MMUL(B4,KCC,NC2,NACT,NACT,TEN)
CALL MMUL(TEN,C1,NC2,NACT,NC12,BCG)
DO 222 I=1,NC2
DO 222 J=1,NC12
222 MAJM(I+P5,J) = BCG(I,J)
CALL MMUL(B4,KCC,NC2,NACT,NACT,TEN)
CALL MMUL(TEN,C2,NC2,NACT,NC22,BCG)
DO 223 I=1,NC2
DO 223 J=1,NC22
223 MAJM(I+P5,J+K) = BCG(I,J)
CALL MMUL(B4,KCC,NC2,NACT,NACT,TEN)
CALL MMUL(TEN,C3,NC2,NACT,NC32,BCG)
DO 224 I=1,NC2
DO 224 J=1,NC32
224 MAJM(I+P5,J+L) = BCG(I,J)
C
C
IF (DEC.EQ.4) THEN
C
C
CALL ABGC(A4,B4,C4,NC2,KCC,NACT,NC2,ABG4)
DO 230 I=1,NC2
DO 230 J=1,NC2
230 MAJM(I+P5,J+P5) = ABG4(I,J)
CALL MMUL(B1,KCC,NC12,NACT,NACT,TEN)
CALL MMUL(TEN,C4,NC12,NACT,NC2,BCG)
DO 232 I=1,NC12
DO 232 J=1,NC2
232 MAJM(I,J+P5) = BCG(I,J)
CALL MMUL(B2,KCC,NC22,NACT,NACT,TEN)
CALL MMUL(TEN,C4,NC22,NACT,NC2,BCG)
DO 233 I=1,NC22
DO 233 J=1,NC2
233 MAJM(I+K,J+P5) = BCG(I,J)
CALL MMUL(B3,KCC,NC32,NACT,NACT,TEN)
CALL MMUL(TEN,C4,NC32,NACT,NC2,BCG)
DO 234 I=1,NC32
DO 234 J=1,NC2

```

```

234 MAJM(I+L, J+P5) = BCG(I, J)
C
ELSE
C
CALL FORMA(A4, D, W, NR, NR2, IR)
CALL ABGC(A4, B4, C4, NR2, KCC, NACT, NSEN, ABG4)
DO 250 I=1, NR2
DO 250 J=1, NR2
250 MAJM(I+P5, J+P5) = ABG4(I, J)
ENDIF
C
C
C NOW WE HAVE THE MAJM MATRIX
C
C
IF (DEC.EQ.4) THEN
PRINT*, ' THE FOUR CONTROLLER MAJOR MATRIX '
ELSE
PRINT*, ' THE THREE CONTROLLER MAJOR MATRIX '
ENDIF
CALL PRNTXL(MAJM, MM, MM)
C
C
C NEXT, FORM E TO THE AT
C
C
PRINT*, ' FOR THE TIME RESPONSE AND THE EIGENVALUE '
* ' ANALYSIS ENTER 1 '
PRINT*, ' FOR ONLY THE EIGENVALUE ANALYSIS ENTER 2 '
READ(8, *) SKIP
IF (SKIP.EQ.2) THEN
PRINT*, '
PRINT*, ' YOU CHOSE "2" SO WE WILL FORGET THE TIME RESPONSE '
GOTO 300
ENDIF
PRINT*, ' YOU WANT BOTH, EH? '
PRINT*, '
PRINT*, ' O.K. HERE THEY ARE! '
PRINT*, '
PRINT*, ' THE TIME RESPONSE IS FIRST... '
DT = 0.1
TOL = 0.001
CALL MEXP(MM, MAJM, DT, EAT2)
PRINT*, ' MEXPOUT '
PRINT' (//)'
PRINT*, ' THE SOLUTION EAT2 IS '
CALL PRNTXL(EAT2, MM, MM)
C
C
C EAT2 IS NOW THE SOLUTION TO E TO THE AT
C
C
CALL FORMX2(XO, INIT)

```

```

      CALL TIME(EAT2,MM,DT,X1,X0,MODE,EAT,WORK,DEC)
300  CONTINUE
C
C
C  EIGENVALUE ANALYSIS SECTION
C
      PRINT' (///) '
      PRINT*, '          OVERALL SYSTEM EIGENVALUES '
      CALL EIGRF(MAJM,MM,NDA,0,Z,TEN,NCOL,WORK,IER)
      PRINT' (//) '
      DO 400 I=1,MM
400  PRINT*, '          ',Z(I)
      PRINT' (//) '
C
      PRINT*, '          EIGENVALUES OF A1 + B1KIC1 '
      CALL EIGRF(ABG1,NC12,NCOL,0,W1,TEN,NCOL,STOR,IER)
      PRINT' (//) '
      DO 401 I=1,NC12
401  PRINT*, '          ',W1(I)
      PRINT' (//) '
C
      PRINT*, '          EIGENVALUES OF A2 + B2KIC2 '
      CALL EIGRF(ABG2,NC22,NCOL,0,W1,TEN,NCOL,STOR,IER)
      PRINT' (//) '
      DO 403 I=1,NC22
403  PRINT*, '          ',W1(I)
      PRINT' (//) '
      PRINT*, '          EIGENVALUES OF A3 + B3KIC3 '
      CALL EIGRF(ABG3,NC32,NCOL,0,W1,TEN,NCOL,STOR,IER)
      PRINT' (//) '
      DO 405 I=1,NC32
405  PRINT*, '          ',W1(I)
      PRINT' (//) '
      IF (NR.EQ.0) THEN
      PRINT*, ' NO RESIDUAL TERM EIGENVALUES '
      PRINT' (//) '
      GOTO 410
      ENDIF
      IF (DEC.EQ.4) THEN
C
      PRINT*, '          EIGENVALUES OF A4 + B4KIC4 '
      CALL EIGRF(ABG4,NR2,NCOL,0,W1,TEN,NCOL,STOR,IER)
      PRINT' (//) '
      DO 407 I=1,NR2
407  PRINT*, '          ',W1(I)
      PRINT' (//) '
      ELSE
C
      PRINT*, '          EIGENVALUES OF THE A RESIDUAL MATRIX '
      CALL EIGRF(A4,NR2,NCOL,0,W1,TEN,NCOL,STOR,IER)
      PRINT' (//) '
      DO 409 I=1,NR2

```

```

409 PRINT*, '          ', W1(I)
      ENDIF
      PRINT '(///)'

```

```

C
410 CONTINUE

```

```

C
C
C THIS SECTION FORMS THE TRANSFORMATION MATRICES.
C

```

```

C
C AFTER THE TRANSFORMATION IS COMPLETE,
C THE THREE OR FOUR CONTROLLER MAJM (WITHOUT RESIDUALS)
C WILL LOOK LIKE:
C

```

```

C
C *****
C *
C *      (A1+B1K1C1)          0          0          0          *
C *
C *      (B2K1C1)          (A2+B2K2C2)      0          0          *
C *
C *      (B3K1C1+B3K2C1)      (B3K2C2)      (A3+B3K3C3)      0          *
C *
C * (B4K1C1+B4K2C1+B4K3C1)  (B4K2C2+B4K3C2)  (B4K3C3)  (A4+B4K4C4) *
C *
C *****
C

```

```

C
C WHERE THE NON-ZERO TERMS INCLUDE THE
C TRANSFORMATION MATRICES.
C (THE RESIDUALS ARE NOT ZEROED SINCE THEY
C ARE NOT ADDRESSED IN THIS PROGRAM)
C

```

```

C
C
C
C
C GENERATE THE TRANSFORMATION MATRICES
C

```

```

C
C      GAMMA1, GAMMA2, GAMMA3      (GAMMA4 = IDENTITY)
C

```

```

C
C      CALL TFR(CT, C2, NSEN, NC22, 1, 2)
C      DO 500 I=1, NC2
C      DO 500 J=1, NSEN
500  V(I, J) = CT(I+NC2, J)
C      CALL TFR(CT, C3, NSEN, NC32, 1, 2)
C      DO 501 I=1, NC3
C      DO 501 J=1, NSEN
501  V(I+NC2, J) = CT(I+NC3, J)
C      IF (DEC.EQ.4) THEN
C      CALL TFR(CT, C4, NSEN, NR2, 1, 2)
C      DO 502 I=1, NR
C      DO 502 J=1, NSEN
502  V(I+NC2+NC3, J) = CT(I+NR, J)
C      NRV = NC2 + NC3 + NR
C      PRINT*, ' V (C2/C3/C4) IS '

```

```

ELSE
PRINT*, ' V (C2/C3) IS '
NRV = NC2 + NC3
ENDIF
CALL PRNT(V, NRV, NSEN)
CALL LSVDF(V, NCOL, NRV, NSEN, TEN, NCOL, -1, SING, STOR, IER)
PRINT*, ' '
PRINT*, ' V OUT OF LSVDF IS '
CALL PRNT(V, NSEN, NSEN)
P1 = NSEN - NRV
IF (P1.LT.1) THEN
DO 503 I=1, NSEN
503 GAMMA1(I, 1) = V(I, NSEN)
P1 = 1
ELSE
DO 504 I=1, NSEN
DO 504 J=1, P1
504 GAMMA1(I, J) = V(I, J+NRV)
ENDIF
PRINT*, ' TRANSFORMATION MATRIX GAMMA1 '
CALL PRNT(GAMMA1, NSEN, P1)
C
C CHECK TO SEE THAT GAMMA1 IS ORTHOGONAL TO BOTH C2 AND C3
C
C NOTE: AKC IN THIS SECTION IS JUST A WORK AREA TO TEST
C THE ORTHOGONALITY OF CT * TR. IN ALL CASES IT
C SHOULD BE A BLOCK ZERO MATRIX.
C
CALL TFR(CT, C2, NSEN, NC22, 1, 2)
CALL MMUL(CT, GAMMA1, NC22, NSEN, P1, AKC)
PRINT*, ' C2T * GAMMA1 '
CALL PRNT(AKC, NC22, P1)
CALL TFR(CT, C3, NSEN, NC32, 1, 2)
CALL MMUL(CT, GAMMA1, NC32, NSEN, P1, AKC)
PRINT*, ' C3T * GAMMA1 '
CALL PRNT(AKC, NC32, P1)
IF (DEC.EQ.4) THEN
CALL TFR(CT, C4, NSEN, NR2, 1, 2)
CALL MMUL(CT, GAMMA1, NR2, NSEN, P1, AKC)
PRINT*, ' C4T * GAMMA1 '
CALL PRNT(AKC, NR2, P1)
ENDIF
C
IF (DEC.EQ.4) THEN
PRINT*, ' C234 SINGULAR VALUES '
ELSE
PRINT*, ' C23 SINGULAR VALUES '
ENDIF
CALL PRNT(SING, NRV, 1)
CALL TFR(TRT, GAMMA1, NSEN, P1, 1, 2)
CALL MMUL(TRT, GAMMA1, P1, NSEN, P1, RK)
CALL GMINV(P1, P1, RK, RK1, J, TAPE)
CALL TFR(CT, C1, NSEN, NC12, 1, 2)

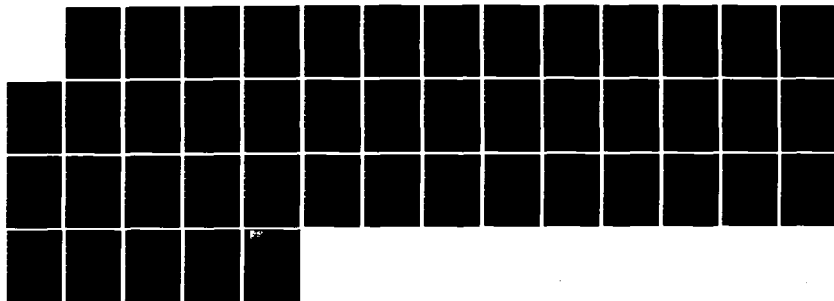
```

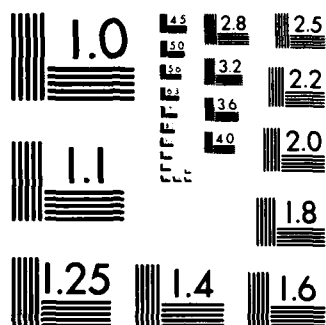
AD-A136 781

DECENTRALIZED CONTROL OF A LARGE SPACE STRUCTURE USING
DIRECT OUTPUT FEEDBACK(U) AIR FORCE INST OF TECH
WRIGHT-PATTERSON AFB OH SCHOOL OF ENGI.. D V THYFAULT
DEC 83 AFIT/GA/AA/83D-8 F/G 12/1

UNCLASSIFIED

NL





MICROCOPY RESOLUTION TEST CHART
NATIONAL BUREAU OF STANDARDS-1963-A

