

POSITION CONTROL AND VIBRATION SUPPRESSION OF A
FLEXIBLE BEAM-LIKE STRUCTURE CANTILEVERED FROM A ROTARY
ACTUATOR WITHIN A GRAVITATIONAL FIELD

by

Michael J. Hersh

Thesis submitted to the Faculty of
the Virginia Polytechnic Institute and State University
in partial fulfillment of the requirements of the degree of

MASTER OF SCIENCE

in

MECHANICAL ENGINEERING

APPROVED:

Harry H. Robertshaw, Chairman

Robert G. Leonard

Scott L. Hendricks

March, 1986

Blacksburg, Virginia

Position Control and Vibration Suppression of a
Flexible Beam-Like Structure Cantilevered From a Rotary
Actuator Within a Gravitational Field

by

Michael J. Hersh

ABSTRACT

An analytical and experimental investigation into the position and vibration control of beam-like structures within a gravity field using only root actuation was performed. Two methods were used to linearize the system's equations of motion. This, in turn, allowed for the use of powerful Modern Control Theory pole-location techniques to determine proper feedback-control gains. The control law was implemented on an IBM PC capable of analog/digital conversion. A DC servo motor served as the control actuator. Gains were computed for a continuous controller rather than for one having a sampled-data nature.

Both simulation and experimental response were good, and were seen to correspond well with computed system-eigenvalues. If gains were chosen to cause more negative eigenvalues, the system's response speed increased, as it should. If eigenvalues are forced to become too negative, however, the system became too fast for the controller, and response deteriorated.

ACKNOWLEDGMENTS

I wish to thank Drs. Harry H. Robertshaw, Robert G. Leonard, and Scott L. Hendricks for serving on my committee and providing useful advice as needed. I also thank NASA-Langley for funding a portion of this work. Additionally, I want to thank Mei-yu Zhou for helping me to interface the IBM PC with the system to be controlled.

I have had the pleasure to sit in the classrooms of a few distinguished professors while here at VPI&SU, and it is from these people that I have learned the most: L. D. Mitchell, S. L. Hendricks, L. Meirovitch, and J. N. Reddy.

My friends, of course, cannot go unmentioned. Their influence on this work may be indirect, but present nonetheless. I especially wish to recognize those who I've crushed on the racquetball court, who I've left looking at my backside during bike rides, who I've outwitted in conversation, and who know when I'm serious and when I'm not.

My family has been supportive through it all, and, needless to say, this is appreciated.

TABLE OF CONTENTS

	<u>Page</u>
ABSTRACT	
ACKNOWLEDGMENTS	iii
TABLE OF CONTENTS	iv
LIST OF FIGURES	vi
LIST OF TABLES	viii
NOMENCLATURE	x
1. INTRODUCTION	1
2. SYSTEM DYNAMICS	9
Kinetic Energy	
Potential Energy	
Generalized Forces	
Equations of Motion	
3. CONTROL	20
Linearization Scheme 1, Control Scheme 1	
Linearization Scheme 2, Control Scheme 2	
4. COMPUTER IMPLEMENTATION	32
Gain Determination	
Simulation of System Response	
Motor Commands	
5. ASPECTS OF THE EXPERIMENTAL SYSTEM	36
Instrumentation	
Model Modifications	
Damping	
Time-Derivatives of Feedback Signals	

TABLE OF CONTENTS (continued)

	<u>Page</u>
6. ANALYTICAL RESULTS	48
Gain Determination	
Control Scheme 1	
Control Scheme 2	
Simulation of System Response	
7. EXPERIMENTAL RESULTS	70
8. SUMMARY AND DISCUSSION	108
REFERENCES	112
APPENDICES	113
A. Gain-Determination Programs	
B. Simulation Program, Control Scheme 1	
C. IBM PC Program, Control Scheme 1	
D. Legendre Polynomials	
E. Experimental Apparatus	

LIST OF FIGURES

	<u>Page</u>
1.1 Schematic representation of the system	3
1.2 Beam shown in a variety of configurations	4
1.3 Major devices for experimental work	6
1.4 Flow diagram of work performed in this study	7
2.1 Beam model with coordinate system	10
2.2 Shape functions for Ritz approximation to lateral displacement	13
2.3 Infinitesimal element along beam length	14
3.1 Gains as a function of θ	28
3.2 Breakdown of lateral beam displacement for control scheme 2	30
5.1 Instrumentation	37
5.2 Estimator block diagram and response characteristics	42
6.1 Gains as a function of θ	57
6.2 Simulation example 1; θ initially displaced	65
6.3 Simulation example 2; q_2 initially displaced	66
6.4 Simulation example 3; θ and q_2 initially displaced	67
6.5 Simulation example 4; shoot for $R(t) = 1$. rad. with three different feedbacks on θ	68
6.6 Simulation example 5; shoot for $R(t) = 1$. rad, q_2 initially displaced	69
7.1 Estimator output as a function of frequency	82
7.2 Initial configurations for experimental tests	83
7.3 Uncontrolled-system response, initial configuration 1	84
7.4 Experimental results, experiment number 1	85
7.5 Simulation results, experiment number 1	86

LIST OF FIGURES (continued)

	<u>Page</u>
7.6 Experimental results, experiment number 2	87
7.7 Simulation results, experiment number 2	88
7.8 Experimental results, experiment number 3	89
7.9 Simulation results, experiment number 3	90
7.10 Experimental results, experiment number 4	91
7.11 Simulation results, experiment number 4	92
7.12 Experimental results, experiment number 5	93
7.13 Simulation results, experiment number 5	94
7.14 Experimental results, experiment number 6	95
7.15 Simulation results, experiment number 6	96
7.16 Experimental results, experiment number 7	97
7.17 Simulation results, experiment number 7	98
7.18 Uncontrolled-system response, initial configuration 2 . . .	99
7.19 Experimental results, experiment number 8	100
7.20 Simulation results, experiment number 8	101
7.21 Uncontrolled-system response, initial configuration 3 . . .	102
7.22 Experimental results, experiment number 9	103
7.23 Simulation results, experiment number 9	104
7.24 Slewing maneuver, $K_{\theta} = 1$, for comparison with results of experiment number 10	105
7.25 Experimental results, experiment number 10	106
7.26 Simulation results, experiment number 10	107
8.1 Characteristics of controller output voltage	110

LIST OF TABLES

		<u>Page</u>
6.1	Eigenvalue-placement, control scheme 2, example 1a	49
6.2	Eigenvalue-placement, control scheme 2, example 2a	50
6.3	Eigenvalue-placement, control scheme 2, example 3a	50
6.4	Eigenvalue-placement, control scheme 2, example 4a	51
6.5	Eigenvalue-placement, control scheme 2, example 5a	51
6.6	Eigenvalue-placement, control scheme 2, example 6a	52
6.7	Eigenvalue-placement, control scheme 2, example 7a	52
6.8	Eigenvalue-placement, control scheme 2, example 8a	53
6.9	Eigenvalue-placement, control scheme 2, example 9a	53
6.10	Eigenvalue-placement, control scheme 2, example 10a	54
6.11	Eigenvalue-placement, control scheme 2, example 11a	54
6.12	Eigenvalue-placement, control scheme 2, example 12a	55
6.13	Eigenvalue-placement, control scheme 2, example 13a	55
6.14	Eigenvalue-placement, control scheme 1, example 1b	59
6.15	Eigenvalue-placement, control scheme 1, example 2b	62
7.1	Control gains and corresponding system eigenvalues, experiment number 1	78
7.2	Control gains and corresponding system eigenvalues, experiment number 2	78
7.3	Control gains and corresponding system eigenvalues, experiment number 3	79
7.4	Control gains and corresponding system eigenvalues, experiment number 4	79
7.5	Control gains and corresponding system eigenvalues, experiment number 5	80
7.6	Control gains and corresponding system eigenvalues, experiment number 6	80

LIST OF TABLES (continued)

	<u>Page</u>
7.7 Control gains and corresponding system eigenvalues, experiment number 7	80
7.8 Control gains and corresponding system eigenvalues, experiment number 8	81
7.9 Control gains and corresponding system eigenvalues, experiment number 9	81
7.10 Control gains and corresponding system eigenvalues, experiment number 10	81

NOMENCLATURE*

A	uncontrolled-system matrix
A'	uncontrolled-system matrix, damping included
A*	uncontrolled-system matrix, damping and estimators included
B	vector relating to control hardware
C	θ damping coefficient
C_1	flexible beam motion damping coefficient
DA_1	
$D_{ijk\ell}$	
DD_{ij}	integration of Ritz shape functions (see Eq. 2.8)
D_i	
DS_1	integration of Ritz shape functions (see Eq. 2.5)
D_{ii}	
E	modulus of elasticity
$EOM(\theta)$	equation of motion relating to generalized coordinate θ
$EOM(q_a)$	equation of motion relating to generalized coordinate q_a
EST1	estimate of $\dot{\theta}$
EST2	estimate of \dot{ST}_1
EST3	estimate of \dot{ST}_2
EST4	estimate of \dot{ST}_3
EST	vector of estimates
F_{damp}	flexible beam motion generalized damping force
g	gravitational acceleration
g_0	estimator circuit feedforward constant

*Some variables represent two or more things. This is unfortunate; however, the correct usage should be clear from the context.

NOMENCLATURE (continued)

g_1	estimator circuit feedback constant
G	vector relating to control hardware
h_1	estimator dummy variable
i_a	motor armature current
I	moment of inertia of beam cross-section; identity matrix
K_m	motor constant
K_b	motor back EMF
\underline{K}	stiffness matrix
k_{ij}	elements of stiffness matrix
K	vector of control gains
k_{sys}	overall system gain
k_i	control gains
l	beam length
m	beam mass
M	mass matrix
m_{ij}	elements of mass matrix
p_i	generalized coordinates
P	vector of system states, estimators included
$q_a(t)$	generalized coordinate relating to beam deflection
$q_{a\theta}$	portion of $q_a(t)$ relating to static equilibrium configuration
q_{at}	portion of $q_a(t)$ relating to time-dependent motion
Q_i	generalized force
R_a	motor armature resistance
$R(t)$	control target command
s	length parameter along beam; Laplace domain variable

NOMENCLATURE (continued)

ST1	voltage of strain gage 1
ST2	voltage of strain gage 2
ST3	voltage of strain gage 3
T	kinetic energy; matrix transpose; instrumentation transformation matrix
Thv	voltage of potentiometer used for θ measurement
T_m	motor torque
u_i	measurements relating to system configuration
U	vector of u_i
V_a	motor armature voltage
V	potential energy
v	beam velocity
x	length parameter along beam-root tangent line
X	system states, no estimators
$y(x,t)$	lateral beam deflection

Greek

Γ	set of desired closed-loop eigenvalues
θ	beam-root angle
κ	beam curvature
λ_i	desired closed-loop eigenvalue
$\phi_i(x)$	Ritz approximation shape function; dummy variable in derivation of gain-determination algorithm

CHAPTER 1 INTRODUCTION

The existence of a space transportation system (the NASA shuttle) makes it possible to conceive of designing large satellites and spacecraft to be carried piecewise into space and constructed there for such purposes as communications, surveillance and space exploration. Large space structures (LSS) involve a high degree of mechanical flexibility. Some characteristics of LSS are:

1. they are distributed parameter systems.
2. they have many low natural frequencies which are often closely spaced.
3. they have very little natural damping.
4. requirements for shape, orientation, alignment, vibration suppression, and pointing accuracy are stringent.

It would therefore seem appropriate to actively control the LSS using a variety of sensors and actuators located about the structure and operating through on-line computer controllers [1]. The engineering community has become increasingly aware over the past decade of LSS and the control problems they pose, and this has resulted in an ever-growing number of LSS-related publications [1,2,3,4,5,6].

The work presented here was specifically undertaken to study situations in which flexible beam-like structures (appendages) are attached to rigid bodies. These appendages are presumed to be cantilevered from rotary actuators which are themselves mounted to the rigid bodies. These actuators supply the torque necessary to reposition the appendages, and also to actively suppress appendage vibrations. The primary task facing the control systems designer is to develop a scheme

for determining what the actuator-torques should be as a function of time and system configuration. Additionally, there will be a need to test this scheme before actually putting it in space. Such testing has long been recognized as being a problem -- primarily because of gravity. This problem is two-fold:

1. the gravitational force may deform these large, flexible structures so that they no longer function as intended, no longer function at all, or are permanently damaged.
2. the gravity-related torque which the rotary actuator must counteract is a nonlinear function of appendage position. This represents a nonlinear control problem, and is fundamentally different than the linear situation to be experienced in space.

Little can be done about the first problem; at least not by the control designer. The second problem, however, may be overcome. In this thesis two control schemes for Earth-testing will be presented, and in both cases they degenerate into a related although simpler scheme for actual space applications.