```

CALL MMUL (TRT, C1, P1, NSEN, NC12, AKC)
CALL MMUL (CT, GAMMA1, NC12, NSEN, P1, KOB1)
CALL MMUL (KOB1, RK1, NC12, P1, P1, STOR)
CALL MMUL (STOR, AKC, NC12, P1, NC12, CTCC1)

C
C
C
C
C
C
CALL TFR (CT, C3, NSEN, NC32, 1, 2)
DO 505 I=1, NC3
DO 505 J=1, NSEN
505 V(I, J) = CT(I+NC3, J)
IF (DEC. EQ. 4) THEN
CALL TFR (CT, C4, NSEN, NR2, 1, 2)
DO 506 I=1, NR
DO 506 J=1, NSEN
506 V(I+NC3, J) = CT(I+NR, J)
NRV = NC3 + NR
PRINT*, ' V (C3/C4) IS '
ELSE
NRV = NC3
PRINT*, ' V (C3) IS '
ENDIF
CALL PRNT (V, NRV, NSEN)
CALL LSVDF (V, NCOL, NRV, NSEN, TEN, NCOL, -1, SING, STOR, IER)
PRINT*, ' '
PRINT*, ' V OUT OF LSVDF IS '
CALL PRNT (V, NSEN, NSEN)
P2 = NSEN - NRV
IF (P2.LT.1) THEN
DO 507 I=1, NSEN
507 GAMMA2(I, 1) = V(I, NSEN)
P2 = 1
ELSE
DO 508 I=1, NSEN
DO 508 J=1, P2
508 GAMMA2(I, J) = V(I, J+NRV)
ENDIF
PRINT*, ' TRANSFORMATION MATRIX GAMMA2 '
CALL PRNT (GAMMA2, NSEN, P2)

C
C CHECK TO SEE THAT GAMMA2 IS ORTHOGONAL TO C3
C
CALL TFR (CT, C3, NSEN, NC32, 1, 2)
CALL MMUL (CT, GAMMA2, NC32, NSEN, P2, AKC)
PRINT*, ' C3T * GAMMA2 '
CALL PRNT (AKC, NC32, P2)
IF (DEC. EQ. 4) THEN
CALL TFR (CT, C4, NSEN, NR2, 1, 2)
CALL MMUL (CT, GAMMA2, NR2, NSEN, P2, AKC)
PRINT*, ' C4T * GAMMA2 '
CALL PRNT (AKC, NR2, P2)

```

```

ENDIF
C
IF (DEC.EQ.4) THEN
PRINT*, ' C34 SINGULAR VALUES '
ELSE
PRINT*, ' C3 SINGULAR VALUES '
ENDIF
CALL PRNT(SING, NR, 1)
C
CALL TFR(TRT, GAMMA2, NSEN, P2, 1, 2)
CALL MMUL(TRT, GAMMA2, P2, NSEN, P2, RK)
CALL GMINV(P2, P2, RK, RK2, J, TAPE)
CALL TFR(CT, C2, NSEN, NC22, 1, 2)
CALL MMUL(TRT, C2, P2, NSEN, NC22, AKC)
CALL MMUL(CT, GAMMA2, NC22, NSEN, P2, KOB2)
CALL MMUL(KOB2, RK2, NC22, P2, P2, STOR)
CALL MMUL(STOR, AKC, NC22, P2, NC22, CTCC2)
C
C
C
C
C
C
IF (DEC.EQ.4) THEN
CALL TFR(CT, C4, NSEN, NR2, 1, 2)
DO 509 I=1, NR
DO 509 J=1, NSEN
509 V(I, J) = CT(I+NR, J)
PRINT*, ' V (C4) IS '
CALL PRNT(V, NR, NSEN)
CALL LSVDF(V, NCOL, NR, NSEN, TEN, NCOL, -1, SING, STOR, IER)
PRINT*, ' V OUT OF LSVDF IS '
CALL PRNT(V, NSEN, NSEN)
P3 = NSEN - NR
IF (P3.LT.1) THEN
DO 510 I=1, NSEN
510 GAMMA3(I, 1) = V(I, NSEN)
P3 = 1
ELSE
DO 511 I=1, NSEN
DO 511 J=1, P3
511 GAMMA3(I, J) = V(I, J+NR)
ENDIF
PRINT*, ' TRANSFORMATION MATRIX GAMMA3 '
CALL PRNT(GAMMA3, NSEN, P3)
C
C CHECK TO SEE THAT GAMMA3 IS ORTHOGONAL TO C4
C
CALL MMUL(CT, GAMMA3, NR2, NSEN, P3, AKC)
PRINT*, ' C4T * GAMMA3 '
CALL PRNT(AKC, NR2, P3)
C
PRINT*, ' C4 SINGULAR VALUES '

```

```

CALL PRNT(SING, NR, 1)
CALL TFR(TRT, GAMMA3, NSEN, P3, 1, 2)
CALL MMUL(TRT, GAMMA3, P3, NSEN, P3, RK)
CALL GMINV(P3, P3, RK, RK3, J, TAPE)
CALL TFR(CT, C3, NSEN, NC32, 1, 2)
CALL MMUL(TRT, C3, P3, NSEN, NC32, AKC)
CALL MMUL(CT, GAMMA3, NC32, NSEN, P3, KOB3)
CALL MMUL(KOB3, RK3, NC32, P3, P3, STOR)
CALL MMUL(STOR, AKC, NC32, P3, NC32, CTCC3)
ENDIF
DO 411 I=1, NSEN
DO 411 J=1, P1
411 GT1(J, I)=GAMMA1(I, J)
DO 412 I=1, NSEN
DO 412 J=1, P2
412 GT2(J, I)=GAMMA2(I, J)
PRINT*, ' GAMMA1 TRANSPOSE '
CALL PRNT(GT1, P1, NSEN)
PRINT*, ' GAMMA2 TRANSPOSE '
CALL PRNT(GT2, P2, NSEN)
IF(DEC.EQ.4) THEN
DO 413 I=1, NSEN
DO 413 J=1, P3
413 GT3(J, I)=GAMMA3(I, J)
PRINT*, ' GAMMA3 TRANSPOSE '
CALL PRNT(GT3, P3, NSEN)
ENDIF
C
C CSTAR = GAMMA * C
C
CALL MMUL(GT1, C1, P1, NSEN, NC12, CSTAR1)
PRINT*, ' (GT1)(C1) '
CALL PRNT(CSTAR1, P1, NC12)
CALL MMUL(GT2, C2, P2, NSEN, NC22, CSTAR2)
PRINT*, ' (GT2)(C2) '
CALL PRNT(CSTAR2, P2, NC22)
IF(DEC.EQ.4) THEN
CALL MMUL(GT3, C3, P3, NSEN, NC32, CSTAR3)
PRINT*, ' (GT3)(C3) '
CALL PRNT(CSTAR3, P3, NC32)
ENDIF
C
C C
C TRANSFORMATION MATRICES T2, T3, T4 (T1 = IDENTITY)
C
DO 512 I=1, NC1
DO 512 J=1, NACT
512 V(I, J) = B1(I+NC1, J)
PRINT*, ' V (B1) IS '
CALL PRNT(V, NC1, NACT)
CALL LSVDF(V, NCOL, NC1, NACT, TEN, NCOL, -1, SING, STOR, IER)
PRINT*, ' V OUT OF LSVDF IS '

```

```

CALL PRNT(V, NACT, NACT)
E2 = NACT - NC1
IF (E2.LT.1) THEN
DO 513 I=1, NACT
513 T2(I,1) = V(I, NACT)
E2 = 1
ELSE
DO 514 I=1, NACT
DO 514 J=1, E2
514 T2(I, J) = V(I, J+NC1)
ENDIF
PRINT*, ' TRANSFORMATION MATRIX T2 '
CALL PRNT(T2, NACT, E2)

C
C CHECK TO SEE THAT T2 IS ORTHOGONAL TO B1
C
C NOTE: IN THIS SECTION, BCG IS THE WORK AREA
C FOR B * T. IN ALL CASES THESE SHOULD
C BE BLOCK ZERO MATRICES.
C
C
CALL MMUL(B1, T2, NC12, NACT, E2, BCG)
PRINT*, ' B1 * T2 '
CALL PRNT(BCG, NC12, E2)

C
PRINT*, ' B1 SINGULAR VALUES '
CALL PRNT(SING, NC1, 1)

C
CALL VMULFM(T2, T2, NACT, E2, E2, NACT, NACT, RK, NACT, IER)
CALL GMINV(E2, E2, RK, RG2, J, TAPE)
CALL MMUL(B2, T2, NC22, NACT, E2, KOB2)
CALL MMUL(KOB2, RG2, NC22, E2, E2, SAT2)
CALL VMULFP(SAT2, T2, NC22, E2, NACT, NCOL, NACT, KOB2, NCOL, IER)
CALL VMULFP(KOB2, B2, NC22, NACT, NC22, NCOL, NCOL, SAT2, NCOL, IER)

C
C THIS SAT2 WILL BE SUBSTITUTED BACK INTO MRIC
C SYSTEM 2 FOR A NEW G2.
C
C
C TRANSFORMATION MATRIX T3
C
C
DO 515 I=1, NC1
DO 515 J=1, NACT
515 V(I, J) = B1(I+NC1, J)
DO 516 I=1, NC2
DO 516 J=1, NACT
516 V(I+NC1, J) = B2(I+NC2, J)

C
C V IS NOW AN NC1+NC2 BY NACT MATRIX
C
PRINT*, ' V (B1/B2) IS '

```

```

NRV = NC1 + NC2
CALL PRNT(V, NRV, NACT)
CALL LSVDF(V, NCOL, NRV, NACT, TEN, NCOL, -1, SING, STOR, IER)
PRINT*, ' V OUT OF LSVDF IS '
CALL PRNT(V, NACT, NACT)
E3 = NACT - NRV
IF (E3.LT.1) THEN
DO 517 I =1, NACT
517 T3(I,1) = V(I, NACT)
E3 = 1
ELSE
DO 518 I=1, NACT
DO 518 J=1, E3
518 T3(I, J) = V(I, J+NRV)
ENDIF
PRINT*, ' TRANSFORMATION MATRIX T3 '
CALL PRNT(T3, NACT, E3)
C
C CHECK TO SEE THAT T3 IS ORTHOGONAL TO B1 AND B2
C
CALL MMUL(B1, T3, NC12, NACT, E3, BCG)
PRINT*, ' B1 * T3 '
CALL PRNT(BCG, NC12, E3)
CALL MMUL(B2, T3, NC22, NACT, E3, BCG)
PRINT*, ' B2 * T3 '
CALL PRNT(BCG, NC22, E3)
C
PRINT*, ' B12 SINGULAR VALUES '
CALL PRNT(SING, NRV, 1)
C
CALL VMULFM(T3, T3, NACT, E3, E3, NACT, NACT, RK, NACT, IER)
CALL GMINV(E3, E3, RK, RG3, J, TAPE)
CALL MMUL(B3, T3, NC32, NACT, E3, KOB3)
CALL MMUL(KOB3, RG3, NC32, E3, E3, SAT3)
CALL VMULFP(SAT3, T3, NC32, E3, NACT, NCOL, NACT, KOB3, NCOL, IER)
CALL VMULFP(KOB3, B3, NC32, NACT, NC32, NCOL, NCOL, SAT3, NCOL, IER)
C
C
IF (DEC.EQ.4) THEN
DO 519 I=1, NC1
DO 519 J=1, NACT
519 V(I, J) = B1(I+NC1, J)
DO 520 I=1, NC2
DO 520 J=1, NACT
520 V(I+NC1, J) = B2(I+NC2, J)
DO 521 I=1, NC3
DO 521 J=1, NACT
521 V(I+NC1+NC2, J) = B3(I+NC3, J)
C
C V IS NOW A (NC1+NC2+NC3) BY NACT MATRIX
C
PRINT*, ' V(B1/B2/B3) IS '
NRV = NC1 + NC2 + NC3

```