The flexible appendage used for this work is a simple steel beam. Its dimensions are 30" x 3" x 1/32"; it is assumed to be flexible only in a plane. A small DC servo motor is used as the rotary actuator. This beam/motor system, as it exists in the test laboratory, is shown schematically in Fig. 1.1. In Fig. 1.2, the system is shown in a variety of configurations. Strain gages affixed to the beam along its length are used to indirectly measure beam displacement. The beam and motor are obviously major components of the total system; however, they are not the only components. A block diagram of all major devices

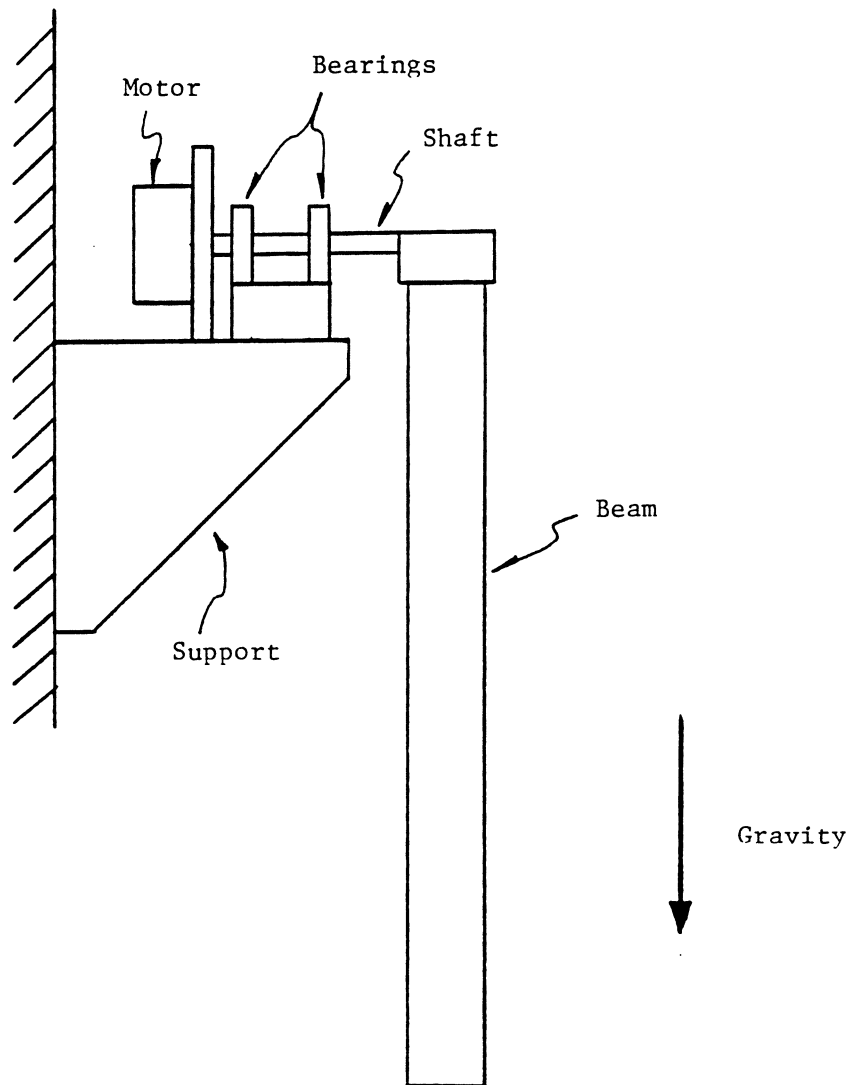


Figure 1.1 Schematic representation of the system

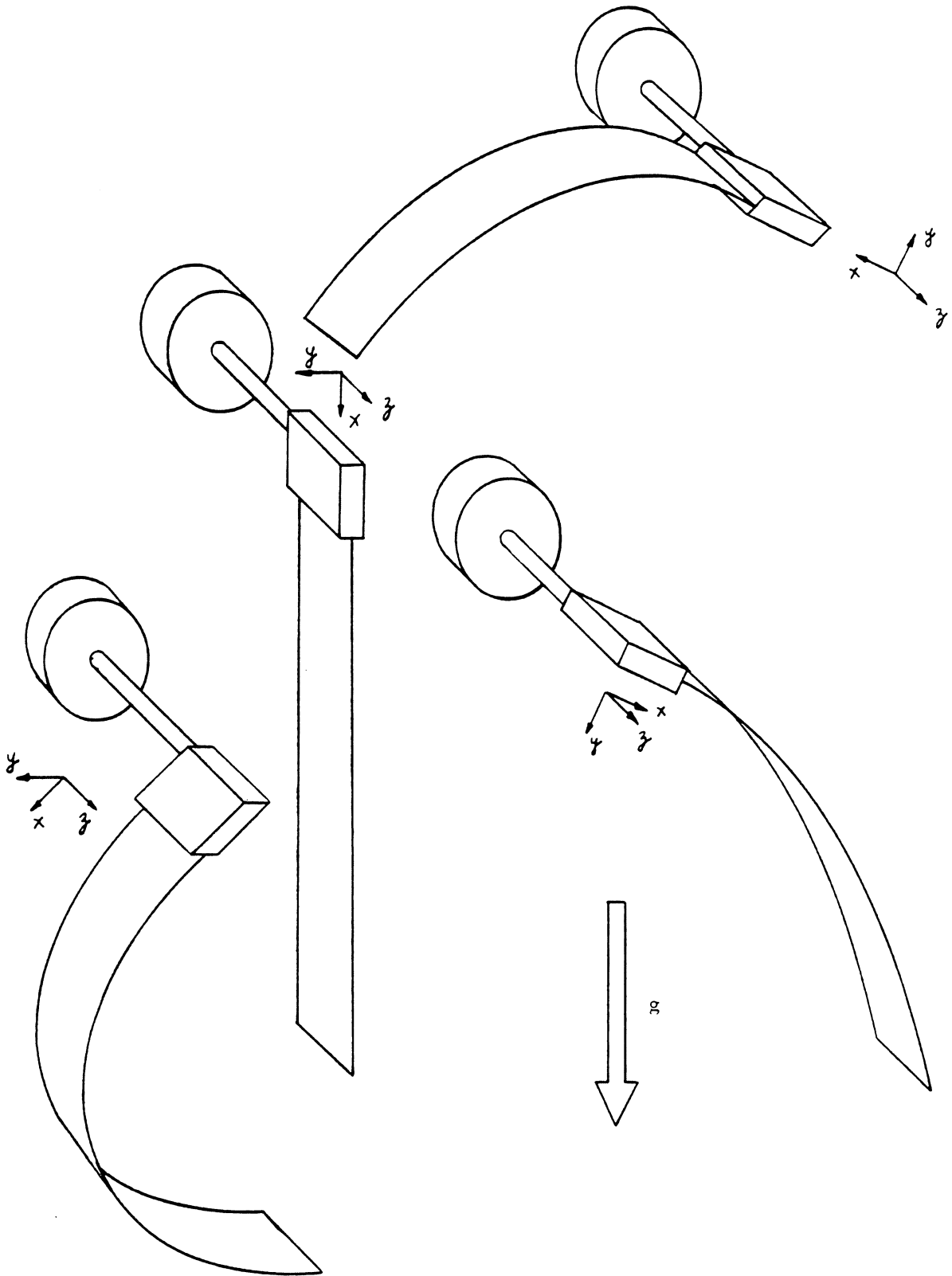


Figure 1.2 Beam shown in a variety of configurations

needed for experimental work is shown in Fig. 1.3. Such devices are typical, and system operating would be as follows.

Instrumentation distributed about the beam translates its motion into measurable quantities -- electrical voltages. These signals are amplified and sent to the control device -- in this case an IBM PC, which is capable of A/D conversion. The controller compares the incoming signals with a target command using an algorithm presented in this thesis. It then outputs a voltage directly proportional to the torque to be generated in the motor, but requires the assistance of a power amplifier to supply enough current to actually drive the motor. The generated torque is then applied to the beam. This control process of sensing system configuration and generating appropriate motor torques repeats continually. It is through these continually-changing torques that we force the system to behave as desired.

Figure 1.4 shows a flow diagram of steps followed during this study to produce good experimental beam response. To understand the system from a dynamic point of view, it is necessary to derive the appropriate equations of motion (EOM). This is done in Chapter 2. A Lagrange energy-balance approach is used. The EOM are found to be nonlinear with respect to both beam curvature and the gravity-induced torque about the beam's pivot axis (see Fig. 1.2). The torque nonlinearity is the more significant of the two. Methods for determining feedback control gains of linear systems are better developed than those for nonlinear systems. Therefore, in Chapter 3, the EOM are first linearized, and then gains are chosen to force the system to adopt prescribed eigenvalues. In Chapter 4, computer programs relating to

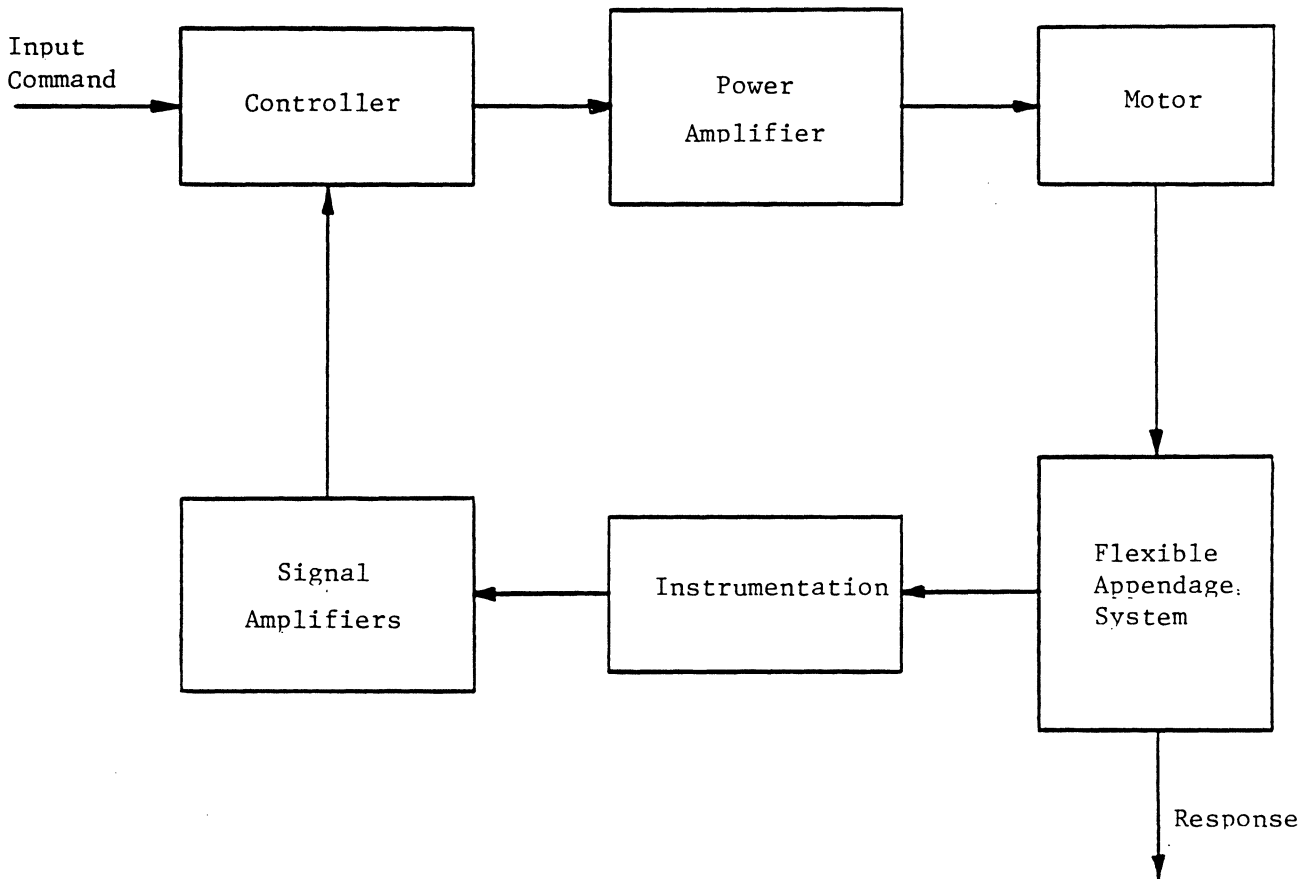


Figure 1.3 Major devices for experimental work

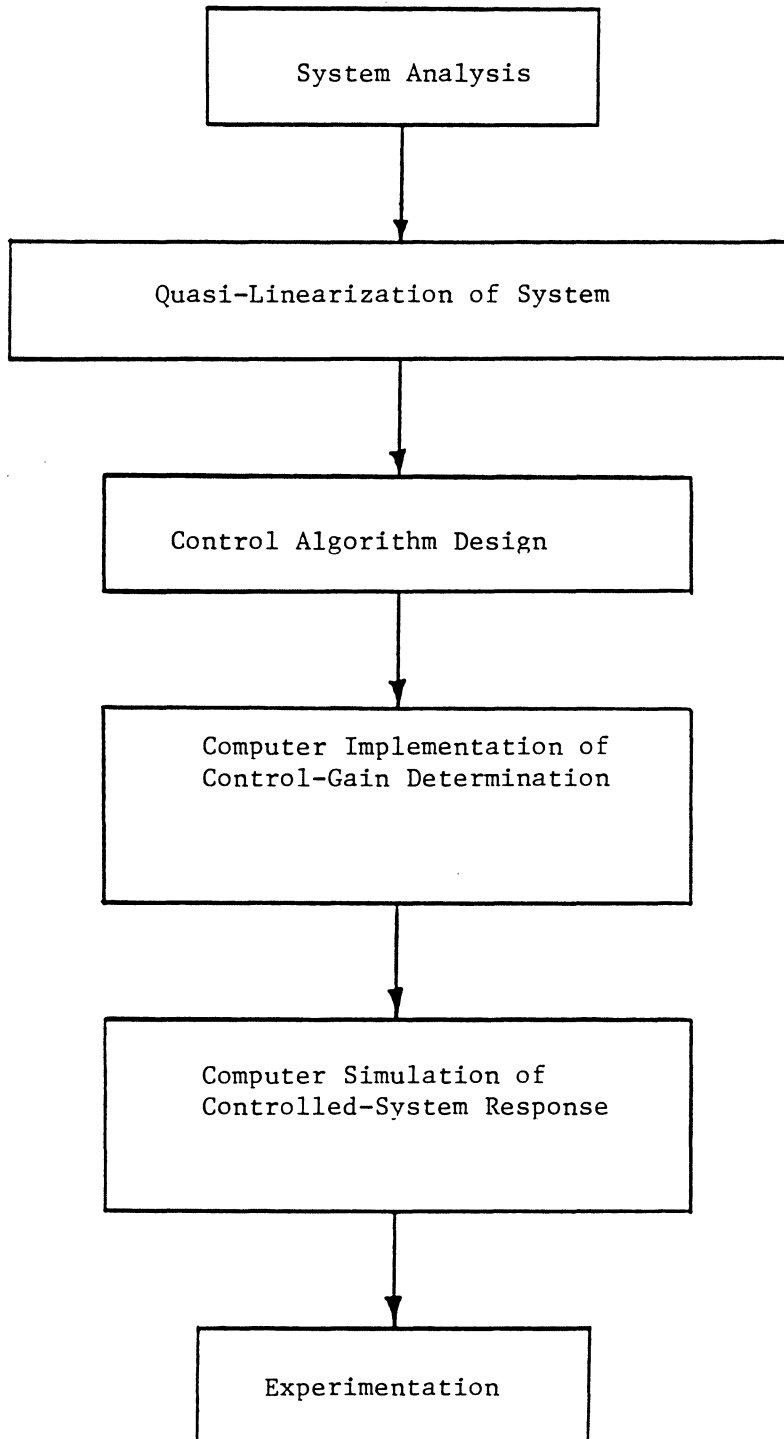


Figure 1.4 Flow diagram of work performed in this study

1. gain-determination
2. simulation of system response
3. experimental control

are discussed. Some aspects of the experimental system are presented in Chapter 5. Instrumentation is discussed here, as is the method for determining system damping constants. Additionally, filters used in conjunction with signal-differentiators are discussed. Analytical results are presented in Chapter 6. Gains are found and the corresponding system eigenvalues are reported. Also, typical simulation plots are shown. Experimental response is presented in Chapter 7. In conjunction with these plots, the following are given:

1. control gains and eigenvalues
2. free response of system subject to identical initial conditions
3. simulation response using the same gains

A brief summary of the work is provided in Chapter 8, along with conclusions and recommendations.