```

CALL MMUL(B2,T2,NC22,NACT,E2,BSTAR2)
CALL VMULFP(BSTAR2,BSTAR2,NC22,E2,NC22,NCOL,NCOL,SAT2,NCOL,IER)
CALL MRIC(NC22,A2,SAT2,QA2,S,ABG2,TOL,IER)
PRINT*, ' THE EIGENVALUES OF A2 + BSTAR2GSTAR2 '
PRINT'(/)'
CALL EIGRF(ABG2,NC22,NCOL,0,W1,TEN,NCOL,STOR,IER)
DO 714 I=1,NC22
714 PRINT*, ' ',W1(I)
PRINT'(/)'
CALL VMULFM(BSTAR2,S,NC22,E2,NC22,NCOL,NCOL,GSTAR2,NCOL,IER)
PRINT*, ' THE GSTAR2 MATRIX '
CALL PRNT(GSTAR2,E2,NC22)
PRINT'(/)'
CALL FINDK(CSTAR2,P2,NC22,WORK,GSTAR2,E2,KSTAR2)
PRINT*, ' KSTAR2 '
CALL PRNT(KSTAR2,E2,P2)

C
C
CALL MMUL(B3,T3,NC32,NACT,E3,BSTAR3)
CALL VMULFP(BSTAR3,BSTAR3,NC32,E3,NC32,NCOL,NCOL,SAT3,NCOL,IER)
CALL MRIC(NC32,A3,SAT3,QA3,S,ABG3,TOL,IER)
PRINT*, ' THE EIGENVALUES OF A3 + BSTAR3GSTAR3 '
PRINT'(/)'
CALL EIGRF(ABG3,NC32,NCOL,0,W1,TEN,NCOL,STOR,IER)
DO 722 I=1,NC32
722 PRINT*, ' ',W1(I)
PRINT'(/)'
CALL VMULFM(BSTAR3,S,NC32,E3,NC32,NCOL,NCOL,GSTAR3,NCOL,IER)
PRINT*, ' THE GSTAR3 MATRIX '
CALL PRNT(GSTAR3,E3,NC32)
PRINT'(/)'
IF(DEC.EQ.3) THEN
CALL FINDK(C3,NSEN,NC32,WORK,GSTAR3,E3,KSTAR3)
ELSE
CALL FINDK(CSTAR3,P3,NC32,WORK,GSTAR3,E3,KSTAR3)
CALL MMUL(B4,T4,NR2,NACT,E4,BSTAR4)
CALL VMULFP(BSTAR4,BSTAR4,NR2,E4,NR2,NCOL,NCOL,SAT4,NCOL,IER)
CALL MRIC(NR2,A4,SAT4,QA4,S,ABG4,TOL,IER)
CALL VMULFM(BSTAR4,S,NR2,E4,NR2,NCOL,NCOL,GSTAR4,NCOL,IER)
PRINT*, ' EIGENVALUES OF A4 + BSTAR4GSTAR4 '
PRINT'(/)'
CALL EIGRF(ABG4,NR2,NCOL,0,W1,TEN,NCOL,STOR,IER)
DO 420 I=1,NR2
420 PRINT*, ' ',W1(I)
PRINT'(/)'
CALL FINDK(C4,NSEN,NR2,WORK,GSTAR4,E4,KSTAR4)
ENDIF

C
C
FORM TKG = T * KSTAR * GAMMA

C
CALL MMUL(KSTAR1,GT1,NACT,P1,NSEN,TKG1)
PRINT*, ' (KSTAR1)(GT1) '
CALL PRNT(TKG1,NACT,NSEN)

```

```

CALL MMUL (T2, KSTAR2, NACT, E2, P2, BCG)
CALL MMUL (BCG, GT2, NACT, P2, NSEN, TKG2)
PRINT*, ' (T2)(KSTAR2)(GT2) '
CALL PRNT (TKG2, NACT, NSEN)
IF (DEC. EQ. 3) THEN
CALL MMUL (T3, KSTAR3, NACT, E3, NSEN, TKG3)
PRINT*, ' (T3)(KSTAR3) '
CALL PRNT (TKG3, NACT, NSEN)
ELSE
CALL MMUL (T3, KSTAR3, NACT, E3, P3, BCG)
CALL MMUL (BCG, GT3, NACT, P3, NSEN, TKG3)
CALL MMUL (T4, KSTAR4, NACT, E4, NSEN, TKG4)
PRINT*, ' (T3)(KSTAR3)(GT3) '
CALL PRNT (TKG3, NACT, NSEN)
PRINT*, ' (T4)(KSTAR4) '
CALL PRNT (TKG4, NACT, NSEN)
ENDIF

```

```

C
C FORM (B*)(K*)(C*) = (B)(TKG)(C) AND A + (B*)(K*)(C*)
C

```

```

CALL MMUL (B1, KSTAR1, NC12, NACT, P1, BCG)
CALL MMUL (BCG, CSTAR1, NC12, P1, NC12, KCC)
PRINT*, ' (B1)(KSTAR1)(CSTAR1) '
CALL PRNT (KCC, NC12, NC12)
CALL UBOAT (A1, KCC, NC12, NC12, KOB1)
PRINT*, ' A1 + (B1)(KSTAR1)(CSTAR1) '
CALL PRNT (KOB1, NC12, NC12)
CALL MMUL (BSTAR2, KSTAR2, NC22, E2, P2, BCG)
CALL MMUL (BCG, CSTAR2, NC22, P2, NC22, KCC)
PRINT*, ' (BSTAR2)(KSTAR2)(CSTAR2) '
CALL PRNT (KCC, NC22, NC22)
CALL UBOAT (A2, KCC, NC22, NC22, KOB2)
PRINT*, ' A2 + (BSTAR2)(KSTAR2)(CSTAR2) '
CALL PRNT (KOB2, NC22, NC22)
IF (DEC. EQ. 3) THEN
CALL MMUL (BSTAR3, KSTAR3, NC32, E3, NSEN, BCG)
CALL MMUL (BCG, C3, NC32, NSEN, NC32, KCC)
PRINT*, ' (BSTAR3)(KSTAR3)(C3) '
CALL PRNT (KCC, NC32, NC32)
CALL UBOAT (A3, KCC, NC32, NC32, KOB3)
PRINT*, ' A3 + (BSTAR3)(KSTAR3)(C3) '
CALL PRNT (KOB3, NC32, NC32)
ELSE
CALL MMUL (BSTAR3, KSTAR3, NC32, E3, P3, BCG)
CALL MMUL (BCG, CSTAR3, NC32, P3, NC32, KCC)
PRINT*, ' (BSTAR3)(KSTAR3)(CSTAR3) '
CALL PRNT (KCC, NC32, NC32)
CALL UBOAT (A3, KCC, NC32, NC32, KOB3)
PRINT*, ' A3 + (BSTAR3)(KSTAR3)(CSTAR3) '
CALL PRNT (KOB3, NC32, NC32)
CALL MMUL (BSTAR4, KSTAR4, NR2, E4, NSEN, BCG)
CALL MMUL (BCG, C4, NR2, NSEN, NR2, KCC)
PRINT*, ' (BSTAR4)(KSTAR4)(C4) '

```

```

CALL PRNT(KCC, NR2, NR2)
CALL UBOAT(A4, KCC, NR2, NR2, KOB4)
PRINT*, ' A4 + (BSTAR4)(KSTAR4)(C4) '
CALL PRNT(KOB4, NR2, NR2)
ENDIF
CALL EIGRF(KOB1, NC12, NCOL, 0, W1, TEN, NCOL, STOR, IER)
PRINT*, ' EIGENVALUES A1 + (B1)(KSTAR1)(CSTAR1) '
PRINT'(/)'
DO 421 I=1, NC12
421 PRINT*, ' ', W1(I)
PRINT'(/)'
PRINT*, ' EIGENVALUES A2 + (BSTAR2)(KSTAR2)(CSTAR2) '
CALL EIGRF(KOB2, NC22, NCOL, 0, W1, TEN, NCOL, STOR, IER)
PRINT'(/)'
DO 422 I=1, NC22
422 PRINT*, ' ', W1(I)
PRINT'(/)'
IF(DEC.EQ.3) THEN
PRINT*, ' EIGENVALUES A3 + (BSTAR3)(KSTAR3)(C3) '
ELSE
PRINT*, ' EIGENVALUES A3 + (BSTAR3)(KSTAR3)(CSTAR3) '
ENDIF
CALL EIGRF(KOB3, NC32, NCOL, 0, W1, TEN, NCOL, STOR, IER)
PRINT'(/)'
DO 423 I=1, NC32
423 PRINT*, ' ', W1(I)
PRINT'(/)'
IF(DEC.EQ.4) THEN
PRINT*, ' EIGENVALUES A4 + (BSTAR4)(KSTAR4)(C4) '
CALL EIGRF(KOB4, NR2, NCOL, 0, W1, TEN, NCOL, STOR, IER)
PRINT'(/)'
DO 424 I=1, NR2
424 PRINT*, ' ', W1(I)
PRINT'(/)'
ENDIF
CALL ABGC(A1, B1, C1, NC12, TKG1, NACT, NSEN, ABG1)
PRINT*, ' A1 + (B1)(TKG1)(C1) '
CALL PRNT(ABG1, NC12, NC12)
CALL ABGC(A2, B2, C2, NC22, TKG2, NACT, NSEN, ABG2)
PRINT*, ' A2 + (B2)(TKG2)(C2) '
CALL PRNT(ABG2, NC22, NC22)
CALL ABGC(A3, B3, C3, NC32, TKG3, NACT, NSEN, ABG3)
PRINT*, ' A3 + (B3)(TKG3)(C3) '
CALL PRNT(ABG3, NC32, NC32)
PRINT*, ' EIGENVALUES OF A1 + (B1)(TKG1)(C1) '
CALL EIGRF(ABG1, NC12, NCOL, 0, W1, TEN, NCOL, STOR, IER)
PRINT'(/)'
DO 723 I=1, NC12
723 PRINT*, ' ', W1(I)
PRINT'(/)'
PRINT*, ' EIGENVALUES OF A2 + (B2)(TKG2)(C2) '
CALL EIGRF(ABG2, NC22, NCOL, 0, W1, TEN, NCOL, STOR, IER)
PRINT'(/)'

```

```

DO 724 I=1,NC22
724 PRINT*, ' ', W1(I)
PRINT'()'
PRINT*, ' EIGENVALUES OF A3 + (B3)(TKG3)(C3) '
CALL EIGRF(ABG3,NC32,NCOL,0,W1,TEN,NCOL,STOR,IER)
PRINT'()'
DO 725 I=1,NC32
725 PRINT*, ' ', W1(I)
PRINT'()'
IF(DEC.EQ.4) THEN
CALL ABGC(A4,B4,C4,NR2,TKG4,NACT,NSEN,ABG4)
PRINT*, ' A4 + (B4)(TKG4)(C4) '
CALL PRNT(ABG4,NR2,NR2)
PRINT*, ' EIGENVALUES A4 + (B4)(TKG4)(C4) '
CALL EIGRF(ABG4,NR2,NCOL,0,W1,TEN,NCOL,STOR,IER)
PRINT'()'
DO 726 I=1,NR2
726 PRINT*, ' ', W1(I)
ENDIF
PRINT'()'
IF(NR.EQ.0.OR.DEC.EQ.4) THEN
GOTO 600
ELSE
DO 348 I=1,NACT
DO 348 J=1,NSEN
348 SAT(I,J)=0.0
C
C KT = -TKG
C
CALL UBOAT(SAT,TKG1,NACT,NSEN,KT1)
CALL UBOAT(SAT,TKG2,NACT,NSEN,KT2)
CALL UBOAT(SAT,TKG3,NACT,NSEN,KT3)
DO 350 I=1,M
DO 350 J=1,M
350 MAJM(I,J)=0.0
CALL FORMA(A1,D,W,NC1,NC12,IC1)
CALL FORMB(B1,PHIA,NC1,NC12,NACT,IC1)
CALL FORMC(C1,PHIS,NC1,NC12,NSEN,IC1)
CALL ABGC(A1,B1,C1,NC12,TKG1,NACT,NSEN,ABG5)
PRINT*, ' A1 + B1(TKG1)C1 '
CALL PRNT(ABG5,NC12,NC12)
DO 351 I=1,NC12
DO 351 J=1,NC12
351 MAJM(I,J)=ABG5(I,J)
CALL FORMA(A2,D,W,NC2,NC22,IC2)
CALL FORMB(B2,PHIA,NC2,NC22,NACT,IC2)
CALL FORMC(C2,PHIS,NC2,NC22,NSEN,IC2)
CALL ABGC(A2,B2,C2,NC22,TKG2,NACT,NSEN,ABG6)
PRINT*, ' A2 + B2(TKG2)C2 '
CALL PRNT(ABG6,NC22,NC22)
DO 352 I=1,NC22
DO 352 J=1,NC22
352 MAJM(I+K,J+K)=ABG6(I,J)

```