CHAPTER 2 SYSTEM DYNAMICS

The method used for control-algorithm design as detailed in Chapter 3 requires that the equations of motion for the system model be known. These equations will be determined in this chapter.

The system has been discussed. Essentially, it consists of a flexible beam cantilevered to the output shaft of a DC motor, as shown in Fig. 1.1. Referring to Fig. 2.1, which shows the beam in an arbitrary configuration, it is seen that beam motion can be completely described with the parameters $\theta(t)$ and $y(s,t)$. The equations of motion are derived using Lagrange's energy-balance approach. The Lagrange equations have the form:

$$\frac{\partial}{\partial t} \left(\frac{\partial T}{\partial \dot{p}_i} \right) - \frac{\partial T}{\partial p_i} + \frac{\partial V}{\partial p_i} = Q_i \quad i = 1, \dots, n \quad (2.1)$$

where

- T \equiv kinetic energy
- V \equiv potential energy
- Q_i \equiv non-conservative, generalized force
- p_i \equiv generalized coordinate
- t \equiv time

One equation of motion (EOM) is needed for each generalized coordinate. These generalized coordinates correspond to degrees of freedom (DOF) of the system. Because a continuous structure has an infinite number of DOF, it would seem that an infinite number of equations of motion are required. To deal with such a large system in

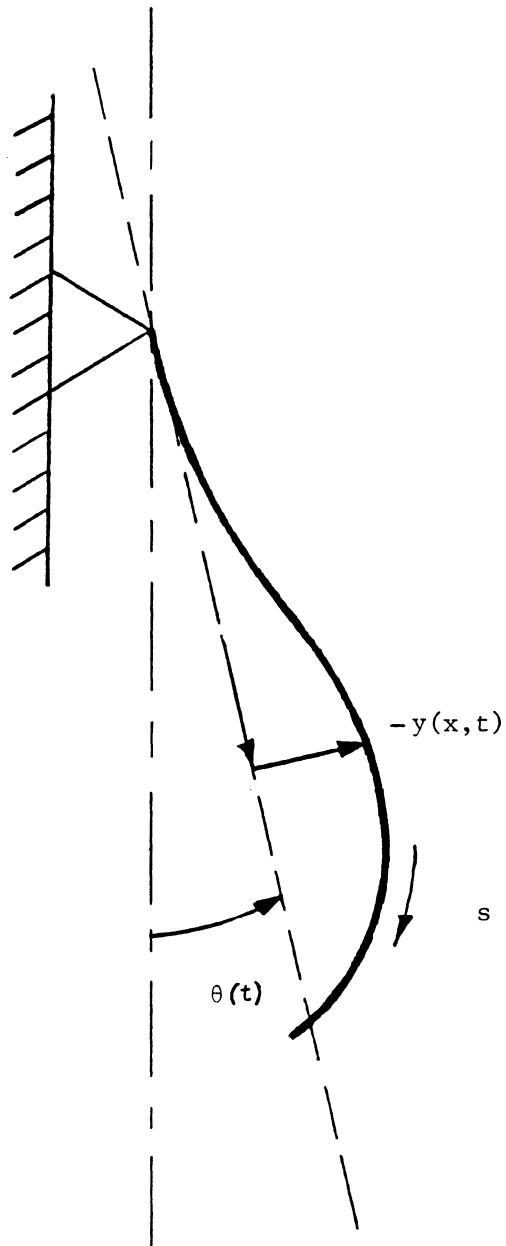


Figure 2.1 Beam model with coordinate system

an exact manner is beyond our means. We therefore seek to approximate the system, thereby reducing the number of required equations. There are two basic methods for doing this. One entails an approximation of the physical system itself; the beam could be visualized as being composed of n discrete masses connected by n lateral springs. The number of EOM would then be $n + 1$, with the 1 corresponding to the coordinate $\theta(t)$. As an alternative to this, beam shape could be approximated using a Ritz series:

$$y(s,t) = \sum_{i=1}^n q_i(t) \phi_i(s) \quad (2.2)$$

where $\phi_i(s)$ are basis or shape functions, and $q_i(t)$ are time-dependent multiplying factors. Again, the number of EOM would be $n + 1$. It is thought that a better approximation to beam shape for a given number n can be achieved using the Ritz approach. The key to successfully using a Ritz approach is in the choice of shape functions. For the type of motion which our beam is likely to experience, frequency of vibration will probably be in the vicinity of the beam's lower natural modes. A linear combination of corresponding mode shapes should then be adequate to describe all possible beam motion.

Lateral beam displacement is therefore approximated by the Ritz method, and the first three natural mode shapes for a clamped-free beam are used as basis functions. Four generalized coordinates are therefore needed, these being: θ, q_1, q_2, q_3 . The units of θ are radians, and the units of q_i are in./in. (cm/cm). The basis functions (mode shapes) are given by the equations:

$$\phi_1(s) = \alpha \left[\frac{\cos(\beta_1 \ell) + \cosh(\beta_1 \ell)}{\sinh(\beta_1 \ell) - \sin(\beta_1 \ell)} (\cos\beta_1 s - \cosh\beta_1 s) + \sinh(\beta_1 s) - \sin\beta_1 s \right] \quad (2.3)$$

where $\beta_1 \ell = 1.875$, $\beta_2 \ell = 4.694$, $\beta_3 \ell = 7.855$, $\alpha = 1.0$ in. (cm)

These basis functions are shown in Fig. 2.2.

Kinetic Energy

The contribution to kinetic energy of one infinitesimal element (Fig. 2.3) is

$$dT = \frac{1}{2} v^2 dm = \frac{m}{2\ell} v^2 ds = \frac{m}{2\ell} (x\dot{\theta} - \dot{y})^2 ds$$

Integrating over the beam length yields the total kinetic energy:

$$T = \frac{m}{2\ell} \int_0^{\ell} (x\dot{\theta} - \dot{y})^2 ds \quad (2.4)$$

For a beam undergoing small lateral deflections, it is generally assumed that $x \approx s$. Our beam is flexible, and this assumption does not seem appropriate. However, the magnitude of the quantity $x\dot{\theta}$ is a much stronger function of $\dot{\theta}$ than it is of the difference between x and s . To simplify Eq. (2.4), then, it is indeed assumed that $x \approx s$. Substituting this and the Ritz approximation for y into Eq. (2.4) yields

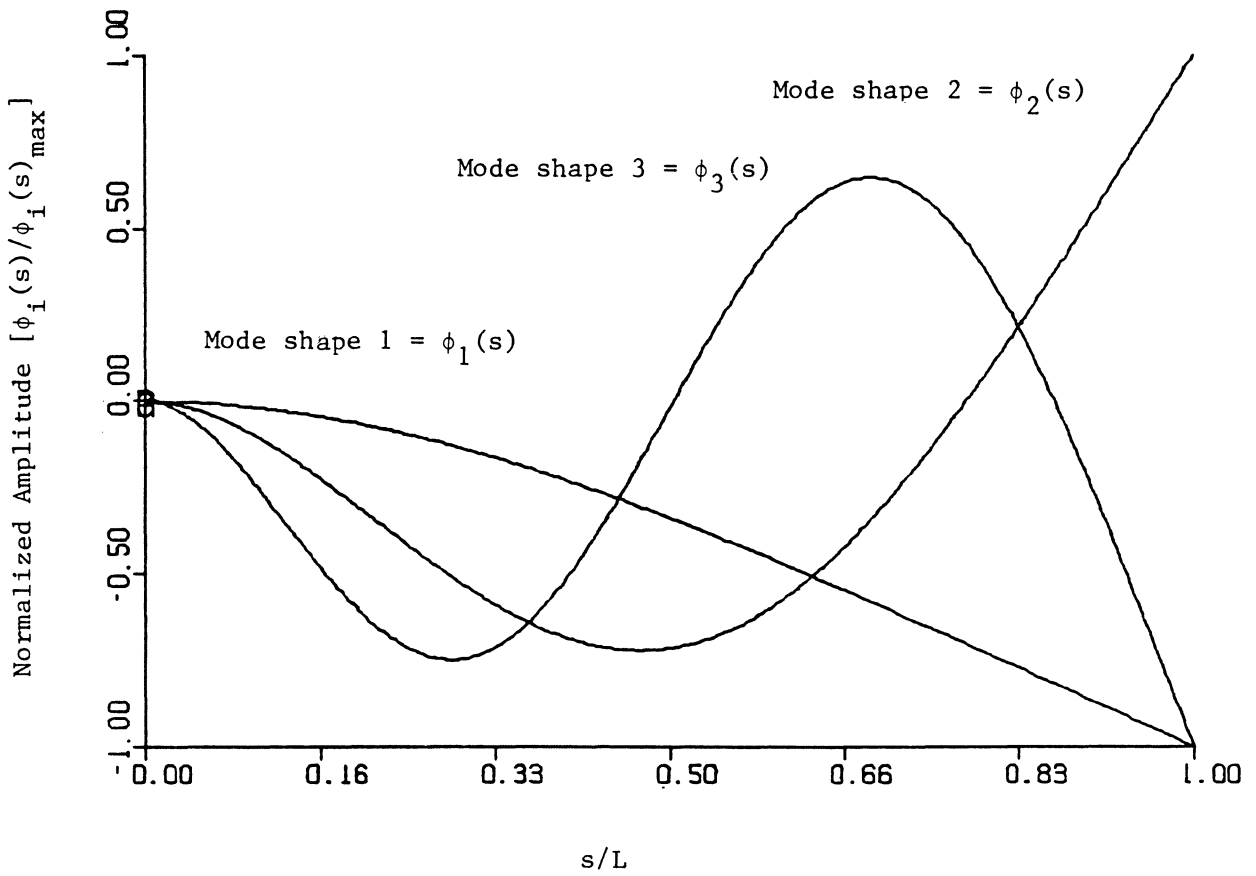


Figure 2.2 Shape functions for Ritz approximation to lateral displacement (refer to Eqs. 3.2)

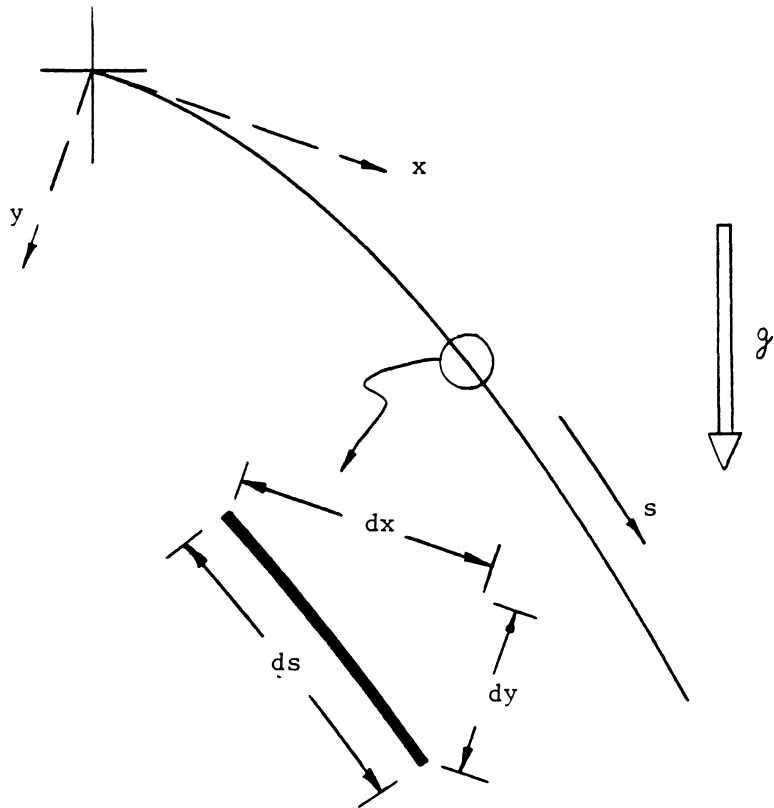


Figure 2.3 Infinitesimal beam element

$$\begin{aligned}
T &= \frac{m}{2\ell} \int_0^\ell [s\dot{\theta} - \sum_i \dot{q}_i \phi_i]^2 ds \\
&= \frac{m}{2\ell} \int_0^\ell [s^2 \dot{\theta}^2 - 2s\dot{\theta} \sum_i \dot{q}_i \phi_i + \sum_{ij} \dot{q}_i \dot{q}_j \phi_i \phi_j] ds \\
&= \frac{m}{2\ell} \left[\frac{\ell^3}{3} \dot{\theta}^2 - 2\dot{\theta} \sum_i \dot{q}_i \int_0^\ell s \phi_i ds + \sum_{ij} \dot{q}_i \dot{q}_j \int_0^\ell \phi_i \phi_j ds \right]
\end{aligned}$$

Because of mode-shape orthogonality, $\int_0^\ell \phi_i \phi_j ds = 0$ for $i \neq j$. Therefore

$$T = \frac{m\ell^2}{6} \dot{\theta}^2 - \frac{m}{\ell} \dot{\theta} \sum_i \dot{q}_i DS_i + \frac{m}{2\ell} \sum_i \dot{q}_i^2 D_{ii} \quad (2.5)$$

where

$$DS_i = \int_0^\ell s \phi_i ds$$

$$D_{ii} = \int_0^\ell \phi_i^2 ds$$

Potential Energy

The two sources of potential energy for the beam are elasticity and gravity. Potential energy can be described mathematically as

$$V = \frac{EI}{2} \int_0^\ell \kappa^2 ds - \frac{mg}{\ell} \int_0^\ell (x \cos\theta + y \sin\theta) ds \quad (2.6)$$

where $\kappa \equiv$ beam curvature

The first term accounts for elastic strain energy and the second for the gravitational potential. The usual approximation to curvature for stiff beams undergoing small lateral deflection is $\kappa \approx y''$. Our beam is

capable of large deflection and this therefore seems inappropriate. The exact expression for curvature, which can be found in most Analytical Geometry texts, is

$$\kappa = \frac{y''}{\sqrt{1 - y'^2}} \quad .$$

where primes denote differentiation with respect to s .

To use this exact expression in Eq. (2.6) would prove difficult. An approximation is therefore desired. Expanding it into a polynomial series yields

$$\kappa = y'' [1 + \frac{1}{2} y'^2 + \dots]^{1/2}$$

Squaring this as would be necessary for Eq. (2.6),

$$\kappa^2 = y''^2 [1 + y'^2 + \frac{1}{4} y'^4 + \dots]$$

Only a finite number of terms can be retained. More terms mean more accuracy, but also more complexity. The best balance between these two opposing criteria is thought to be achieved with two terms. Therefore:

$$\kappa^2 = y''^2 [1 + y'^2] \tag{2.7}$$

With respect to the second term of Eq. (2.6), the gravitational term, it was at first intended to use the approximation $x \approx s$, as was done for the kinetic energy expression. However, this assumption

implies that no work is done by the component of the gravitational force in the x-direction (Fig. 2.3). This, in turn, implies that static deflection beam shapes are symmetrical about $\theta = 90^\circ$, which is known intuitively to be incorrect. A more accurate description of x is therefore needed. The following equations follow directly from Fig. 2.3.

$$(dx)^2 + (dy)^2 = (ds)^2$$

$$\left(\frac{dx}{ds}\right)^2 + \left(\frac{dy}{ds}\right)^2 = 1$$

$$x'^2 = 1 - y'^2$$

$$x' = [1 - y'^2]^{1/2}$$

Expanding this last expression into a polynomial series, truncating to two terms and then integrating produces the needed relationship:

$$x = \int_0^s \left(1 - \frac{1}{2} y'^2\right) ds$$

Using this and the curvature approximation of Eq. (2.7) gives

$$v = \frac{EI}{2} \int_0^{\ell} (y''^2 + y''^2 y'^2) ds - \frac{mg}{\ell} \int_0^{\ell} \left[\int_0^s \left(1 - \frac{1}{2} y'^2\right) ds \cos\theta + y \sin\theta \right] ds$$

With the Ritz approximation included, this becomes

$$\begin{aligned}
V = & \frac{EI}{2} \left[\sum_i q_i^2 DA_i + \sum_{ijkl} q_i q_j q_k q_l D_{ijkl} \right] \\
& - \frac{mg\ell}{2} \cos\theta + mg \frac{\cos\theta}{2\ell} \sum_{ij} q_i q_j DD_{ij} - \frac{mg}{\ell} \sin\theta \sum_i q_i D_i
\end{aligned} \tag{2.8}$$

where

$$\begin{aligned}
DA_i &= \int_0^\ell \phi_i''^2 ds \\
D_{ijkl} &= \int_0^\ell \phi_i'' \phi_j'' \phi_k' \phi_l' ds \\
DD_{ij} &= \int_0^\ell \int_0^{s_x} \phi_i' \phi_j' ds ds \\
D_i &= \int_0^\ell \phi_i ds
\end{aligned}$$

These integrals are evaluated numerically.

Generalized Forces

The only force purposely applied to the beam is torque from the motor. Other forces do exist. Damping exists. Damping forces are difficult to predict analytically, and so discussion of these is postponed until testing is discussed in Chapter 5. Disturbance forces may also exist (unintended impact with a moving object, perhaps). However, they have no influence on the characteristics of transient system response (system eigenvalues), and are therefore of no present interest.

Motor torque is directly related to motor current:

$$T_m = K_m i_a$$

The motor gets its power from an amplifier. If using a voltage

amplifier, it is necessary to express motor current as a function of motor voltage: $i_a \approx (V_a - K_b \dot{\theta})/R_a$. Ours is an amplifier which generates a current proportional to its input voltage. The current from the amplifier equals the current through the motor. The input/output equation for our amplifier is:

$$i_a = K_a V_a$$

where V_a is the voltage signal from the controller. Motor torque can now be expressed as

$$T_m = K_a K_m V_a \quad (2.9)$$

Equations of Motion

The four equations of motion are obtained by substituting the energy and force relationships into the Lagrange equations. The results are

$$\begin{aligned} \text{EOM,}(\theta): \quad & \frac{m\ell^2}{3} \ddot{\theta} - \frac{m}{\ell} \sum_i \ddot{q}_i D S_i + \frac{mg\ell}{2} \sin\theta - \frac{mg}{\ell} \cos\theta \sum_i q_i D_i \\ & - \frac{mg}{2\ell} \sin\theta \sum_{ij} q_i q_j D D_{ij} = K_a K_m V_a \end{aligned} \quad (2.10)$$

$$\begin{aligned} \text{EOM,}(q_a): \quad & -\frac{m}{\ell} \ddot{\theta} D S_a + \frac{m}{\ell} \ddot{q}_a D_{aa} + EI[q_a D_a^2 + \sum_{jkl} q_j q_k q_l D_{ajkl} \\ & + \sum_{ijk} q_i q_j q_k D_{ijka}] + \frac{mg}{\ell} \cos\theta \sum_i q_i D D_{ai} - \frac{mg}{\ell} \sin\theta D_a = 0 \end{aligned}$$

$$(a = 1, 2, 3)$$

CHAPTER 3 CONTROL

A feedback control scheme is used, as indicated in Fig. 1.2. The configuration of the system (Fig. 2.1) is continuously monitored, and this information is used to calculate control signals. The equation used for this calculation has the form

$$V_a = k_{sys} [R(t) - \sum_i k_i u_i(t)] \quad (3.1)$$

where $R(t)$ relates to the desired target, $u_i(t)$ are measurements relating to system configuration, k_{sys} is the overall system gain (treated as a constant = 0.21), and k_i are the feedback gains to be determined. Various methods are available for finding gains which lead to desirable system response. The most highly-developed methods, of course, correspond to linear systems. Our system is nonlinear; however, it can be linearized. The schemes used for linearization are an important part of this thesis, and will be discussed shortly. For the time being, it should be assumed that the system has been linearized.

The linear equations of motion without damping can be rewritten in vector-matrix notation as

$$\begin{bmatrix} m_{11} & m_{12} & m_{13} & m_{14} \\ m_{21} & m_{22} & m_{23} & m_{24} \\ m_{31} & m_{32} & m_{33} & m_{34} \\ m_{41} & m_{42} & m_{43} & m_{44} \end{bmatrix} \begin{bmatrix} \ddot{\theta} \\ \ddot{q}_1 \\ \ddot{q}_2 \\ \ddot{q}_3 \end{bmatrix} + \begin{bmatrix} k_{11} & k_{12} & k_{13} & k_{14} \\ k_{21} & k_{22} & k_{23} & k_{24} \\ k_{31} & k_{32} & k_{33} & k_{34} \\ k_{41} & k_{42} & k_{43} & k_{44} \end{bmatrix} \begin{bmatrix} \theta \\ q_1 \\ q_2 \\ q_3 \end{bmatrix} = \begin{bmatrix} K_a K_m k_{sys} \\ 0 \\ 0 \\ 0 \end{bmatrix} V_a^*$$

* k_{sys} has been transferred from V_a to its coefficient vector for convenience of notation.

or

$$M\ddot{X} + \underline{K}X = GV_a \quad (3.2)$$

Matrix mathematics (linear algebra) is a highly-developed field, and therefore very powerful. Modern control theory exploits this. To increase the utility of modern control theory even further, a state-space representation of the system is generally used. This unifies the notation for all linear systems, thereby unifying the solution techniques.

The state variables of a system consist of a minimum set of parameters which completely summarize the system's status: if at any time t_0 the values of the state variables are known, then the system response at time $t_1 > t_0$ can be uniquely determined if all system inputs during the interval t_0, t_1 are known. A reasonable choice for the state variables of our system are the generalized coordinates decided upon in the previous chapter and their time-derivatives: $\theta, q_1, q_2, q_3, \dot{\theta}, \dot{q}_1, \dot{q}_2, \dot{q}_3$. Equation (3.2) can then be rewritten as

$$\begin{bmatrix} \dot{X} \\ \ddot{X} \\ X \end{bmatrix} = \begin{bmatrix} 0 & I \\ -M^{-1}K & 0 \end{bmatrix} \begin{bmatrix} X \\ \dot{X} \end{bmatrix} + \begin{bmatrix} 0 \\ M^{-1}G \end{bmatrix} v_a$$

where $X = [\theta, q_1, q_2, q_3]^T$

or

$$\dot{P} = AP + BV_a \quad (3.3)$$

where $P = [X \ \dot{X}]^T$

Specifying the control input to be as shown in Eq. (3.1) changes this to

$$\dot{P} = AP + B[R(t) - KU]$$

Assuming that U is linearly related to the system states

$$U = TP \tag{3.4}$$

this becomes

$$\dot{P} = [A - BKT]P + BR(t) \tag{3.5}$$

The eigenvalues of the state-variable coefficient-matrix indicate what the characteristics of transient system response are: stable or unstable, fast or slow, underdamped or overdamped. The addition of feedback-control changes this matrix from [A] to [A - BKT] and therefore changes the system response characteristics. For stable response all eigenvalues must have negative real parts. The rate of response increases as these parts become more negative.

Brogan [8] outlines a method for calculating gains which force the system to adopt prespecified eigenvalues. The method is used extensively in this thesis, and will therefore be presented.

The eigenvalues of the state-variable coefficient-matrix are roots of

$$\Delta'(\lambda) \triangleq |\lambda I_n - A + BKT| = 0 \quad (3.6)$$

where n is the system order. In order to synthesize the system with real hardware, all elements of the gain vector K must be real. This will be the case if, for each complex eigenvalue $\lambda_i \in \Gamma$, $\bar{\lambda}_i$ is also assigned to Γ (where $\Gamma \equiv$ set of desired closed-loop eigenvalues). Equation (3.6) is rewritten as

$$\begin{aligned} \Delta'(\lambda) &= |(\lambda I_n - A) [I_n + (\lambda I_n - A)^{-1} BKT]| \\ &= |\lambda I_n - A| \cdot |I_n + (\lambda I_n - A)^{-1} BKT| \end{aligned} \quad (3.7)$$

Defining $\Delta(\lambda) = |\lambda I_n - A| \equiv$ open loop eigenvalues, and $\phi(\lambda) = (\lambda I_n - A)^{-1}$, Eq. (3.7) becomes

$$\Delta'(\lambda) = \Delta(\lambda) |I_r + \phi(\lambda) BKT| = \Delta(\lambda) |I_r + KT \phi(\lambda) B| \quad (3.8)$$

where r is the number of feedback signals ($r < n$). The second form of Eq. (3.8) is obtained by using an easily proved determinant identity provided by Brogan. The vector K must be selected so that $\Delta'(\lambda_i) = 0$ for each $\lambda_i \in \Gamma$. This will be accomplished by forcing the $r \times r$ determinant to vanish. If any desired λ_i is also a root of $\Delta(\lambda)$, the following procedure is still valid. A sufficient condition for the

determinant of $(I_r + K T \phi(\lambda) B)$ to be zero is if any row or column is zero. More general conditions under which a determinant will be zero are well known, such as the existence of any linear dependencies among the rows or columns. Columns will be forced to zero here because of the simplicity that results. Define the j th column of I_r as e_j and define $\Psi(\lambda_j) = \phi(\lambda_j) B$, with the j th column being ψ_j . Then λ_j is a root of $\Delta'(\lambda)$ if K is selected to satisfy $e_j + K\psi_j(\lambda_j) = 0$, since this forces column j to be zero. Thus

$$K\psi_j(\lambda_j) = -e_j$$

This equation by itself is not sufficient for determining K . However, if an independent equation of this type can be found for every $\lambda_j \in \Gamma$, then K can be determined. If all the desired λ_j are distinct, it will always be possible to find n linearly independent columns $\psi_{j1}(\lambda_1)$, $\psi_{j2}(\lambda_2)$, ..., $\psi_{jn}(\lambda_n)$ from the columns of the $n \times nr$ matrix $[\Psi(\lambda_1), \Psi(\lambda_2), \dots, \Psi(\lambda_n)]$. Thus

$$K = -[e_{j1}, e_{j2}, \dots, e_{jn}] [\psi_{j1}(\lambda_1), \psi_{j2}(\lambda_2), \dots, \psi_{jn}(\lambda_n)]^{-1} \quad (3.9)$$

Example

Let the system of Eq. (3.3) have $A = \begin{bmatrix} 0 & 2 \\ 0 & 3 \end{bmatrix}$, $B = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$

Then

$$\phi(\lambda) = \frac{\begin{vmatrix} \lambda-3 & 2 \\ 0 & \lambda \end{vmatrix}}{\lambda(\lambda-3)}, \quad \psi(n) = \psi_1(\lambda) = \frac{2}{\lambda(\lambda-3)}$$

If the desired poles are $\lambda_1 = -3$, $\lambda_2 = -4$ then $\psi_1(\lambda_1) = [1/9 - 1/6]^T$ and $\psi_1(\lambda_2) = [1/14 - 1/7]^T$ are linearly independent. Equation (3.9) gives $K = -[1 \quad 1] \begin{bmatrix} 36 & 18 \\ -42 & -28 \end{bmatrix} = [6 \quad 10]$. This feedback gain matrix gives closed-loop eigenvalues at $\lambda = -3$ and -4 .