```

CALL FORMA(A3, D, W, NC3, NC32, IC3)
CALL FORMB(B3, PHIA, NC3, NC32, NACT, IC3)
CALL FORMC(C3, PHIS, NC3, NC32, NSEN, IC3)
CALL ABGC(A3, B3, C3, NC32, TKG3, NACT, NSEN, ABG7)
PRINT*, ' A3 + B3(TKG3)C3 '
CALL PRNT(ABG7, NC32, NC32)
DO 353 I=1, NC32
DO 353 J=1, NC32
353 MAJM(I+L, J+L)=ABG7(I, J)
CALL MMUL(B2, KT1, NC22, NACT, NSEN, BCG)
CALL MMUL(BCG, C1, NC22, NSEN, NC12, KCC)
PRINT*, ' B2(TKG1)C1 '
CALL PRNT(KCC, NC22, NC12)
DO 354 I=1, NC22
DO 354 J=1, NC12
354 MAJM(I+K, J)=KCC(I, J)
CALL MMUL(B3, KT1, NC32, NACT, NSEN, BCG)
CALL MMUL(BCG, C1, NC32, NSEN, NC12, KOB1)
CALL MMUL(B3, KT2, NC32, NACT, NSEN, BCG)
CALL MMUL(BCG, C1, NC32, NSEN, NC12, KOB2)
CALL MADD(KOB1, KOB2, NC32, NC12, KCC)
PRINT*, ' B3(TKG1)C1 + B3(TKG2)C1 '
CALL PRNT(KCC, NC32, NC12)
DO 355 I=1, NC32
DO 355 J=1, NC12
355 MAJM(I+L, J)=KCC(I, J)
CALL MMUL(B3, KT2, NC32, NACT, NSEN, BCG)
CALL MMUL(BCG, C2, NC32, NSEN, NC22, KCC)
PRINT*, ' B3(TKG2)C2 '
CALL PRNT(KCC, NC32, NC22)
DO 356 I=1, NC32
DO 356 J=1, NC22
356 MAJM(I+L, J+K)=KCC(I, J)
CALL MMUL(B4, KT1, NR2, NACT, NSEN, BCG)
CALL MMUL(BCG, C1, NR2, NSEN, NC12, KOB1)
CALL MMUL(B4, KT2, NR2, NACT, NSEN, BCG)
CALL MMUL(BCG, C1, NR2, NSEN, NC12, KOB2)
CALL MMUL(B4, KT3, NR2, NACT, NSEN, BCG)
CALL MMUL(BCG, C1, NR2, NSEN, NC12, KOB3)
CALL MADD(KOB1, KOB2, NR2, NC12, BCG)
CALL MADD(BCG, KOB3, NR2, NC12, KOB4)
PRINT*, ' B4(TKG1)C1 + B4(TKG2)C1 + B4(TKG3)C1 '
CALL PRNT(KOB4, NR2, NC12)
DO 357 I=1, NR2
DO 357 J=1, NC12
357 MAJM(I+P5, J)=KOB4(I, J)
CALL MMUL(B4, KT2, NR2, NACT, NSEN, BCG)
CALL MMUL(BCG, C2, NR2, NSEN, NC22, KOB2)
CALL MMUL(B4, KT3, NR2, NACT, NSEN, BCG)
CALL MMUL(BCG, C2, NR2, NSEN, NC22, KOB3)
CALL MADD(KOB2, KOB3, NR2, NC22, KOB4)
PRINT*, ' B4(TKG2)C2 + B4(TKG3)C2 '
CALL PRNT(KOB4, NR2, NC22)

```

```

      DO 358 I=1, NR2
      DO 358 J=1, NC22
358  MAJM(I+P5, J+K)=KOB4(I, J)
      CALL MMUL(B4, KT3, NR2, NACT, NSEN, KOB1)
      CALL MMUL(KOB1, C3, NR2, NSEN, NC32, KOB2)
      PRINT*, '  B4(TKG3)C3 '
      CALL PRNT(KOB2, NR2, NC32)
      DO 359 I=1, NR2
      DO 359 J=1, NC32
359  MAJM(I+P5, J+L)=KOB2(I, J)
      CALL MADD(KT1, KT2, NACT, NSEN, BCG)
      CALL MADD(BCG, KT3, NACT, NSEN, KCC)
      CALL MMUL(KCC, C4, NACT, NSEN, NR2, BCG)
      CALL MMUL(B1, BCG, NC12, NACT, NR2, ABK1)
      PRINT*, '  B1(TKG1 + TKG2 + TKG3)C4 '
      CALL PRNT(ABK1, NC12, NR2)
      DO 361 I=1, NC12
      DO 361 J=1, NR2
361  MAJM(I, J+P5)=ABK1(I, J)
      CALL MMUL(B2, BCG, NC22, NACT, NR2, ABK2)
      PRINT*, '  B2(TKG1 + TKG2 + TKG3)C4 '
      CALL PRNT(ABK2, NC22, NR2)
      DO 362 I=1, NC22
      DO 362 J=1, NR2
362  MAJM(I+K, J+P5)=ABK2(I, J)
      CALL MMUL(B3, BCG, NC32, NACT, NR2, ABK3)
      PRINT*, '  B3(TKG1 + TKG2 + TKG3)C4 '
      CALL PRNT(ABK3, NC32, NR2)
      DO 363 I=1, NC32
      DO 363 J=1, NR2
363  MAJM(I+L, J+P5)=ABK3(I, J)
      CALL MADD(TKG1, TKG2, NACT, NSEN, BCG)
      CALL MADD(BCG, TKG3, NACT, NSEN, KCC)
      CALL FORMA(A4, D, W, NR, NR2, IR)
      CALL FORMB(B4, PHIA, NR, NR2, NACT, IR)
      CALL FORMC(C4, PHIS, NR, NR2, NSEN, IR)
      CALL ABGC(A4, B4, C4, NR2, KCC, NACT, NSEN, ABG8)
      PRINT*, '  A4 + B4(TKG1 + TKG2 + TKG3)C4 '
      CALL PRNT(ABG8, NR2, NR2)
      DO 365 I=1, NR2
      DO 365 J=1, NR2
365  MAJM(I+P5, J+P5)=ABG8(I, J)
      PRINT*, '          LOWER TRIANGLE THREE CONTROLLER MAJOR MATRIX '
      PRINT*, '          WITH RESIDUALS '
      CALL PRNT(MAJM, M, M)
      PRINT' (///) '
      PRINT*, '          TRANSFORMED SYSTEM EIGENVALUES '
      CALL EIGRF(MAJM, M, NDA, 0, Z, TEN, NCOL, WORK, IER)
      PRINT' (/) '
      DO 366 I=1, M
366  PRINT*, '          ', Z(I)
      PRINT' (///) '
      ENDIF

```

```

600  CONTINUE
C
C
C
PRINT' (///)'
PRINT*, ' *****'
PRINT*, ' **'
PRINT*, ' **      END      DOFB335      PROGRAM      **'
PRINT*, ' **'
PRINT*, ' *****'
PRINT' (///)'
END
SUBROUTINE FORMX2(X0, INIT)
C
C
C FORMX2 IS THE SAME FOR THREE OR FOUR
C CONTROLLERS SINCE THE RESIDUALS BECOME
C THE FOURTH CONTROLLER.
C
COMMON/NUM/IC1(21), IC2(21), IC3(21), IR(21), NC1, NC2, NC3, NR
REAL X0(44), INIT(4, 12)
INTEGER M, I, J, K, L
DO 1 I=1, NC1
M = IC1(I)
X0(I) = INIT(1, M)
X0(I+NC1) = INIT(2, M)
X0(I+NC1*2) = INIT(3, M)
1 X0(I+NC1*3) = INIT(4, M)
J = NC1*4
DO 2 I=1, NC2
M = IC2(I)
X0(I+J) = INIT(1, M)
X0(I+J+NC2) = INIT(2, M)
X0(I+J+NC2*2) = INIT(3, M)
2 X0(I+J+NC2*3) = INIT(4, M)
K = J + NC2*4
DO 3 I=1, NC3
M = IC3(I)
X0(I+K) = INIT(1, M)
X0(I+K+NC3) = INIT(2, M)
X0(I+K+NC3*2) = INIT(3, M)
3 X0(I+K+NC3*3) = INIT(4, M)
L = K + NC3*4
DO 4 I=1, NR
M = IR(I)
X0(I+L) = INIT(1, M)
X0(I+L+NR) = INIT(2, M)
X0(I+L+NR*2) = INIT(3, M)
4 X0(I+L+NR*3) = INIT(4, M)
END
SUBROUTINE FACTOR(N, A, S, MR)
C
A=S'S
DIMENSION A(1), S(1)

```

```

COMMON/MAINB/ NCOL,NCOL1
COMMON/INOUT/KOUT
TOL=1.E-6
MR=0
NN=N*NCOL
TOL1=0.
DO 1 I=1,NN,NCOL1
R=ABS(A(I))
1 IF (R.GT.TOL1) TOL1=R
TOL1=TOL1*1.E-12
II=1
DO 50 I=1,N
IM1=I-1
DO 5 JJ=I,NN,NCOL
5 S(JJ)=0.
ID=II+IM1
R=A(ID)-DOT(IM1,S(II),S(II))
IF (ABS(R).LT.(TOL*A(ID)+TOL1)) GO TO 50
IF (R) 15,50,20
15 MR=-1
WRITE(KOUT,1000)
1000 FORMAT(37H0TRIED TO FACTOR AN INDEFINITE MATRIX )
RETURN
20 S(ID)=SQRT(R)
MR=MR+1
IF (I.EQ.N) RETURN
L=II+NCOL
DO 25 JJ=L,NN,NCOL
IJ=JJ+IM1
25 S(IJ)=(A(IJ)-DOT(IM1,S(II),S(JJ)))/S(ID)
50 II=II+NCOL
RETURN
END
SUBROUTINE FORMA(A,D,W,N,N2,IC)
COMMON/MAINB/NCOL
REAL A(NCOL,NCOL),W(17),D(17)
INTEGER IC(N),I,J,N,M
DO 1 I=1,N2
DO 1 J=1,N2
A(I,J)=0.0
1 CONTINUE
DO 2 I=1,N
M= IC(I)
A((I+N),(I+N))=D(M)
A(I,(I+N)) = 1.0
A((I+N),I) = -(W(M)**2)
2 CONTINUE
RETURN
END
SUBROUTINE FORMB(B,PHI,N,N2,NACT,IC)
COMMON/MAINB/NCOL
REAL B(NCOL,NCOL),PHI(NCOL,NCOL)
INTEGER IC(N),NACT,N,M,I,J

```

```

DO 1 I=1,N2
DO 1 J=1,NACT
B(I,J) = 0.0
1 CONTINUE
DO 2 I=1,N
M = IC(I)
DO 2 J=1,NACT
B((N+I),J) = PHI(M,J)
2 CONTINUE
RETURN
END
SUBROUTINE FORMC(C,PHIS,N,N2,NSEN,IC)
COMMON/MAINB/NCOL
REAL C(NCOL,NCOL),PHIS(NCOL,NCOL)
INTEGER IC(N),M,NSEN,N,N2,I,J
DO 1 I=1,NSEN
DO 1 J=1,N2
C(I,J) = 0.0
1 CONTINUE
DO 2 I=1,NSEN
DO 2 J=1,N
M = IC(J)
C(I,N+J)=PHIS(M,I)
2 CONTINUE
RETURN
END
SUBROUTINE FORMQ(Q,A,N,IC)
COMMON/MAINB/NCOL
REAL A(NCOL),Q(NCOL,NCOL)
INTEGER I,J,K,M,N,N2,IC(NCOL)
N2 = N * 2
DO 1 I=1,N2
DO 1 J=1,N2
Q(I,J) = 0.0
1 CONTINUE
DO 2 I=1,N
J = 1
M = IC(I)
DO 3 K=I-1,I
3 Q(I+K,J+K) = A(M)
2 CONTINUE
RETURN
END
SUBROUTINE GMINV(NR,NC,A,U,MR,MT)
DIMENSION A(1),U(1)
COMMON/MAIN1/ NDIM,NDIM1,S(1)
COMMON/MAINB/NCOL,NCOL1
COMMON/INOUT/KOUT
TOL=1.E-12
MR=NC
NRM1=NR-1
TOL1=1.E-20
JJ=1

```