Linearization Schemes

In order to use the attractive pole-assignment method presented in the previous section to determine proper control gains, the system-model's equations of motion must be linear. At present they are not. There are two sources of nonlinearity: the torque about the pivot axis due to beam weight is a nonlinear function of θ , and curvature is a nonlinear function of lateral beam deflection. Two methods of linearization will be presented. In the first, the EOM are assumed to be piecewise linear. Each EOM-set is approximately true over a small range of θ . In the second scheme, nonlinear terms are added directly to the motor torque to cancel nonlinearities of $EOM(\theta)$. The nonlinearities of $EOM(q_a)$ are considered to be insignificant, and are simply ignored*.

Scheme 1 - Linearization About Discrete Angles

The effects of curvature nonlinearity are small relative to those due to gravity. Ignoring all terms containing $\sum q_1 q_j$ therefore, reduces the equations of motion (Eq. 2.10) to

*They are ignored now when determining control parameters. However, for the mathematical simulations to be discussed in Chapter 5, the fully-nonlinear equations are used.

$$\text{EOM}(\theta): \quad \frac{m\ell^2}{3} \ddot{\theta} - \frac{m}{\ell} \sum_i \ddot{q}_i DS_i + \frac{mg\ell}{2} \sin\theta - \frac{mg}{\ell} \cos\theta \sum_i q_i D_i = K_A K_T V \quad (3.10)$$

$$\text{EOM}(q_a): \quad -\frac{m}{\ell} \ddot{\theta} DS_a + \frac{m}{\ell} \ddot{q}_a D_{aa} + EI q_a D_{aa}^2 - \frac{mg}{\ell} \cos\theta \sum_i q_i DD_{aj} - \frac{mg}{\ell} \sin\theta D_a = 0$$

The equations are now nonlinear only with respect to θ . Perturbation-type solutions can be determined for a sequence of discrete θ within the range of probable beam motion (say, from $\theta = -180^\circ$ to $\theta = +180^\circ$).

The matrices associated with Eq. (3.10) would then take the form:

$$M = \begin{bmatrix} \frac{m\ell^2}{3} & -\frac{m}{\ell} DS_1 & -\frac{m}{\ell} DS_2 & -\frac{m}{\ell} DS_3 \\ -\frac{m}{\ell} DS_1 & \frac{m}{\ell} D_{11} & 0 & 0 \\ -\frac{m}{\ell} DS_2 & 0 & \frac{m}{\ell} D_{22} & 0 \\ -\frac{m}{\ell} DS_3 & 0 & 0 & \frac{m}{\ell} D_{33} \end{bmatrix}$$

$$K = \frac{mg}{\ell} \begin{bmatrix} \frac{\ell^2}{2} \frac{\sin\theta}{\theta} & -D_1 \cos\theta & -D_2 \cos\theta & -D_3 \cos\theta \\ -\frac{\sin\theta}{\theta} D_1 & \frac{EI\ell}{mg} D_{11}^2 + DD_{11} \cos\theta & DD_{12} \cos\theta & DD_{13} \cos\theta \\ -\frac{\sin\theta}{\theta} D_2 & DD_{12} \cos\theta & \frac{EI\ell}{mg} D_{22}^2 + DD_{22} \cos\theta & DD_{23} \cos\theta \\ -\frac{\sin\theta}{\theta} D_3 & DD_{13} \cos\theta & DD_{23} \cos\theta & \frac{EI\ell}{mg} D_{33}^2 + DD_{33} \cos\theta \end{bmatrix}$$

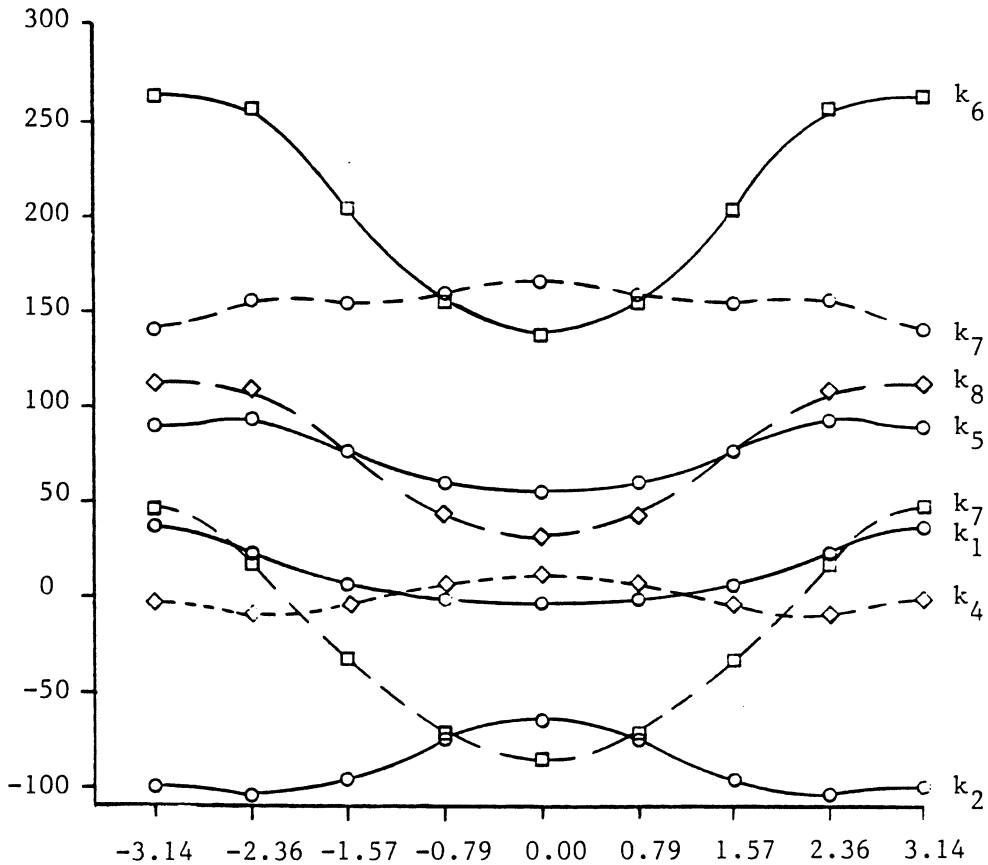
$$G = \begin{bmatrix} K_a K_m k_{sys} \\ 0 \\ 0 \\ 0 \end{bmatrix}$$

The "stiffness" matrix is evidently the only one dependent upon θ . The pole-assignment routine can then be applied for each θ_i within the defined sequence, the results being similar to those shown in Figure 3.1.

As far as implementation of these results on an actual system is concerned, two methods are available.

1. The gains could be used as they are (in tabular form). A digital controller would first sense system configuration, which includes the parameter θ , and then find the associated gains within a gain-table stored in memory. For θ between those values stored in the table, a closest match might be sufficient or, perhaps, linear interpolation could be used to determine more accurate gains.
2. Curves could be fit to each of the gains as a function of θ . The controller would then be provided with the corresponding mathematical expressions and could compute the gains for any angle θ within the specified range as need be. Depending upon the accuracy of the curve fits, this method may or may not prove to be superior to the one discussed above. However, it is this method which has in fact been implemented for experimental work, and not the other. Least-squares curve fits using Legendre polynomials are used.

θ	k_1	k_2	k_3	k_4	k_5	k_6	k_7	k_8
-5.50	16.67	- 53.68	-44.94	19.58	33.30	120.88	118.21	17.93
-4.71	23.55	- 74.25	- 8.75	8.29	43.08	162.53	104.64	43.26
-3.93	32.93	- 90.82	30.45	-0.55	64.03	218.41	110.38	80.31
-3.14	36.03	- 97.99	45.88	-4.09	90.31	264.28	141.05	113.19
-2.36	22.56	-103.36	15.59	-8.38	93.80	256.96	157.23	108.99
-1.57	6.81	- 94.79	-33.33	-4.62	75.48	204.26	155.46	73.69
-0.79	-0.79	- 75.24	-71.00	6.34	60.41	155.69	160.89	42.75
0.00	-2.71	- 65.42	-85.22	11.85	55.33	137.02	165.63	31.37
0.79	-0.81	- 75.24	-71.00	6.34	60.35	155.57	160.76	42.67
1.57	6.81	- 94.77	-33.32	-4.61	75.46	204.21	155.42	73.66
2.36	22.61	-103.32	15.64	-8.35	93.88	257.09	157.39	109.09
3.14	36.03	- 97.99	45.88	-4.09	90.31	264.28	141.05	113.19
3.93	32.92	- 90.84	30.41	-0.58	64.03	218.41	110.40	80.33
4.71	23.55	- 74.22	- 8.71	8.28	43.05	162.47	104.56	43.25
5.50	16.67	- 53.68	-44.94	19.58	33.30	120.88	118.21	17.83
6.28	13.40	- 45.41	-61.00	24.02	31.84	106.79	128.51	10.01

Figure 3.1 Gains as a function of θ

Scheme 2 - Direct Cancellation of Nonlinear Terms

Beam deflection can be thought of as a summation of two parts, as indicated in Fig. 3.2. y_θ corresponds to the equilibrium shape for a particular angle θ , and y_t corresponds to time-dependent motion.

$$y = y_\theta + y_t = \sum_i (q_{i\theta} + q_{it}) \phi_i$$

At static equilibrium, the EOM(q_a) degenerate into

$$EI q_{a\theta} D_a^2 + \frac{mg}{\ell} \cos\theta \sum_i q_{i\theta} DD_{ai} - \frac{mg}{\ell} \sin\theta D_a = 0$$

when all time-dependent terms are set to zero. Solving for $\frac{mg}{\ell} \sin\theta D_a$ and substituting this back into the EOM(q_a) results in

$$-\frac{m}{\ell} \ddot{\theta} DS_a + \frac{m}{\ell} \ddot{q}_a D_{aa} + EI q_{at} D_a^2 + \frac{mg}{\ell} \cos\theta \sum_i q_{it} DD_{ai} = 0 \quad (3.11)$$

The nonlinear term involving DD_{ai} exists only because of the large-deflection assumption. This term is therefore ignored (set equal to zero), leaving us with the linear equations

$$-\frac{m}{\ell} \ddot{\theta} DS_a + \frac{m}{\ell} \ddot{q}_a D_{aa} + EI q_{at} D_a^2 = 0 \quad (3.12)$$

Of course, the equation of motion for θ (Eq. 3.10) is still nonlinear, but this is easily remedied. Terms to cancel these nonlinearities can be added to the control input V_a . If

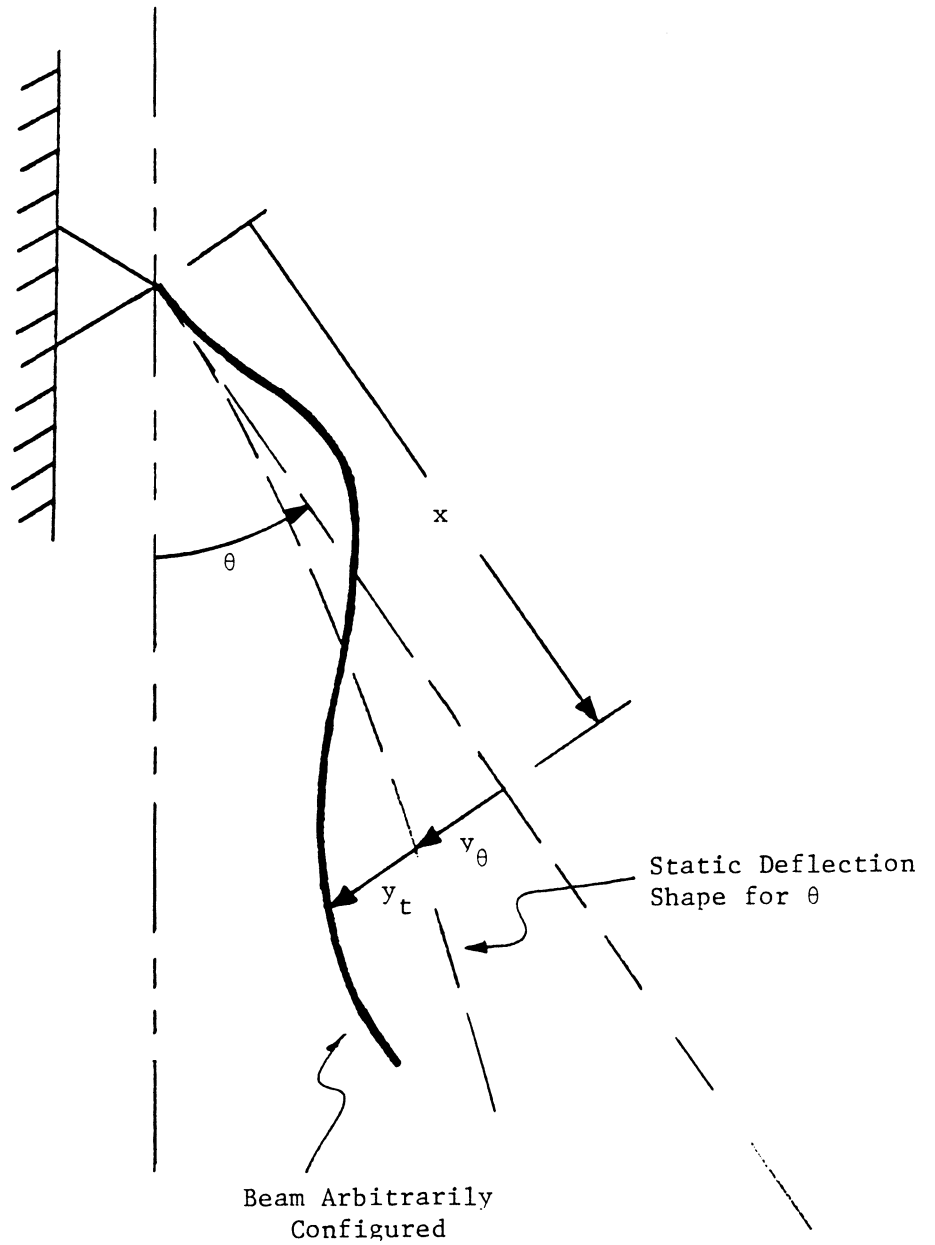


Figure 3.2 Breakdown of lateral beam displacement for control scheme 2

$$V_a = V_1 + \frac{1}{K_a K_m} \left(\frac{mg\ell}{2} \sin\theta - \frac{mg}{\ell} \cos\theta \sum q_i D_i \right)$$

then EOM(θ) becomes

$$\frac{m\ell^2}{3} \ddot{\theta} - \frac{m}{\ell} \sum_i \ddot{q}_i D S_i = K_a K_m V_1 \quad (3.13)$$

leaving us with a linear system (Eqs. 3.12 and 3.13) for which only one set of gains needs to be determined.

These beautifully linear equations, however, seem to be unusable. The gain-determination scheme will not recognize θ , q_a and q_{at} as all being system states because they are interdependent (see Eq. 3.11). However, this problem can be overcome if we assume that \ddot{q}_a is a much stronger function of \ddot{q}_{at} than it is of $\ddot{q}_{a\theta}$. The term \ddot{q}_{at} relates to beam oscillation about static equilibrium configurations, and $\ddot{q}_{a\theta}$ relates to the change of these static equilibrium configurations as the beam rotates. With this in mind, it seems legitimate to approximate \ddot{q}_a as:

$$\ddot{q}_a = \ddot{q}_{a\theta} + \ddot{q}_{at} \approx \ddot{q}_{at}$$

The terms q_{at} have now replaced the q_a as system states, and the equations of motion become

$$\text{EOM}(\theta): \quad \frac{m\ell^2}{3} \ddot{\theta} - \frac{m}{\ell} \sum_i \ddot{q}_{it} D S_i = K_a K_m V_1 \quad (3.14)$$

$$\text{EOM}(q_a): \quad -\frac{m}{\ell} \ddot{\theta} D S_a + \frac{m}{\ell} \ddot{q}_{at} D_{aa} + EI q_{at} D_a^2 = 0 \quad (a = 1, 2, 3)$$

CHAPTER 4 COMPUTER IMPLEMENTATION

The computer is used for all major aspects of this work. A set of programs was written to determine what gains force the system to adopt prespecified eigenvalues. Simulation programs were written so that system response to these calculated gains could be analytically determined. Finally, the experimental controller (an IBM PC) required programs in order to calculate real-time motor commands from the incoming instrumentation signals.

Gain Determination

The method used to calculate gains (for Eq. 3.5) was presented in Brogan [7] and is discussed in detail in chapter 3. The programs have been constructed to be as versatile as possible. They are presented in Appendix A. Some of the nice features of this program-set are:

- any linear system may be evaluated, or any nonlinear system which can be linearized about a particular variable (in our case θ). All non-varying system-specific information is contained in a data file (defined by user). All varying system-specific information is contained in a subroutine (KGETA). It is therefore as easy to change systems as it is to change these two files.
- different systems may be of different sizes (the sizes of the A, B, K, and T matrices of Eq. (3.5) change).
- it is possible to determine solutions assuming incomplete state feedback. The number of feedback signals and the particular signals fed back are at the user's discretion; or, it may not be up to the user at all -- it may not be possible to outfit the system

with proper instrumentation to measure all system states. One should realize, though, that the number of eigenvalues which can be 'placed' is never more than the number of feedback signals; the remaining eigenvalues are affected, but in an uncontrollable way.

- gains are necessarily computed for discrete values of the varying parameter (θ). However, the range of θ and the intervals between discrete θ can be specified by the user.
- the gains can be left as a table of discrete values, or curves can be fit in a least-squares sense to the gains as functions of θ . The particular curves to be used can also be specified. For our beam problem Legendre polynomials are used.
- eigenvalues of the system using the calculated gains (discrete values or curve-fits) are computed using the IMSL routine EIGCC. This provides invaluable information to the analyst. First of all, he can determine whether the discrete gains do indeed produce the desired eigenvalues. These discrete gains can be purposely truncated to see if the system is sensitive to the gains. The curve-fits may only be useful over a certain range of θ ; eigenvalue determination will uncover this. Lastly, it is necessary to know where the 'unspecified' eigenvalues fall when using incomplete state feedback.

Simulation

The commercially available Advanced Continuous Simulation Language (ACSL) was used to generate computer simulations of system response. This Fortran-based language was specifically developed by Mitchell and

Gauthier [8] to solve simultaneous, nonlinear differential equations, and is therefore well-suited to our beam problem. The heart of the code is in its numerical integration capabilities. The user is given six different integration schemes to choose from (Runge Kutta, etc.) or, if desired, he can define his own. A variety of other useful functions are available within the ACSL Library. The things which the user will find it absolutely necessary to input are:

- a. system equations
- b. length of time over which to determine solutions
- c. time increment to be used for numerical integration error checking
- d. initial conditions of system states.

Also available in ACSL are plotting routines. Two simulation programs were written, each corresponding to a different control algorithm. The system model is the same for both, represented mathematically by Eq. (2.10) (fully nonlinear). A listing of the program relating to control scheme 1 is provided in Appendix B.

Motor Commands

An IBM PC is used to generate motor commands. It is equipped with an Intel 8087 math co-processor and a Data Translation DT2801 analog/digital converter. It is possible to program the IBM PC in a variety of languages, even when used in conjunction with the DT2801 card. For this application Fortran was used. Two programs were written -- one for each control algorithm. These programs can be found in Appendix C. The portion of the programs relating to A/D conversion may be unfamiliar to

the reader. It might prove helpful to consult Data Translation's PCLAB manual SP0141-CA.

Gains are computed under the assumption that the system is entirely continuous (analog). This, of course, is not true. The PC can only sample beam configuration at discrete times. However, if this sample rate is fast relative to beam motion then the computed gains should be adequate. Efforts were made to minimize calculation-time within the programs; however, this was seen to have only a minor effect upon the rate at which the PC could output control commands. Currently, we are able to output commands at approximately 40 Hz. The factor which limits the system to this relatively low value is unknown.

CHAPTER 5 ADDITIONAL FEATURES OF THE EXPERIMENTAL SYSTEM

A. Instrumentation

Instrumentation provides the controller with real-time information about system configuration. The number and type of sensors needed depend upon the particular information desired. The control algorithm discussed in chapter 3

$$V_a = R(t) - KTP$$

uses the system states, P , as feedback signals; the states, then, are desired.

Figure 5.1 shows the system outfitted with instrumentation. A resistive potentiometer is used to indicate motor shaft angle. Its terminals are excited with $\pm E_{in}$ volts so that wiper potential will be zero volts when motor shaft angle is zero degrees. Strain gages affixed to the beam along its length ($s = 0.7$ in., 6.7 in., 14.65 in.) are used to deduce beam shape. The strain is measured at three locations so that the three generalized coordinates of the Ritz approximation (Eq. 2.2) can be uniquely determined. The voltages produced by the gages are small, and need to be amplified before being used by the controller. To do this, each strain measurement location is fitted with four strain gages rather than one (a bridge circuit), and the outputs are sent through battery-operated bridge amplifiers.

The relationships between bridge outputs and generalized coordinates were determined by clamping the cantilever-end of the beam within a

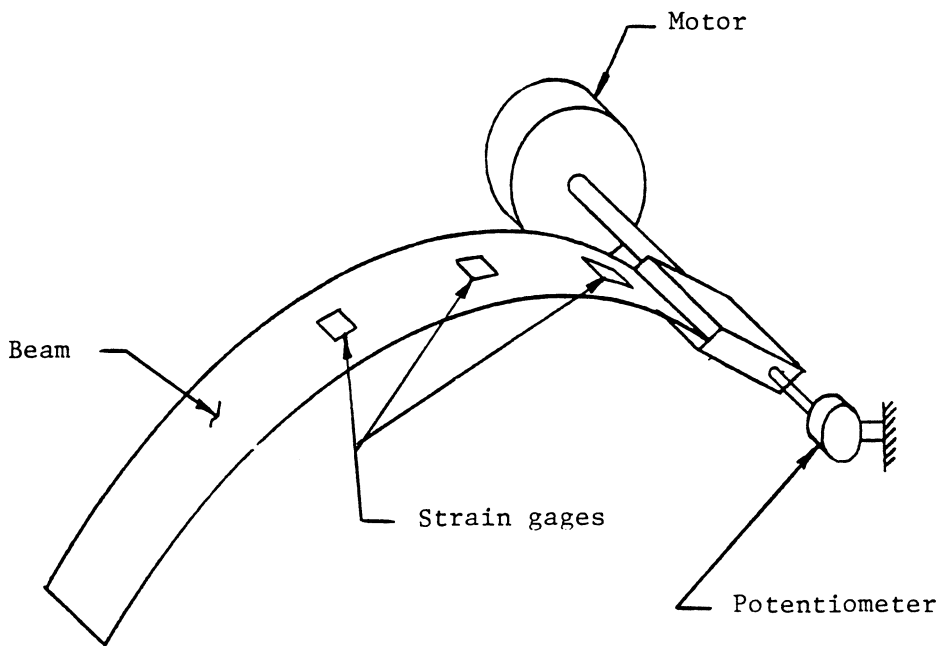


Figure 5.1 Instrumentation

wise and deflecting the beam tip. The resultant shape was taken as equivalent to that associated with the beam's first natural mode. Bridge outputs were measured and then related to calculated strains. It would have been difficult to statically configure the beam into second and third mode shapes. Instead, the strains at the gage locations were theoretically calculated for these configurations, and the corresponding bridge outputs were inferred from the first-mode-shape results. The transformation matrix associated with the sought-after relationship

$$U = TX$$

is obviously problem-specific, depending upon such things as the orientation of gages with respect to beam axes and the charge of the bridge amplifier batteries. Nevertheless, the transformation matrix for this system will be presented:

$$T = \begin{bmatrix} 3.183^* & 0 & 0 & 0 \\ 0 & .0749^{**} & .3109 & .8156 \\ 0 & .0681 & -.0125 & -.6255 \\ 0 & .0418 & -.3783 & -.0651 \end{bmatrix}$$

The system has been instrumented to sense position but not velocity. Velocity is determined by differentiating the position signals. This is done digitally, and is discussed in more detail on

*units are volts per radian.

**units are volts per in./in.

page 41.

Signal noise is an insidious problem for all measurement systems. To minimize its effects, shielded cable has been used where possible. Differentiation is particularly sensitive to noise. A digital filter is therefore used in conjunction with this operation, but, again, this is discussed on page 41.

Model Modifications in View of Experimental Realities

There still remain two features of the experimental system which have yet to be incorporated into the system model. One is damping, and the other is the difficulty associated with determining time-derivatives of the feedback signals.

Damping Effects

As the beam rotates and vibrates, it experiences frictional resistance. This is due primarily to the motor-brushes, the shaft bearings, and the air. It would be difficult to predict these things analytically. Instead, simple experiments were performed. So as to preserve system linearity (at the cost of accuracy), all damping has been assumed to be linearly related to beam velocity.

Angular beam motion

The nominal position of the beam is vertically downward, with $\theta = 0^\circ$ (see Fig. 2.1). The beam was manually lifted to approximately $\theta = 90^\circ$ and then released. Little, if any, beam vibration was induced. The resultant angular motion (pendulum motion) was recorded using a strip

chart recorder; it was seen to resemble a damped sinusoid.

As mentioned, $\dot{\theta}$ damping is assumed to be of the form $c\dot{\theta}$; it is necessary to determine the coefficient c . To do so, a damping term of this form was added to the ACSL simulation model. The coefficient c was varied until simulation results matched the experimental results, which occurred at $c = .7 \text{ lb}_f \text{ sec in (N s m)}$.

Flexible Beam Motion

A similar procedure was followed to determine the additional damping due to beam deflection. The beam was detached from the motor's output shaft and clamped instead within a vise. The beam tip was displaced and then released, and the outputs of the strain gages were recorded with, again, a strip chart recorder.

The generalized forces associated with this damping had to be derived for the ACSL simulation model so that the proper damping coefficient could be determined as above. For an infinitesimal element, it was assumed that

$$dF_{\text{damp}} = c_1 \dot{y} ds = c_1 \sum_i \dot{q}_i \phi_i ds$$

The same damping coefficient c_1 was assumed for each mode. Integrating over beam length produces

$$F_{\text{damp}} = \int_0^{\ell} c_1 \sum_i \dot{q}_i \phi_i ds = c_1 \sum_i \dot{q}_i D_i = c_1 D^T \dot{X}$$

These, then, were the terms added to the equations of motion. The

coefficient c_1 was varied until simulation and experimental results coincided, which occurred with $c_1 = 1.632 \text{ E-5 lb}_f \text{ sec/in (N s/m)}$.