```

DO 100 J=1, NC
FAC=DOT(NR, A(JJ), A(JJ))
JRM=J-1
JRM=JJ+NRM1
JCM=JJ+JRM1
DO 20 I=JJ, JCM
20 U(I)=0.
U(JCM)=1.0
IF (J.EQ.1) GO TO 54
KK=1
DO 30 K=1, JRM1
IF (S(K).EQ.1.0) GO TO 30
TEMP=-DOT(NR, A(JJ), A(KK))
CALL VADD(K, TEMP, U(JJ), U(KK))
30 KK=KK+NCOL
DO 50 L=1, 2
KK=1
DO 50 K=1, JRM1
IF (S(K).EQ.0.) GO TO 50
TEMP=-DOT(NR, A(JJ), A(KK))
CALL VADD(NR, TEMP, A(JJ), A(KK))
CALL VADD(K, TEMP, U(JJ), U(KK))
50 KK=KK+NCOL
TOL1=TOL*FAC
FAC=DOT(NR, A(JJ), A(JJ))
54 IF (FAC.GT.TOL1) GO TO 70
DO 55 I=JJ, JRM
55 A(I)=0.
S(J)=0.
KK=1
DO 65 K=1, JRM1
IF (S(K).EQ.0.) GO TO 65
TEMP=-DOT(K, U(KK), U(JJ))
CALL VADD(NR, TEMP, A(JJ), A(KK))
65 KK=KK+NCOL
FAC=DOT(J, U(JJ), U(JJ))
MR=MR-1
GO TO 75
70 S(J)=1.0
KK=1
DO 72 K=1, JRM1
IF (S(K).EQ.1.) GO TO 72
TEMP=-DOT(NR, A(JJ), A(KK))
CALL VADD(K, TEMP, U(JJ), U(KK))
72 KK=KK+NCOL
75 FAC=1./SQRT(FAC)
DO 80 I=JJ, JRM
80 A(I)=A(I)*FAC
DO 85 I=JJ, JCM
85 U(I)=U(I)*FAC
100 JJ=JJ+NCOL
IF (MR.EQ.NR.OR.MR.EQ.NC) GO TO 120
IF (MT.NE.0) WRITE(KOUT, 110)NR, NC, MR

```

```

110 FORMAT(13,1HX,I2,8H M RANK,I2)
120 NEND=NC*NCOL
    JJ=1
    DO 135 J=1,NC
    DO 125 I=1,NR
    II=I-J
    S(I)=0.
    DO 125 KK=JJ,NEND,NCOL
125 S(I)=S(I)+A(II+KK)*U(KK)
    II=J
    DO 130 I=1,NR
    U(II)=S(I)
130 II=II+NCOL
135 JJ=JJ+NCOL1
    RETURN
    END
    SUBROUTINE GMINV1(NR,NC,A,U,MR,MT)
    DIMENSION A(1),U(1)
    COMMON/MAIN1/ NDM1,NDIM1,S(1)
    COMMON/MAINB/NCOL,NCOL1
    COMMON/MAINA/NDA,NDA1
    COMMON/INOUT/KOUT
    TOL=1.E-12
    MR=NC
    NRM1=NR-1
    TOL1=1.E-20
    JJ=1
    DO 100 J=1,NC
    FAC=DOT(NR,A(JJ),A(JJ))
    JM1=J-1
    JRM=JJ+NRM1
    JCM=JJ+JM1
    DO 20 I=JJ,JCM
20 U(I)=0.
    U(JCM)=1.0
    IF (J.EQ.1) GO TO 54
    KK=1
    DO 30 K=1,JM1
    IF (S(K).EQ.1.0) GO TO 30
    TEMP=-DOT(NR,A(JJ),A(KK))
    CALL VADD(K,TEMP,U(JJ),U(KK))
30 KK=KK+NDA
    DO 50 L=1,2
    KK=1
    DO 50 K=1,JM1
    IF (S(K).EQ.0.) GO TO 50
    TEMP=-DOT(NR,A(JJ),A(KK))
    CALL VADD(NR,TEMP,A(JJ),A(KK))
    CALL VADD(K,TEMP,U(JJ),U(KK))
50 KK=KK+NDA
    TOL1=TOL*FAC
    FAC=DOT(NR,A(JJ),A(JJ))
54 IF (FAC.GT.TOL1) GO TO 70

```

```

DO 55 I=JJ, JRM
55  A(I)=0.
   S(J)=0.
   KK=1
   DO 65 K=1, JMI
   IF (S(K).EQ.0.) GO TO 65
   TEMP=-DOT(K, U(KK), U(JJ))
   CALL VADD(NR, TEMP, A(JJ), A(KK))
65  KK=KK+NDA
   FAC=DOT(J, U(JJ), U(JJ))
   MR=MR-1
   GO TO 75
70  S(J)=1.0
   KK=1
   DO 72 K=1, JMI
   IF (S(K).EQ.1.) GO TO 72
   TEMP=-DOT(NR, A(JJ), A(KK))
   CALL VADD(K, TEMP, U(JJ), U(KK))
72  KK=KK+NDA
75  FAC=1./SQRT(FAC)
   DO 80 I=JJ, JRM
80  A(I)=A(I)*FAC
   DO 85 I=JJ, JCM
85  U(I)=U(I)*FAC
100 JJ=JJ+NDA
   IF (MR.EQ.NR.OR.MR.EQ.NC) GO TO 120
   IF (MT.NE.0) WRITE(KOUT, 110)NR, NC, MR
110 FORMAT(I3, 1HX, I2, 8H M RANK, I2)
120 NEND=NC*NDA
   JJ=1
   DO 135 J=1, NC
   DO 125 I=1, NR
   II=I-J
   S(I)=0.
   DO 125 KK=JJ, NEND, NDA
125  S(I)=S(I)+A(II+KK)*U(KK)
   II=J
   DO 130 I=1, NR
   U(II)=S(I)
130 II=II+NDA
135 JJ=JJ+NDA1
   RETURN
   END
SUBROUTINE INTEG(N, A, C, S, T)
C  S=INTEGRAL EA*C*EA FROM 0 TO T
C  C IS DESTROYED
   DIMENSION A(1), C(1), S(1)
   COMMON/MAIN1/ NDIM, NDIM1, X(1)
   COMMON/MAINB/NCOL, NCOL1
   COMMON/MAIN2/COEF(100)
   NN=N*NCOL
   NM1=N-1
   IND=0

```

```

ANORM=XNORM(N,A)
DT=T
5  IF (ANORM*ABS(DT).LE.0.5) GO TO 10
    DT=DT/2.
    IND=IND+1
    GO TO 5
10  DO 15 I=1,NN,NCOL
    J=I+NM1
    DO 15 JJ=I,J
15  S(JJ)=DT*C(JJ)
    T1=DT**2/2.
    DO 25 IT=3,15
    CALL MMUL(A,C,N,N,N,X)
    DO 20 I=1,N
    II=(I-1)*NCOL
    DO 20 JJ=I,NN,NCOL
    II=II+1
    C(JJ)=(X(JJ)+X(II))*T1
20  S(JJ)=S(JJ)+C(JJ)
25  T1=DT/FLOAT(IT)
    IF (IND.EQ.0) GO TO 100
    COEF(11)=1.0
    DO 30 I=1,10
    II=11-I
30  COEF(II)=DT*COEF(II+1)/FLOAT(I)
    II=1
    DO 40 I=1,NN,NCOL
    J=I+NM1
    DO 35 JJ=I,J
35  X(JJ)=A(JJ)*COEF(1)
    X(II)=X(II)+COEF(2)
40  II=II+NCOL1
    DO 55 L=3,11
    CALL MMUL(A,X,N,N,N,C)
    II=1
    T1=COEF(L)
    DO 55 I=1,NN,NCOL
    J=I+NM1
    DO 50 JJ=I,J
50  X(JJ)=C(JJ)
    X(II)=X(II)+T1
55  II=II+NCOL1
C  X=EXP(A*DT)
    L=0
60  L=L+1
    CALL MMUL(X,S,N,N,N,C)
    II=1
    DO 90 I=1,N
    J=II
    IF (I.EQ.1) GO TO 75
    DO 70 JJ=I,II,NCOL
    S(JJ)=S(J)
70  J=J+1

```

```

75  DO 85 JJ=I,N
    KK=JJ
    DO 80 K=I,NN,NCOL
      S(J)=S(J)+C(K)*X(KK)
80  KK=KK+NCOL
85  J=J+NCOL
    DO 87 JJ=I,NN,NCOL
87  C(JJ)=X(JJ)
90  II=II+NCOL
    IF (L.EQ.IND) GO TO 100
    CALL MMUL(C,C,N,N,N,X)
    GO TO 60
100 CONTINUE
    RETURN
    END
    SUBROUTINE MEXP(N,A,T,EA)
    DIMENSION A(3364),EA(3364),C(100),D(101),E(100)
    COMMON/MAINA/NDA,NDA1
    COMMON/MAIN1/NDIM,NDIM1,TEN,X(3364)
    COMMON/MAINB/NCOL,NCOL1
    NN=NDA*N
    NM1=N-1
    PRINT*,' MEXPIN '
    IF (N.GT.1) GO TO 5
    EA(1)=EXP(T*A(1))
    RETURN
5   W=1.0
    DO 10 I=1,NN,NDA
      IL=I+NM1
      DO 10 J=I,IL
10  EA(J)=A(J)
      C1=XNORM1(N,A)
      IND=0
      L=1
      T1=T
15  IF (ABS(T1*C1).LE.3.0) GO TO 20
      T1=T1/2.
      IND=IND+1
      GO TO 15
20  C2=0.
      DO 25 I=1,NN,NDA1
25  C2=C2-EA(I)
      C2=C2/FLOAT(L)
      C(L)=C2
      D(L+1)=0.
      II=N+1-L
      E(II)=W
      II=1
      DO 35 I=1,NN,NDA
        IL=I+NM1
        DO 30 J=I,IL
30  X(J)=EA(J)
      X(II)=X(II)+C2

```

```

35 II=II+NDA1
   IF (L.EQ.N) GO TO 40
   CALL MMUL(X,A,N,N,N,EA)
   W=W*T1/FLOAT(L)
   L=L+1
   GO TO 20
40 CONTINUE
C***** CAN CHECK X 0 FOR ACCURACY
   J=N+25
   DO 50 L=N,J
   DO 45 K=1,N
   D(K)=(D(K+1)-W*C(K))*T1/FLOAT(L)
45 E(K)=E(K)+D(K)
50 W=D(1)
   II=1
   DO 60 I=1,NN,NDA
   IL=I+NM1
   DO 55 J=I,IL
55 EA(J)=E(1)*A(J)
   EA(II)=EA(II)+E(2)
60 II=II+NDA1
   IF (N.EQ.2) GO TO 85
   DO 80 L=3,N
   CALL MMUL(EA,A,N,N,N,X)
   II=1
   C2=E(L)
   DO 75 I=1,NN,NDA
   IL=I+NM1
   DO 70 J=I,IL
70 EA(J)=X(J)
   EA(II)=EA(II)+C2
75 II=II+NDA1
80 CONTINUE
85 IF (IND.EQ.0) RETURN
   DO 100 L=1,IND
   DO 90 I=1,NN,NDA
   IL=I+NM1
   DO 90 J=I,IL
90 X(J)=EA(J)
100 CALL MMUL(X,X,N,N,N,EA)
   RETURN
   END
SUBROUTINE MLINEQ(N,A,C,X,TOL,IER)
C SOLVES A'X+XA+C=0
C A AND X CAN BE IN SAME LOCATION
C ANSWER RETURNED IN C AND X
   DIMENSION A(1),C(1),X(1)
   COMMON/MAINB/ NCOL, NCOL1
   COMMON/MAIN3/F(1)
   ADV=TOL*1.E-6
   DT=.5
   DT1=0.
   NN=N*NCOL

```

```

      DO 5 II=1,NN,NCOL1
5      DT1=DT1-A(II)
      DT1=DT1/N
      IF (DT1.GT.4.0) DT=DT*4.0/DT1
      II=1
      DO 20 I=1,N
      DO 15 JJ=I,NN,NCOL
15      X(JJ)=DT*A(JJ)
      X(II)=X(II)-.5
20      II=II+NCOL1
      CALL GMINV(N,N,X,F,MR,0)
      IER=4
      IF (MR.NE.N) RETURN
      CALL MMUL(C,F,N,N,N,X)
C      INITIALIZATION OF X,F
      I=1
      DO 40 II=1,NN,NCOL
      J=II
      IF (I.EQ.1) GO TO 30
      DO 25 JJ=I,II,NCOL
      C(J)=C(JJ)
25      J=J+1
30      ID=J
      DO 35 JJ=II,NN,NCOL
      C(J)=DT*DOT(N,F(II),X(JJ))
35      J=J+1
      F(ID)=F(ID)+1.0
40      I=I+1
      DO 90 IT=1,20
      NEZ=0
      CALL MMUL(C,F,N,N,N,X)
      I=1
      II=1
      J=1
      GO TO 70
60      J=II
      DO 65 JJ=I,II,NCOL
      C(J)=C(JJ)
65      J=J+1
70      ID=J
      DT1=C(J)
      DO 75 JJ=II,NN,NCOL
      C(J)=C(J)+DOT(N,F(II),X(JJ))
75      J=J+1
      J=J-1
      DO 80 JJ=II,J
80      X(JJ)=F(JJ)
      IF (ABS(C(ID)).GT.1.E150) GO TO 95
      IF (ABS(C(ID)-DT1).LT.(ADV+TOL*ABS(C(ID)))) NEZ=NEZ+1
      I=I+1
      II=II+NCOL
      IF (I.LE.N) GO TO 60
      IF (NEZ.EQ.N) GO TO 150

```

```

      CALL MMUL(X,X,N,N,N,F)
90  CONTINUE
95  IER=1
      RETURN
150 CONTINUE
      NM1=N-1
      DO 155 I=1,NN,NCOL
        II=I+NM1
        DO 155 JJ=I,II
155  X(JJ)=C(JJ)
      IER=0
      RETURN
      END
      SUBROUTINE MMUL(X,Y,N1,N2,N3,Z)
      COMMON /MAINB/ NCOL
      DIMENSION X(NCOL,1),Y(NCOL,1),Z(NCOL,1)
      DO 3 J=1,N3
        DO 2 I=1,N1
          S=0.
          DO 1 K=1,N2
1          S=S+X(I,K)*Y(K,J)
2          Z(I,J)=S
3          CONTINUE
        END
      SUBROUTINE MRIC(N,A,S,Q,X,Z,TOL,IER)
      DIMENSION A(1),S(1),Q(1),X(1),Z(1)
      COMMON/MAIN1/NDIM,NDIM1,F(1)
      COMMON/MAINB/NCOL,NCOL1
      COMMON/MAIN2/TR(1)
      COMMON/INOUT/KOUT
      ADV=TOL*1.E-6
      NN=N*NCOL
      NM1=N-1
      IND=1
      COUNT=0.
      IF (IER.EQ.1) COUNT=99.
      IF (IER.EQ.1) MR=N
      IF (IER.EQ.1) GO TO 100
      T1=-1.
300 CONTINUE
      IER=0
      COUNT=COUNT+1.
      DO 15 I=1,N
        DO 15 J=I,NN,NCOL
15      X(J)=-S(J)
        CALL INTEG(N,A,X,Z,T1)
        CALL FACTOR(N,Z,X,MR)
        IER=1
        IF (MR.LT.0) GO TO 200
        IER=0
        CALL GMINV(N,N,X,Z,MR,0)
        CALL TFR(TR,Z,N,N,1,2)
        CALL MMUL(Z,TR,N,N,N,X)

```

```

DO 18 II=1,NN,NCOL1
  I=II
DO 17 J=II,NN,NCOL
  X(J)=(X(J)+X(I))/2.
  X(1)=X(J)
17  I=I+1
18  CONTINUE
100 CONTINUE
DO 16 I=1,N
16  TR(I)=-1.0
C    A+5X IS STABLE
C    POSSIBLE UNCONTROLLABILITY IF MR.NE.N
C    JIM DILLOW IS A NUTTY MATH PROF
  TOL1=TOL/10.
  MAXIT=40
DO 40 IT=1,MAXIT
  IF (IER.EQ.1) GO TO 101
  CALL MMUL(S,X,N,N,N,F)
  CALL MMUL(X,F,N,N,N,Z)
DO 20 I=1,NN,NCOL
  II=I+NM1
DO 20 J=I,II
  X(J)=A(J)-F(J)
20  Z(J)=Z(J)+Q(J)
101 CONTINUE
  IER=0
  CALL MLINEQ(N,X,Z,X,TOL1,IER)
  IF (IER.NE.0) GO TO 200
  L=0
  C1=0.0
  II=1
DO 25 I=1,N
  IF (ABS(X(II)-TR(I)).LT.(ADV+TOL*X(II))) L=L+1
  TR(I)=X(II)
  II=II+NCOL1
25  C1=C1+TR(I)
  IF (ABS(C1).GT.1.E20) GO TO 50
  IF (L.NE.N) GO TO 40
  CALL GMINV(N,N,Z,F,MR,0)
  CALL MMUL(S,X,N,N,N,Z)
DO 30 I=1,NN,NCOL
  II=I+NM1
DO 30 J=I,II
30  Z(J)=A(J)-Z(J)
  IF (MR.NE.N) WRITE(KOUT,35)MR
35  FORMAT(27H0RICCATI SOLN IS PSD--RANK ,I3)
  GO TO 65
40  CONTINUE
  WRITE(KOUT,45) MAXIT
45  FORMAT(27H0RICCATI NON-CONVERGENT IN ,I2,11H ITERATIONS)
  GO TO 60
50  WRITE(KOUT,55)IT,T1
55  FORMAT(30H0RICCATI BLOW-UP AT ITERATION ,I2,12H INITIAL T= ,F10.5)

```

```

60  IER=1
65  RETURN
200 IF (IND.EQ.2) GO TO 250
    IF (COUNT.GE.10.) RETURN
    T1=T1/(2.**COUNT)
    IND=2
    GO TO 300
250 T1=T1*(2.**COUNT)
    IND=1
    GO TO 300
    END
    SUBROUTINE PRNT(MAT,N,M)
    COMMON/MAINB/NCOL
    REAL MAT(NCOL,NCOL)
    INTEGER N,I,J,K,M
    PRINT*, ' '
    IF (M.GT.12) GOTO 2
    DO 1 I=1,N
    PRINT'(1X,12F10.4)',(MAT(I,J),J=1,M)
1   CONTINUE
    GOTO 10
2   CONTINUE
    IF (M.GT.24) THEN
    CALL PRNTXL(MAT,N,M)
    RETURN
    ENDIF
    DO 3 I=1,N
    PRINT'(1X,12F10.4)',(MAT(I,J),J=1,12)
3   CONTINUE
    PRINT'('//)'
    DO 4 I=1,N
    PRINT'(1X,12F10.4)',(MAT(I,J),J=13,M)
4   CONTINUE
10  PRINT'('///)'
    RETURN
    END
    SUBROUTINE PRNTXL(MAT,N,M)
    COMMON/MAINB/NCOL
    COMMON/MAINB/NDA
    REAL MAT(NDA,NDA)
    INTEGER I,J,K,L,M,N
    PRINT*, ' '
    DO 1 L=1,M,12
    K = L + 11
    IF (M-L.LT.11) K = M
    DO 2 I=1,N
    PRINT'(1X,12F10.5)',(MAT(I,J),J=L,K)
2   CONTINUE
    PRINT'('//)'
1   CONTINUE
    PRINT'('///)'
    RETURN
    END

```

```

SUBROUTINE TFR(X,A,N,M,K,I)
C
C      I - 1  GIVES  X = A
C           2  GIVES  X = A'
C           3  GIVES  X = A AS A VECTOR
C           4  GIVES  A = X   WHERE X WAS A VECTOR
C
      DIMENSION X(1),A(1)
      COMMON/MAINB/NCOL
      JS=(K-1)*NCOL*M
      JEND=M*NCOL
      GO TO (10,30,50,70),I
10    DO 20 II=1,N
      DO 20 JJ=II,JEND,NCOL
20    X(JJ)=A(JJ+JS)
      RETURN
30    DO 40 II=1,N
      KK=(II-1)*NCOL
      DO 40 JJ=1,M
      LL=(JJ-1)*NCOL+II
40    X(KK+JJ)=A(LL+JS)
      RETURN
50    KK=0
      DO 60 II=1,JEND,NCOL
      LL=II+N-1
      DO 60 JJ=II,LL
      KK=KK+1
60    X(KK)=A(JJ+JS)
      RETURN
70    KK=M*N+1
      DO 80 II=1,M
      LL=(M-II)*NCOL+1
      DO 80 IJ=1,N
      KK=KK-1
      JJ=LL+N-IJ
80    A(JJ+JS)=X(KK)
      RETURN
      END
SUBROUTINE TIME(EAT2,MM,DT,X1,X0,MODE,EAT,WORK,DEC)
COMMON/MAINA/NDA,NDA1
COMMON/SAVE/T(100),TS(100)
COMMON/NUM/IC1(21),IC2(21),IC3(21),IR(21),NC1,NC2,NC3,NR
REAL X0(NDA),EAT2(NDA,NDA),EAT(NDA,NDA),DT,MODE(2,12)
REAL WORK(NDA,NDA),A,AA,Z,X1(NDA)
INTEGER DEC,I,J,K,L,M,N,KK,LL,MM
PRINT*, ' TIMEIN '
DO 222 I=1,MM
DO 222 J=1,MM
222  WORK(I,J) = EAT2(I,J)
N = 1
Z = DT*5.
PRINT*, '      TIME              X              Y '
KK = 0

```

```

      A = 0
203  CONTINUE
      M = 0
201  CONTINUE
      PRINT*, ' 201 '
      IF (KK.GT.0) THEN
      CALL VMULFF(WORK, EAT2, MM, MM, MM, NDA, NDA, EAT, NDA, IER)
      PRINT*, ' 201 V '
      DO 112 I=1, MM
      DO 112 J=1, MM
      EAT2(I, J) = EAT(I, J)
112  CONTINUE
      PRINT*, ' 112 '
      ENDIF
      KK = 1
      CALL VMULFF(EAT2, X0, MM, MM, 1, NDA, NDA, X1, NDA, IER)
      PRINT*, ' 112 V '
      DO 113 I=1, MM
113  X0(I) = X1(I)
      M = M+1
      IF ((M*DT).EQ.Z) THEN
      A = A + Z
      DO 202 K=1, 2
      AA = 0.0
      DO 204 I=1, NC1
      J = IC1(I)
204  AA = AA + MODE(K, J) * X1(I)
      LL = NC1 * 4
      DO 205 I=1, NC2
      J = IC2(I)
205  AA = AA + MODE(K, J) * X1(I+LL)
      L = KK + NC2 * 4
      DO 206 I=1, NC3
      J = IC3(I)
206  AA = AA + MODE(K, J) * X1(I+L)
      IF (DEC.EQ.4) THEN
      M = L + NC3 * 4
      DO 211 I=1, NR
      J = IR(I)
211  AA = AA + MODE(K, J) * X1(I+M)
      ENDIF
      T(K) = AA
      N = N+1
202  CONTINUE
      WRITE(*, 9) A, T(1), T(2)
      IF (A.GE.20.0) GOTO 210
      GOTO 203
      ELSE
      GOTO 201
      ENDIF
210  CONTINUE
      PRINT*, ' 210 '
      PRINT*, '

```

```

9   FORMAT(3X,F4.1,4X,F15.6,4X,F15.6)
    END
    FUNCTION DOT(NR,A,B)
    DIMENSION A(1),B(1)
    DOT=0.
    DO 1 I=1,NR
1   DOT=DOT+A(I)*B(I)
    RETURN
    END
    SUBROUTINE VADD(N,C1,A,B)
    DIMENSION A(1),B(1)
    DO 1 I=1,N
1   A(I)=A(I)+C1*B(I)
    RETURN
    END
    FUNCTION XNORM1(N,A)
C   COMPUTES AN APPROXIMATION TO NORM OF A-- NOT A BOUND
    DIMENSION A(N)
    COMMON/MAINA/ NDA,NDA1
    NN=N*NDA
    C1=0.
    TR=A(1)
    IF (N.EQ.1) GO TO 20
    I=2
    DO 10 II=NDA1,NN,NDA
    J=II
    DO 5 JJ=I,II,NDA
    C1=C1+ABS(A(J)*A(JJ))
5   J=J+1
    TR=TR+A(J)
10  I=I+1
    TR=TR/FLOAT(N)
    DO 15 II=1,NN,NDA1
15  C1=C1+(A(II)-TR)**2
20  XNORM1=ABS(TR)+SQRT(C1)
    RETURN
    END
    FUNCTION XNORM(N,A)
C   COMPUTES AN APPROXIMATION TO NORM OF A -- NOT A BOUND
    DIMENSION A(1)
    COMMON/MAINB/NCOL,NCOL1
    NN=N*NCOL
    C1=0.
    TR=A(1)
    IF (N.EQ.1) GO TO 20
    I=2
    DO 10 II=NCOL1,NN,NCOL
    J=II
    DO 5 JJ=I,II,NCOL
    C1=C1+ABS(A(J)*A(JJ))
5   J=J+1
    TR=TR+A(J)
10  I=I+1

```