Incorporation into System Model

Previously, the mathematical model of the system was expressed as in Eq. (3.5).

$$\dot{P} = [A - BKT] P + BR(t) \quad (3.5)$$

where

$$A = \begin{bmatrix} 0 & I \\ -M^{-1}K & 0 \end{bmatrix}$$

The addition of damping-effects modifies this A matrix to

$$A' = \begin{bmatrix} 0 & I \\ -M^{-1}K & -M^{-1}C \end{bmatrix} \quad (5.1)$$

Time-Derivative of Feedback Signals

It is notoriously difficult to differentiate real-world (noisy) signals. One method often used to circumvent the problem with analog control is to construct an electrical circuit which both filters and differentiates the incoming signal. It was decided to use a similar approach here. However, an actual analog circuit was not used; instead, the equations relating to the circuit were incorporated in the digital controller. The block diagram for the equivalent analog circuit and its magnitude and phase characteristics are shown in Fig. 5.2. It can be

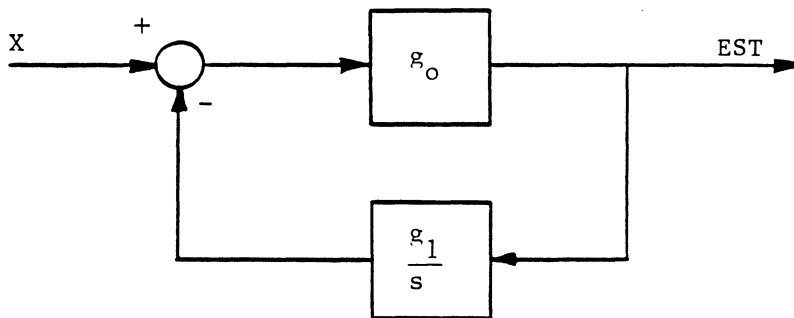
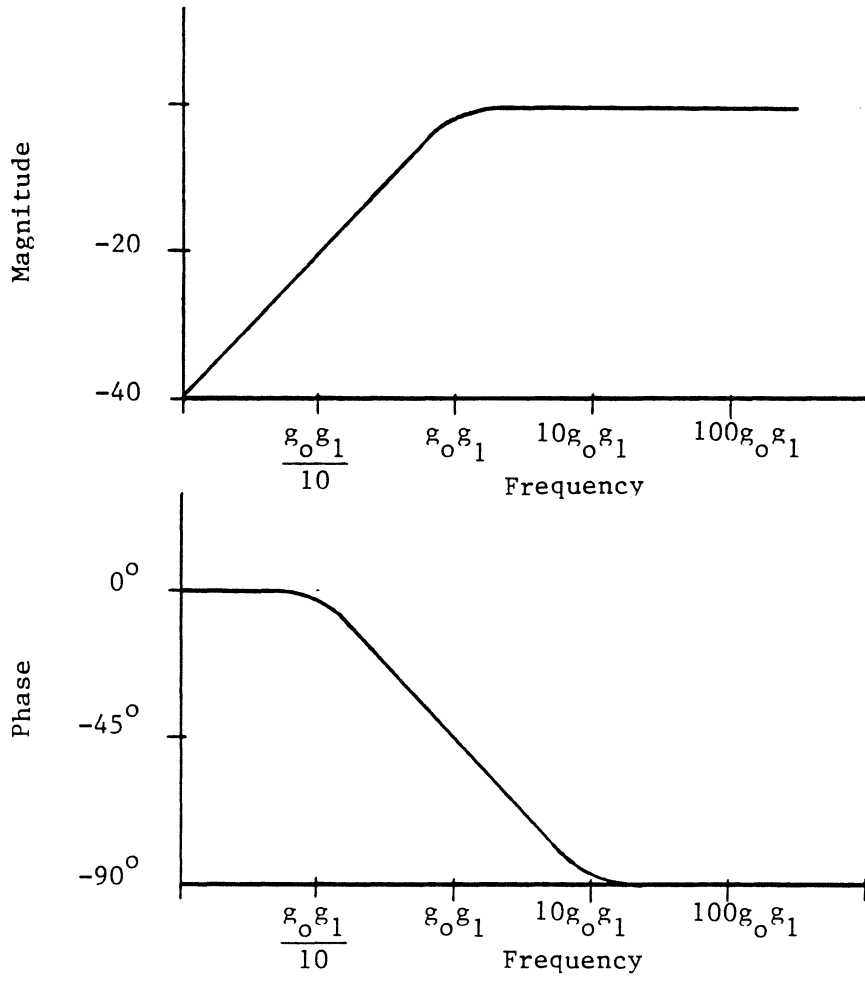


Figure 5.2 Estimator block diagram and response characteristics

seen that the circuit does indeed differentiate the signal, but magnitude and phase are distorted as a function of signal frequency. The circuit, therefore, has dynamics of its own, and must be incorporated into the system model. Previously, the model had eight states; with the inclusion of four "estimators" (of the time-derivatives), one for each feedback signal, the number of states increases to twelve. The continuous-time expression for each of these estimators is

$$\text{EST}_i = \frac{1}{g_1} \frac{d X_i}{dt} - \frac{1}{g_1 g_o} \frac{d \text{EST}_i}{dt} \quad (5.2)$$

Filter/Differentiator Circuit

As indicated previously, the control device (IBM PC) samples instrumentation signals continually but not continuously. The above-given estimator equation must therefore be modified to account for this. Referring to the estimator diagram of Fig. 5.2, the input/output relationship (Laplace domain) is seen to be

$$\text{EST}(s) = g_o \left[X(s) - \frac{g_1}{s} \text{EST}(s) \right]$$

Rearranging,

$$\text{EST}(s) \left[1 + \frac{g_o g_1}{s} \right] = g_o X(s)$$

$$\frac{\text{EST}(s)}{X(s)} = \frac{g_o}{1 + \frac{g_o g_1}{s}} = \frac{s/g_1}{\frac{s}{g_o g_1} + 1}$$

This may be written as

$$\frac{EST(s)}{X(s)} = \frac{EST(s)}{h(s)} \frac{h(s)}{X(s)}$$

where

$$\frac{EST(s)}{h(s)} = \frac{s}{g_1} \quad (5.3)$$

and

$$\frac{h(s)}{X(s)} = \frac{1}{\frac{s}{g_0 g_1} + 1} = \frac{g_0 g_1}{s + g_0 g_1}$$

Rearranging,

$$h(s)[s + g_0 g_1] = g_0 g_1 X(s)$$

$$sh(s) = g_0 g_1 [X(s) - h(s)]$$

In the time domain, this would take the finite-difference representation:

$$\left(\frac{\Delta h}{\Delta t}\right)_n = g_0 g_1 [X_n - h_n] \quad (5.4)$$

where n refers to a particular moment in time. Noting that

$$\Delta h = h_n - h_{n-1}$$

Eq. (5.4) becomes

$$h_n = h_{n-1} + g_o g_1 [X_n - h_n] \Delta t$$

or

$$h_n [1 + g_o g_1 \Delta t] = h_{n-1} + g_o g_1 X_n \Delta t$$

and

$$h_n = \frac{h_{n-1} + g_o g_1 X_n \Delta t}{1 + g_o g_1 \Delta t} \quad (5.5)$$

Referring back to Eq. (5.3),

$$EST(s) = \frac{1}{g_1} s h(s)$$

or

$$EST_n = \frac{1}{g_1} \left(\frac{\Delta h}{\Delta t} \right)_n$$

From Eq. (5.4)

$$EST = g_o [X_n - h_n] \quad (5.6)$$

The parameter h_n is that which was sought. Eqs. (5.5) and (5.6), then, are the equations needed to implement the estimators on the digital

controller.

Incorporation into System Model

With damping the model is mathematically represented by Eqn. (5.7).

Recall that A' is defined in Eq. (5.1).

$$\dot{P} = [A' - BKT] P + BR(t) \quad (5.7)$$

or,

$$\begin{bmatrix} \dot{X} \\ \ddot{X} \end{bmatrix} = \begin{bmatrix} 0 & I \\ -M^{-1}K & -M^{-1}C \end{bmatrix} \begin{bmatrix} X \\ \dot{X} \end{bmatrix} - \begin{bmatrix} 0 \\ M^{-1}G \end{bmatrix} [K_1 \quad K_2] \begin{bmatrix} T_{11} & T_{12} \\ T_{21} & T_{22} \end{bmatrix} \begin{bmatrix} X \\ \dot{X} \end{bmatrix} + \begin{bmatrix} 0 \\ M^{-1}G \end{bmatrix} R(t)$$

The estimators add four more states and modify the gain vector:

$$\begin{bmatrix} \dot{X} \\ \ddot{X} \\ \dot{H} \end{bmatrix} = \begin{bmatrix} 0 & I & 0 \\ -M^{-1}K & -M^{-1}C & 0 \\ 0 & g_0 I & -g_0 g_1 I \end{bmatrix} \begin{bmatrix} X \\ \dot{X} \\ H \end{bmatrix} - \begin{bmatrix} 0 \\ M^{-1}G \\ 0 \end{bmatrix} [K_1 \quad 0 \quad K_2] \begin{bmatrix} T_{11} & T_{12} & 0 \\ T_{21} & T_{22} & 0 \\ 0 & 0 & I \end{bmatrix} \begin{bmatrix} X \\ \dot{X} \\ H \end{bmatrix} + \begin{bmatrix} 0 \\ M^{-1}G \\ 0 \end{bmatrix} R(t)$$

or

$$\dot{p}^* = [A^* - B^*K^*T^*] p^* + B^*R(t) \quad (5.8)$$

The stars (*) are used to indicate that this system includes derivative estimators, and is now composed of 12 states.

CHAPTER 6 ANALYTICAL RESULTS

Two types of analytical results are presented here: those corresponding to Gain Determination and those corresponding to Simulation of System Response.

A. Gain Determination

The outputs of the program FFlex (presented in Chapter 4) include closed-loop gains and related system eigenvalues. The gains may be given as discrete values for a sequence of different θ , or as the coefficients associated with least-squares curve-fits. The number of feedback signals may be varied, as well as the particular signals to be fed back. If r signals are used, a maximum of r eigenvalues can be 'placed'. The model used for gain-determination may be of a smaller order than that used for eigenvalue determination; this might indicate whether a cruder model would be sufficient for control purposes.

In this section, results are presented for a variety of situations which cover the gamut of FFlex usage. Gain-determination is considerably simpler for control scheme 2, and this is therefore discussed first.

1. Control Scheme 2 - Nonlinear Cancellation

The system model as developed in chapter 3 and modified in chapter 5 involves 12 states. These include positions, velocities, and velocity estimates. Feedbacks are only possible for 8 of these states: positions and velocity estimates.

Gains are not functions of θ . All θ -related terms originally in

the EOM have been cancelled using direct nonlinear torque feedback, or have simply been ignored (set equal to zero). Therefore, only one gain-set needs to be determined. Typical results will now be presented. Examples have been selected to show both the usefulness and limitations of the method. The reader will perhaps find it best to read through the examples in order, as results are often compared to these previously discussed. The system-related information for each example is formatted as follows:

inputs to program	}	a. number of feedback signals
		b. the signals that are fed back
		c. desired eigenvalues
outputs of program	}	d. gains
		e. eigenvalues

1. The eigenvalues for the uncontrolled system are

eigenvalues	-8.00	-8.00	-8.00
	-8.00	-762.79	0.00
	-13.71-1116.60	-88.78	-6.52-134.90
	-13.70+1116.60	-1.55	-6.52+134.90

Table 6.1 Example 1a

The pole at $0.0+i0.0$ is due to no θ feedback, and the multiple poles at $-8.0+i0.0$ are due to the estimator circuits. If the filters associated with these estimators were given a cut-off frequency (see Fig. 5.1) of 16 rad/sec the multiple poles would be at $-16.0+i0.0$.

2.	number of feedbacks	1	
	signals fed back	θ	
	desired eigenvalues	$-0.5+10.0$	
	gains k_{θ}	$= 0.1363^*$	
	eigenvalues		
		-8.00	-8.00
		-8.00	-762.39
		-13.67+1116.59	-89.38
		-13.67-1116.59	-0.50
			-8.00
			-1.03
			-6.47+134.94
			-6.47-134.94

Table 6.2 Example 2a

With only one feedback signal, only one eigenvalue can be specified. Furthermore, one can only be assured of attaining this eigenvalue if it is real (because the system matrix $A^* - B^*kT$ is real, all complex eigenvalues must be accompanied by their conjugates). The specified eigenvalue was attained, and the other eigenvalues are seen to be stable. In fact, these other eigenvalues did not move much from the uncontrolled case. This is because we have not tried to radically change any of the eigenvalues presented in example 1a.

3.	number of feedbacks	1	
	signals fed back	θ	
	desired eigenvalues	$-0.7+11.0$	
	gains k_{θ}	$= .4162$	
	eigenvalues		
		-8.00	-8.00
		-8.00	-761.56
		-13.59+1116.56	-0.74+11.00
		-13.59-1116.56	-0.74-11.00
			-8.00
			-90.61
			-6.37+135.02
			-6.37-135.02

Table 6.3 Example 3a

The one specified eigenvalue is complex, and it was not attained exactly, although nearly so. When specifying an odd number of complex

*All gains have units of volts per volt (v/v).

eigenvalues, such good results cannot be expected as a rule. While it may not be desired to specify an odd number of complex eigenvalues, it may be unavoidable: we may seek to move eigenvalues which happen to be complex, but may find that some of our feedback signals are 1. unreliable because of hardware problems, or 2. too sensitive to gains, and therefore unusable.

4.	number of feedbacks	1		
	signals fed back	θ		
	desired eigenvalues	$-3.0+i100.0$		
	gains	$k_{\theta} = -0.9480$		
	eigenvalues			
		-8.00	-8.00	-8.00
		-8.00	-765.57	-84.56
		-13.97+i116.69	1.28	-6.88+i34.62
		-13.97-i116.69	-3.00	-6.88-i34.62

Table 6.4 Example 4a

An attempt was made to throw the eigenvalue originally at the origin all the way out to -3.0 . This was achieved, but at the expense of causing its sister pole to go unstable. One can easily imagine what a root-locus plot for these two poles would show; however, with eight gains to play with and twelve system eigenvalues, the root-locus plot loses its attraction.

5.	number of feedbacks	2		
	signals fed back	$\theta, EST1$		
	desired eigenvalues	$-3.0+i1.0$		
		$-3.0-i1.0$		
	gains	$k_{\theta} = 0.7522$	$k_{est1} = 2.9378$	
	eigenvalues			
		-8.00	-8.00	-751.62
		-3.00-i1.01	-12.69-i116.39	-5.53-i36.02
		-3.00+i1.01	-12.69+i116.39	-5.53+i36.02
		-8.00	-105.63	-1.86

Table 6.5 Example 5a

Two feedback signals are used: θ and EST1. It is now a simple matter to achieve eigenvalues having real parts of -3.0 . However, it can also be seen that another pole has been forced to -1.86 . One must be somewhat careful, then, when choosing desired eigenvalues. Using a trial and error process, one can usually iterate to satisfactory results very quickly (unless radical pole movements are attempted).

6.	number of feedbacks	2		
	signals fed back	θ , EST1		
	desired eigenvalues	-3.0		
		-5.0		
	gains	$k_{\theta} = .2086$	$k_{est1} = 1.9275$	
	eigenvalues			
		-8.00	-8.00	-756.36
		-8.00	$-13.11-1116.46$	$-5.92-135.60$
		-98.76	$-13.11+1116.46$	$-5.92+135.60$
		-5.00	-3.00	-0.37

Table 6.6 Example 6a

An attempt to place poles even further to the left causes the pole at -1.86 to slip over to -0.37 .