```

TR=IR/FLOAT(N)
DO 15 II=1,NN,NCOL1
15 C1=C1+(A(II)-TR)**2
20 XNORM=ABS(TR)+SQRT(C1)
RETURN
END
SUBROUTINE ABGC(A,B,C,N2,KCC,NACT,NSEN,ABG)
COMMON/MAINB/NCOL,NCOL1
REAL B(NCOL,1),A(NCOL,1),C(NCOL,1),KCC(NCOL,1),ABG(NCOL,1)
INTEGER N2,NACT,NSEN
CALL MMUL(B,KCC,N2,NACT,NACT,ABG)
CALL MMUL(ABG,C,N2,NACT,N2,KCC)
CALL UBOAT(A,KCC,N2,N2,ABG)
RETURN
END
SUBROUTINE MADD(A,B,N,M,C)
COMMON/MAINB/NCOL
REAL A(NCOL,NCOL),B(NCOL,NCOL),C(NCOL,NCOL)
INTEGER N,M
DO 1 I=1,N
DO 1 J=1,M
C(I,J)=A(I,J)+B(I,J)
1 CONTINUE
RETURN
END
SUBROUTINE UBOAT(A,B,N,M,C)
COMMON/MAINB/NCOL
REAL A(NCOL,NCOL),B(NCOL,NCOL),C(NCOL,NCOL)
INTEGER N,M
DO 1 I=1,N
DO 1 J=1,M
C(I,J)=A(I,J)-B(I,J)
1 CONTINUE
RETURN
END
SUBROUTINE FINDK(C,M,N,W,G,L,K)
REAL UT(21,21),VP(21,21),QPLUS(21,21)
REAL SI(24),CPLUS(21,21),QUT(21,21)
REAL V(21,21),C(21,21),G(21,21)
REAL K(21,21),W(48,48)
DO 1 I=1,M
DO 1 J=1,M
1 UT(I,J)=0.0
DO 2 I=1,M
2 UT(I,I)=1.0
DO 3 I=1,M
DO 3 J=1,N
3 VP(I,J)=C(I,J)
CALL LSVDF(VP,21,M,N,UT,21,M,SI,W,IER)
DO 4 I=1,N
DO 4 J=1,M
4 QPLUS(I,J)=0.0
DO 5 I=1,N

```

```

5  IF(SI(I).GT.0.00001) QPLUS(I,I)=1/SI(I)
   DO 6 I=1,N
   DO 6 J=1,N
6  V(I,J)=VP(I,J)
   CALL MMUL(QPLUS,UT,N,M,M,QUT)
   CALL MMUL(V,QUT,N,N,M,CPLUS)
   CALL MMUL(G,CPLUS,L,N,M,K)
   RETURN
   END

```

Appendix B

NASTRAN Analysis - 12 Modes

ϕ^T_D Matrix (12x21)

Mode 4	0.004775	0.004876	-.004429	0.000000
	0.004775	0.004876	-.004429	0.000000
	-.013837	0.003015	-.002568	-.013837
	0.003015	-.002568	0.009529	0.009529
	0.000000	-.013837	-.009082	-.013837
	-.009082			
Mode 5	0.000001	0.003619	0.003618	-.004642
	0.000001	-.003618	-.003619	0.013451
	-.000002	0.003619	0.003618	-.000002
	-.003618	-.003619	0.003620	-.003617
	0.013452	-.000002	0.003617	-.000002
	-.003620			
Mode 6	-.006023	0.000267	0.000266	-.008232
	0.006023	-.000266	-.000266	-.003888
	-.006023	0.000266	0.000266	0.006023
	-.000266	-.000266	0.000266	-.000266
	0.015687	-.006023	0.000266	0.006023
	-.000266			
Mode 7	0.000026	-.001570	-.001533	0.002816
	-.000026	0.001569	0.001532	-.006275
	0.000098	-.001565	-.001541	-.000096
	0.001563	0.001540	-.001585	0.001583
	-.006567	0.000089	-.001520	-.000088
	0.001519			
Mode 12	-.000452	-.000859	-.000903	0.010104
	0.000432	0.000869	0.000927	-.001977
	-.000531	-.000880	-.000909	0.000536
	0.000891	0.000928	-.000822	0.000824
	-.000105	-.000570	-.000940	0.000577
	0.000970			
Mode 13	-.016036	-.002018	0.006244	0.000073
	-.016045	-.002002	0.006278	-.000012
	0.000644	-.000401	0.004628	0.000668
	-.000382	0.004658	-.006268	-.006263
	0.000033	0.000658	0.010503	0.000686
	0.010546			
Mode 17	-.000811	-.000616	-.000255	0.000002
	-.000812	-.000619	-.000257	-.000000
	-.000085	-.000550	-.000331	-.000086
	-.000552	-.000333	-.000810	-.000812
	-.000000	-.000085	-.000073	-.000086
	-.000075			

$\Phi^T D$ Matrix (12x21) - (Continued)

Mode 21	-.001659	-.001165	0.000520	-.001659
	0.001649	0.001152	-.000527	0.001115
	0.001645	-.000847	0.000291	-.006164
	0.000840	-.000291	-.002043	0.002030
	-.004348	0.001618	0.001328	-.001622
	-.001337			
Mode 22	0.000981	-.002849	-.003470	0.001226
	-.000980	0.002859	0.003478	0.000638
	-.000292	-.002955	-.003384	0.000293
	0.002965	0.003389	-.002561	0.002570
	0.001357	-.000201	-.003782	0.000206
	0.003795			
Mode 24	-.019403	-.005697	0.005622	-.017699
	0.019497	0.005625	-.005680	0.001840
	0.002955	-.003584	0.004155	-.002891
	0.003542	-.004167	-.011645	0.011593
	-.008292	0.002726	0.011238	-.002757
	-.011342			
Mode 28	-.000000	-.000004	0.000001	-.000003
	-.000003	-.000005	0.000001	-.000000
	-.000008	0.000001	-.000004	-.000008
	-.000000	-.000003	0.000004	0.000003
	0.000000	-.000009	-.000009	-.000009
	-.000007			
Mode 30	-.000003	0.000010	0.000015	-.000005
	-.000009	0.000008	0.000016	0.000000
	-.000004	0.000018	0.000016	-.000004
	0.000016	0.000016	0.000023	0.000018
	0.000000	-.000005	0.000011	-.000004
	0.000014			

Appendix C

NASTRAN Analysis - 33 Modes

$\Phi^T D$ Matrix (33x21)

Mode 4	.004775	.004876	-.004430	.000000
	.004775	.004876	-.004430	-.000000
	-.013837	.003015	-.002568	-.013837
	.003015	-.002568	.009529	.009529
	-.000000	-.013837	-.009082	-.013837
	-.009082			
Mode 5	.000001	.003619	.003618	-.004642
	.000001	-.003618	-.003619	.013452
	-.000002	.003619	.003618	-.000002
	-.003618	-.003619	.003620	-.003617
	.013452	-.000002	.003617	-.000002
	-.003620			
Mode 6	-.006023	.000267	.000266	-.008232
	.006023	-.000266	-.000266	-.003888
	-.006023	.000266	.000266	.006023
	-.000266	-.000266	.000267	-.000266
	.015687	-.006023	.000266	.006023
	-.000267			
Mode 7	.000026	-.001570	-.001532	.002816
	-.000026	.001569	.001532	-.006275
	.000098	-.001564	-.001541	-.000096
	.001563	.001540	-.001585	.001583
	-.006567	.000090	-.001520	-.000088
	.001520			
Mode 8	-.000057	-.002771	-.002719	.000002
	-.000057	-.002771	-.002719	-.000000
	.000049	-.002762	-.002730	.000049
	-.002762	-.002730	-.002799	-.002799
	-.000000	.000049	-.002693	.000049
	-.002693			
Mode 9	.003516	.000906	-.000769	.000000
	.003515	.000906	-.000769	.000001
	.000165	.000572	-.000434	.000164
	.000572	-.000434	.001745	.001745
	-.000001	.000165	-.001607	.000163
	-.001607			
Mode 10	-.005664	.000499	.000306	-.008190
	.005664	-.000497	-.000305	-.003592
	-.006042	.000463	.000333	.006041
	-.000461	-.000332	.000599	-.000597
	.016056	-.006038	.000215	.006038
	-.000213			

$\Phi^T D$ Matrix (33x21) - (Continued)

Mode 11	.001926	.009437	.009099	-.000042
	.001924	.009438	.009099	.000005
	.001279	.009391	.009196	.001274
	.009392	.009196	.009638	.009639
	-.000007	.001284	.008962	.001279
	.008961			
Mode 12	-.000452	-.000859	-.000904	.010105
	.000432	.000869	.000927	-.001977
	-.000531	-.000880	-.000909	.000536
	.000891	.000929	-.000822	.000824
	-.000105	-.000570	-.000940	.000577
	.000970			
Mode 13	-.016036	-.002018	.006244	.000073
	-.016045	-.002002	.006278	-.000012
	.000644	-.000401	.004628	.000668
	-.000382	.004658	-.006268	-.006263
	.000033	.000658	.010503	.000686
	.010546			
Mode 14	.000554	-.001865	-.001122	.007250
	-.000518	.001872	.001116	.001287
	.002022	-.001729	-.001220	-.002008
	.001735	.001220	-.002262	.002277
	-.005328	.002010	-.000762	-.002002
	.000748			
Mode 15	-.006917	-.001207	.001357	-.000168
	-.006931	-.001238	.001291	.000031
	-.001849	-.000686	.000834	-.001932
	-.000724	.000773	-.002469	-.002481
	-.000105	-.001852	.002606	-.001935
	.002522			
Mode 16	-.000400	-.001771	-.003199	.014531
	.000377	.001782	.003200	-.002532
	-.003176	-.002052	-.003001	.003124
	.002057	.002995	-.001045	.001058
	.007844	-.003162	-.003875	.003123
	.003873			
Mode 17	-.000911	-.000616	-.000255	.000002
	-.000813	-.000619	-.000257	-.000000
	-.000085	-.000550	-.000331	-.000086
	-.000552	-.000333	-.000810	-.000812
	-.000000	-.000085	-.000073	-.000086
	-.000075			

$\Phi^T D$ Matrix (33x21) - (Continued)

Mode 18	.002003	.000107	-.000857	.000002
	.002006	.000106	-.000859	.000000
	.000053	-.000082	-.000671	.000054
	-.000082	-.000672	.000606	.000606
	.000001	.000050	-.001360	.000051
	-.001362			
Mode 19	.000000	.000000	-.000000	-.000001
	.000001	-.000000	-.000000	-.000000
	-.000000	-.000000	.000000	-.000000
	.000000	-.000000	-.000000	.000000
	-.000000			
Mode 20	-.000479	.000083	.000025	-.000018
	-.000475	.000088	.000027	.000001
	-.000675	.000089	.000022	-.000676
	.000094	.000025	.000162	.000170
	-.000003	-.000680	-.000057	-.000683
	-.000057			
Mode 21	-.001659	-.001165	.000520	-.001659
	.001650	.001152	-.000527	.001115
	.001645	-.000847	.000291	-.001636
	.000840	-.000291	-.002043	.002030
	-.004348	.001618	.001328	-.001621
	-.001337			
Mode 22	.000981	-.002849	-.003470	.001226
	-.000980	.002859	.003478	.000638
	-.000292	-.002955	-.003384	.000293
	.002965	.003389	-.002561	.002570
	.001357	-.000201	-.003782	.000206
	.003795			
Mode 23	-.000486	-.021293	-.020538	-.005602
	.000464	.021322	.020568	.003478
	.000061	-.021077	-.020851	-.000042
	.021124	.020881	-.021704	.021719
	.001631	.000545	-.020744	-.000513
	.020813			
Mode 24	-.019403	-.005697	.005622	-.017699
	.019407	.005625	-.005680	.001840
	.002955	-.003584	.004155	-.002891
	.003542	-.004167	-.011645	.011593
	-.008292	.002726	.011238	-.002757
	-.011342			

Φ^T D Matrix (33x21) - (Continued)

Mode 25	.000001	.000000	-.000000	.000001
	-.000001	-.000000	.000000	-.000000
	-.000000	.000000	-.000000	.000000
	-.000000	.000000	.000000	-.000000
	.000000	-.000000	-.000000	.000000
	.000000			
Mode 26	-.000001	-.000001	-.000001	-.000001
	-.000004	-.000002	-.000001	-.000000
	-.000012	.000001	-.000003	-.000011
	.000000	-.000003	.000006	.000003
	.000000	-.000012	-.000008	-.000012
	-.000007			
Mode 27	-.000000	.000000	-.000000	.000000
	.000000	.000000	-.000000	.000000
	.000000	-.000000	.000000	.000000
	.000000	.000000	-.000000	-.000000
	-.000000	.000000	.000000	.000000
	.000000			
Mode 28	-.000000	-.000004	.000001	-.000003
	-.000003	-.000005	.000001	-.000000
	-.000008	.000001	-.000004	-.000008
	-.000000	-.000003	.000004	.000003
	.000000	-.000009	-.000009	-.000009
	-.000007			
Mode 29	-.000043	-.000006	.000008	.000000
	-.000042	-.000006	.000008	-.000000
	-.000004	-.000008	.000006	-.000004
	-.000008	.000006	-.000027	-.000027
	-.000000	-.000003	.000024	-.000003
	.000024			
Mode 30	-.000003	.000010	.000015	-.000005
	-.000009	.000008	.000016	.000000
	-.000004	.000018	.000016	-.000004
	.000016	.000016	.000023	.000018
	.000000	-.000005	.000011	-.000004
	.000014			
Mode 31	-.000195	-.000069	.000030	-.000195
	.000197	.000069	-.000030	.000001
	.000002	-.000051	.000019	-.000001
	.000052	-.000019	-.000124	.000125
	-.000006	.000001	.000085	-.000001
	-.000086			

$\Phi^T D$ Matrix (33x21) - (Continued)

Mode 32	-.000070	.000105	.000112	-.000011
	.000071	-.000105	-.000111	.000001
	.000004	.000107	.000137	-.000002
	-.000107	-.000136	.000081	-.000080
	-.000020	.000005	.000169	-.000006
	-.000170			
Mode 33	-.001316	-.002646	.001407	-.000041
	-.001433	-.003269	.001633	.000746
	.001072	.008585	.006652	.002688
	.008136	.006664	.017513	.016781
	.000026	.000412	-.000583	.001442
	-.000294			
Mode 34	-.011539	-.001697	.003121	.002881
	.011322	-.000778	-.003301	.000771
	-.000215	-.011488	-.007493	.003215
	.010932	.006959	-.021277	.023013
	.001047	.001293	.010261	-.003702
	-.014854			
Mode 35	-.005960	.007119	.007164	.003841
	.000284	.009320	.007115	-.002497
	-.002435	-.000120	.002235	-.006592
	.004583	.001767	-.008845	.000859
	-.000228	.002261	.018397	-.000966
	.012288			
Mode 36	-.010398	-.003798	-.005374	.014434
	.010019	-.000317	.001925	-.000109
	-.000137	-.004307	.003640	.001805
	.003038	-.004431	-.010575	.011795
	-.001445	-.000139	.004044	.000032
	-.010171			

Bibliography

1. Aldridge, 2nd Lt Edward S. Decentralized Control of a Large Space Structure as Applied to the CSDL 2 Model. MS Thesis, GA/AA/82D-1. School of Engineering, Air Force Institute of Technology (AU), Wright-Patterson AFB OH, December 1982 (AD-A124702).
2. Calico, Robert A. Jr. and W. T. Miller. "Decentralized Control of a Flexible Spacecraft," AIAA/AAS Astrodynamics Conference. AIAA-82-1414. San Diego, CA August 9-11, 1982.
3. D'Azzo, John J. and Constantine H. Houpis. Linear Control System Analysis and Design: Conventional and Modern. (Second Edition). New York: McGraw-Hill Inc., 1981.
4. Henderson, Timothy. VCOSS Design Model and Disturbances. Revised Report for CSDL Publications C-5347, Charles Stark Draper Laboratory, Inc., Cambridge Massachusetts, November 12, 1981.
5. -----. Active Control of Space Structures (ACOSS) Model 2. Work performed by the Charles Stark Draper Laboratory, Inc. (CSDL) under Contract F30602-80-C-0096 as part of the ACOSS program for the Rome Air Development Center, September 1981.
6. Hess, Jeffrey W. Unpublished notes on Decentralized Control Using Direct Output Feedback. School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB OH, 1982.
7. Kleinman, D. L. A Description of Computer Programs for Use in Linear Systems Studies. The University of Connecticut School of Engineering, TR-77-2, Storrs, Connecticut, July 1977.
8. Laub, Alan J. "A Schur Method for Solving Algebraic Riccati Equations," IEEE Transactions on Automatic Control, Vol. AC-24, No. 6, p. 913.
9. Maybeck, Peter S. Stochastic Models, Estimation, and Control, Vols. 1-3. New York: Academic Press, 1979, 1982.
10. Meirovitch, L. et al. "A Comparison of Control Techniques for Large Flexible Systems," Journal of Guidance, Control and Dynamics, Vol. 6, No. 4, July-August 1983.
11. Meirovitch, L. and H. Oz. "Computational Aspects of the Control of Large Flexible Structures," Proceedings of 18th IEEE Conference on Decision and Control. December 12-14, 1979.
12. -----. "Modal-Space Control of Distributed Gyroscopic Systems," Journal of Guidance and Control, Vol. 3, No. 2, p. 140, 1978.

13. Miller, William T. "Decentralized Control of Large Space Structures." MS Thesis, School of Engineering, Air Force Institute of Technology, Wright-Patterson AFB OH, December 1981.
14. RADC-TR-83-56. Modifications to ACOSS Model #2 Design. A Study conducted by Charles Stark Draper Lab, Inc. for Rome Air Development Center, March 1983.
15. RADC-TR-82-21. ACOSS FIVE (Active Control of Space Structures) Phase IA. Study conducted by Lockheed Missiles and Space Company, Inc. for the Rome Air Development Center, March 1982 (AD-A116655).
16. RADC-TR-81-242. ACOSS EIGHT (Active Control of Space Structures) Phase II. Study conducted by TRW for Rome Air Development Center, September 1981 (AD-A108629).
17. RADC-TR-80-78. ACOSS FOUR (Active Control of Space Structures) Theory, Vol I (of two). Study conducted by Charles Stark Draper Lab, Inc. for Rome Air Development Center, April 1980 (AD-A085654).
18. RADC-TR-79-268. Actively Controlled Structures Theory. Theory of Design Methods, Vol I (of two). Study conducted by Charles Stark Draper Laboratory, Inc. for Rome Air Development Center, November 1979 (AD-B044535).
19. Salzwedel, Horst and J. H. Vincent. "Modeling, Identification, and Control of Flexible Aircraft," AIAA Atmospheric Flight Mechanics Conference. Gatlinburg, TN, August 15-17, 1983.
20. Sandell, Nils et al. "Survey of Decentralized Control Methods for Large Scale Systems," IEEE Transactions on Automatic Control, Vol. AC-23, No. 2, p. 108.
21. Sesak, J. R. et al. "Flexible Spacecraft Control by Model Error Sensitivity Suppression," The Journal of the Astronautical Sciences, Vol. 27, No. 2, pp. 131-156, April-June 1974.
22. Strang, Gilbert. Linear Algebra and Its Application. New York: Academic Press, 1976.

VITA

David Vernon Thyfault is a native of Colorado. He received his Bachelor of Science in Aerospace Engineering Sciences from the University of Colorado in May 1974 and was commissioned as a second lieutenant in the United States Air Force through the Officer Training School, Lackland AFB, Texas in September 1974. He served as a space vehicle engineer with the 6595th Space Test Group, Special Space Projects Branch, Vandenberg AFB, California. While at Vandenberg, he participated in design, test, and launch activities for the Defense Meteorological Satellite Program, Space Test Program vehicles P72-2, P76-5, P80-1, and P80-2, and the NASA meteorological series TIROS/NOAA program. His following assignment was as a space laser project officer for the Air Force Weapons Laboratory, Advanced Beam Control Branch, Kirtland AFB, New Mexico, where he evaluated control system concepts for high energy laser programs.

Permanent address: 1849 Mariposa Avenue
Boulder, Colorado 80302

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

REPORT DOCUMENTATION PAGE

1. REPORT SECURITY CLASSIFICATION UNCLASSIFIED			1b. RESTRICTIVE MARKINGS	
2a. SECURITY CLASSIFICATION AUTHORITY			3. DISTRIBUTION/AVAILABILITY OF REPORT Approved for public release; distribution unlimited.	
2b. DECLASSIFICATION/DOWNGRADING SCHEDULE				
4. PERFORMING ORGANIZATION REPORT NUMBER(S) AFIT/GA/AA/83-D-8			5. MONITORING ORGANIZATION REPORT NUMBER(S)	
6a. NAME OF PERFORMING ORGANIZATION School of Engineering		6b. OFFICE SYMBOL (If applicable) AFIT/EN	7a. NAME OF MONITORING ORGANIZATION	
6c. ADDRESS (City, State and ZIP Code) Air Force Institute of Technology Wright-Patterson AFB, Ohio 45433			7b. ADDRESS (City, State and ZIP Code)	
8a. NAME OF FUNDING/SPONSORING ORGANIZATION Air Force Flight Dynamics Laboratory		8b. OFFICE SYMBOL (If applicable) FIGCA	9. PROCUREMENT INSTRUMENT IDENTIFICATION NUMBER	
8c. ADDRESS (City, State and ZIP Code) Wright-Patterson AFB, Ohio 45433			10. SOURCE OF FUNDING NOS.	
11. TITLE (Include Security Classification) See Box 19			PROGRAM ELEMENT NO.	TASK NO.
			PROJECT NO.	WORK UNIT NO.
12. PERSONAL AUTHOR(S) David V. Thyfault, Capt., USAF				
13a. TYPE OF REPORT MS Thesis		13b. TIME COVERED FROM _____ TO _____		14. DATE OF REPORT (Yr., Mo., Day) 1983 December
				15. PAGE COUNT 137
16. SUPPLEMENTARY NOTATION <i>John Wolan</i> INTELLIGENCE INFORMATION SYSTEMS Development Air Force Institute of Technology (AFIT)				
17. COSATI CODES			18. SUBJECT TERMS (Continue on reverse if necessary and identify by block number)	
FIELD	GROUP	SUB. GR.	Closed Loop System, Decentralization, Equations of State, Control Theory, Actuators, Riccati Equation, Feedback, Decoupling (Interaction), Stability, Control Systems	
22	02			
19. ABSTRACT (Continue on reverse if necessary and identify by block number)				
Title: DECENTRALIZED CONTROL OF A LARGE SPACE STRUCTURE USING DIRECT OUTPUT FEEDBACK				
Thesis Chairman: Dr. Robert A. Calico, Jr.				
20. DISTRIBUTION/AVAILABILITY OF ABSTRACT UNCLASSIFIED/UNLIMITED ☒ SAME AS RPT. ☐ DTIC USERS ☐			21. ABSTRACT SECURITY CLASSIFICATION UNCLASSIFIED	
22a. NAME OF RESPONSIBLE INDIVIDUAL Robert A. Calico, Jr., Professor		22b. TELEPHONE NUMBER (Include Area Code) (513) 255-2104		22c. OFFICE SYMBOL AFIT/ENY

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

Direct output feedback control methods are used to develop a multiple-input multiple-output controller. The controller is applied to the ACOSS 2 Model (CSDL 2). NASTRAN is employed to generate modal approximations of the model. The first 36 modes are utilized and implemented in the controller. The control problem is formulated in state space form and direct output feedback is implemented. The state is represented as modal amplitudes and rates. System outputs are obtained by rate sensors and control is applied by point force actuators. Since the output matrices are not of full rank, their generalized inverses are obtained. These and steady full-state feedback optimal gains are used to determine suboptimal feedback gains. Control and observation spillover are eliminated using singular value decomposition. Decentralized control is accomplished using three and four controllers. Conditions for which the stability of the model is assured are developed. Sensitivity to uncontrolled residual modes was examined. Direct output feedback response was compared to that obtained using optimal full-state feedback. Full controller decoupling was achieved and stability maintained.

UNCLASSIFIED

SECURITY CLASSIFICATION OF THIS PAGE

END

FILMED

2-84

DTIC