7.	number of feedbacks	2		
	signals fed back	θ , EST1		
	desired eigenvalues	$-2.00+110.0$		
		$-2.00-110.0$		
	gains	$k_{\theta} = -54.14$	$k_{est1} = 31.239$	
	eigenvalues			
		-8.00	-8.00	-823.64
		-8.00	22.72	-27.32
		$-18.45+1122.42$	$-16.20+146.09$	-2.00
		$-18.45-1122.42$	$-16.20-146.09$	-2.00

Table 6.7 Example 7a

The two poles being specified were moved back towards the imaginary axis, but their imaginary components were substantially increased.

These poles were attained, but again at the expense of forcing a pole to the unstable region.

8.	number of feedbacks	3		
	signals fed back	θ , ST1, EST1		
	desired eigenvalues	-3.00+11.0	-4.0	
		-3.00-11.0		
	gains	$k_{\theta} = 0.6037$	$k_{ST1} = 34.668$	$k_{est1} = 0.5011$
	eigenvalues			
		-8.00	-8.00	-834.13
		-8.00	-4.00	-23.03
		-9.39-1124.36	-3.00-11.00	-7.81-141.85
		-9.39+1124.36	-3.00+11.00	-7.81+141.85

Table 6.8 Example 8a

Three feedbacks are now used, and our freedom of pole-placement has increased. The desired poles are achieved, and the other poles remain in suitable locations.

9.	number of feedbacks	3		
	signals fed back	θ , ST1, EST1		
	desired eigenvalues	-5.00+11.0	-6.0	
		-5.00-11.0		
	gains	$k_{\theta} = 0.6647$	$k_{st1} = 44.46$	$k_{est1} = -.0185$
	eigenvalues			
		-6.00-144.27	-8.00	-853.85
		-6.00+144.27	-8.00	-8.00
		-7.04+1125.77	-5.76+10.06	-5.06+11.04
		-7.04-1125.77	-5.76-10.06	-5.06-11.04

Table 6.9 Example 9a

More negative poles were sought, and gotten (although not exactly).

The remaining cases are presented without comment.

10.	number of feedbacks	4		
	signals fed back	$\theta, ST1, ST2, ST3$		
	desired eigenvalues	-5.00+i1.0	-13.71+i117.0	
		-5.00-i1.0	-13.71-i117.0	
	gains	$k_{\theta} = 0.7425$	$k_{ST1} = 14.776$	
		$k_{ST2} = 11.456$	$k_{ST3} = 45.289$	
	eigenvalues			
		-8.00	-8.00	-8.00
		-8.00	-773.50	-26.12
		-46.75	-13.71-i117.00	-5.01-i0.99
		-9.74	-13.71+i117.00	-5.01+i0.99

Table 6.10 Example 10a

11.	number of feedbacks	4		
	signals fed back	$\theta, ST1, ST2, ST3$		
	desired eigenvalues	- 5.00+i1.0	-13.71+i117.0	
		-11.0	-13.71-i117.0	
	gains	$k_{\theta} = .7293$	$k_{ST1} = 14.706$	
		$k_{ST2} = 11.403$	$k_{ST3} = 45.136$	
	eigenvalues			
		-8.00	-8.00	-8.00
		-8.00	-773.48	-24.89
		-13.71-i117.00	-4.77-i0.93	-47.25
		-13.71+i117.00	-4.77+i0.93	-10.97

Table 6.11 Example 11a

12.	number of feedbacks	8			
	signals fed back	$\theta, ST1, ST2, ST3, EST1, EST2, EST3, EST4$			
	desired eigenvalues	$-5.0+i1.0$	$-13.71+i117.0$	$-6.52+i34.9$	-26.5
		$-5.0-i1.0$	$-13.71-i117.0$	$-6.52-i34.9$	$-773.$
	gains	$k_{\theta}=13.401$	$k_{ST1}=-45.41$	$k_{ST2}=-61.00$	
		$k_{ST3}=24.024$	$k_{est1}=31.844$	$k_{est2}=106.79$	
		$k_{est3}=128.51$	$k_{est4}=10.015$		
	eigenvalues				
		$-6.52-i34.90$	$-13.71-i117.00$	$-5.01-i0.99$	
		$-6.52+i34.90$	$-13.71+i117.00$	$-5.01+i0.99$	
		-773.00	-26.47	-51.60	
		-8.00	-8.00	-8.00	

Table 6.12 Example 12a

13.	number of feedbacks	8			
	signals fed back	$\theta, ST1, ST2, ST3, EST1, EST2, EST3, EST4$			
	desired eigenvalues	$-10.0+i1.0$	$-17.71+i117.0$	$-9.52+i34.9$	-26.5
		$-10.0-i1.0$	$-17.71-i117.0$	$-9.52-i34.9$	$-773.$
	gains	$k_{\theta}=29.111$	$k_{ST1}=-77.62$	$k_{ST2}=46.685$	
		$k_{ST3}=6.3808$	$k_{est1}=34.601$	$k_{est2}=160.72$	
		$k_{est3}=59.867$	$k_{est4}=39.481$		
	eigenvalues				
		$-27.18+i1.88$	$-17.71+i116.99$	$-9.52+i34.89$	
		$-27.18-i1.88$	$-17.71-i116.99$	$-9.52-i34.89$	
		-773.00	-8.00	$-9.99+i1.65$	
		-7.77	-8.00	$-9.99-i1.65$	

Table 6.13 Example 13a

In general, it is seen that the pole-placement algorithm works well. It can also be noted that:

- Each of the feedback signals is seen to affect certain poles more than others. When seeking to relocate specific poles, then, it is worthwhile to determine which signals are most effective.
- The uncontrolled-system poles were presented in Example 7a. If only minor modifications to these poles are attempted, and if the

appropriate signals are used, the necessary gains will, more often than not, be small. Large gains can cause a variety of problems (this will be discussed in Chapter 7). If major pole changes are attempted, unstable poles often result.

- As more feedback signals are used, we have more freedom to place poles without fear of inducing instabilities.

2. Control Scheme 1 - Linearization About Discrete Angles

The A^* matrix of Equation (5.8) is a function of θ , and so are the gains. In this scheme the gains are computed for a sequence of θ . The appropriate gains for the θ not in the sequence are related in some logical manner to those that are in the sequence, perhaps using one of the methods indicated in Fig. 6.1.

The results to be presented below do, in fact, relate to a step function and a curve-fit. The range of θ over which a solution is to be found, as well as the interval between discrete θ , can be specified by the user. The results here are for a range of ± 6.282 rad. ($\pm 360^\circ$)[†] and an interval size of .1963 rad (11.25°). An appropriate value for interval size, of course, depends upon the behavior of the true gain curves. If smooth, the interval size can be increased. If erratic, this should be decreased.

[†]It is assumed that a multi-turn potentiometer is used for θ - measurement.

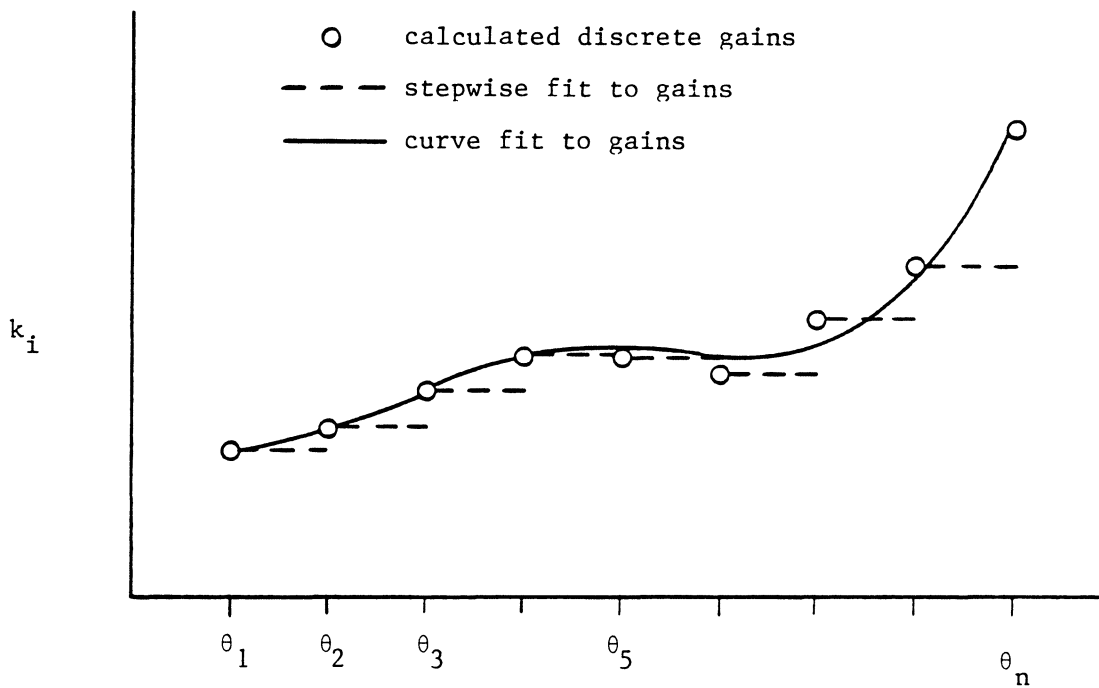


Figure 6.1 Gains as a function of θ

a. Discrete Gains -- Step Functions

To each discrete θ there corresponds a set of discrete gains to control the system. As θ changes, we need to use the appropriate gain-set. All comments made in section A of this chapter regarding gains are true for these gains as well. The difference here is that for each eigenvalue set there are 64 gain-sets (each set corresponds to a specific value of θ within the sequence defined in Chapter 3). Because of the volume of numbers involved, only one solution as generated by the program FFlex is presented here. Furthermore, only 16 of the 64 gain-sets and eigenvalue-sets are shown. The format of these results, Table 6.14, is similar to that used in the previous section.

Number of feedbacks 8
 signals fed back $\theta, ST1, ST2, ST3, EST1, EST2, EST3, EST4$
 desired eigenvalues $-5.0+11.0 \quad -13.71+1117.0 \quad -6.52+134.9 \quad -26.5+10.0$
 $-5.0-11.0 \quad -13.71-1117.0 \quad -6.52-134.9 \quad -773.+10.0$

range of θ -5.50 radians \rightarrow 6.28 radians
 interval size 0.79 radians

Gains (as function of θ)

θ	K_{θ}	K_{ST1}	K_{ST2}	K_{ST3}	K_{EST1}	K_{EST2}	K_{EST3}	K_{EST4}
-5.50	16.67	- 53.68	-44.94	19.58	33.30	120.88	118.21	17.93
-4.71	23.55	- 74.25	- 8.75	8.29	43.08	162.53	104.64	43.26
-3.93	32.93	- 90.82	30.45	-0.55	64.03	218.41	110.38	80.31
-3.14	36.03	- 97.99	45.88	-4.09	90.31	264.28	141.05	113.19
-2.36	22.56	-103.36	15.59	-8.38	93.80	256.96	157.23	108.99
-1.57	6.81	- 94.79	-33.33	-4.62	75.48	204.26	155.46	73.69
-0.79	-0.79	- 75.24	-71.00	6.34	60.41	155.69	160.89	42.75
0.00	-2.71	- 65.42	-85.22	11.85	55.33	137.02	165.63	31.37
0.79	-0.81	- 75.24	-71.00	6.34	60.35	155.57	160.76	42.67
1.57	6.81	- 94.77	-33.32	-4.61	75.46	204.21	155.42	73.66
2.36	22.61	-103.32	15.64	-8.35	93.88	257.09	157.39	109.09
3.14	36.03	- 97.99	45.88	-4.09	90.31	264.28	141.05	113.19
3.93	32.92	- 90.84	30.41	-0.58	64.03	218.41	110.40	80.33
4.71	23.55	- 74.22	- 8.71	8.28	43.05	162.47	104.56	43.25
5.50	16.67	- 53.68	-44.94	19.58	33.30	120.88	118.21	17.83
6.28	13.40	- 45.41	-61.00	24.02	31.84	106.79	128.51	10.01

eigenvalues (as functions of θ)

θ	Set of 12 Eigenvalues			
-5.4967	772.96	-7.83	-6.95	-52.10
	-6.55+134.91	-13.71-1117.04	-25.25	-3.96
	-6.55-134.91	-13.71+1117.04	-8.00	-8.00
-4.7115	-773.16	-8.00	-30.67	-8.03
	-4.08+12.71	-6.41+134.83	-13.72-1116.83	-49.26
	-4.08-12.71	-6.41-134.83	-13.72+1116.83	-8.00
-3.9263	-772.84	-8.00	-23.00	-53.14
	-8.25+10.69	-6.61+134.96	-13.70+1117.13	-3.46
	-8.25-10.69	-6.61-134.96	-13.70-1117.13	-8.00
-3.1410	-772.89	-6.00	-7.87	-51.99
	-6.56+135.00	-13.70+1117.10	-26.22	-4.07
	-6.56-135.00	-13.70-1117.10	-8.00	-8.00

Table 6.14 Example 1b

θ	Sets of 12 Eigenvalues			
-2.3558	-773.10 -6.47+134.84 -6.47-134.84	-7.77 -13.72-1116.91 -13.72+1116.91	-28.05 -4.74+12.00 -4.74-12.00	-50.78 -8.00 -8.00
-1.5705	-773.14 -6.48+134.74 -6.48-134.74	-26.86 -5.06+11.89 -5.06-11.89	-8.00 -13.72+1116.87 -13.72-1116.87	-51.15 -7.88 -8.00
-0.7853	-773.00 -6.51-134.91 -6.51+134.91	-8.00 -4.99+10.47 -4.99-10.47	-27.07 -13.71+1117.00 -13.71-1117.00	-51.36 -7.70 -8.00
0.0000	-772.86 -6.54+135.06 -6.54-135.06	-7.50 -13.70-1117.12 -13.70+1117.12	-6.97 -26.66 -8.00	-51.81 -3.27 -8.00
0.7852	-772.89 -6.56-134.95 -6.56+134.95	-8.00 -7.79+10.46 -7.79-10.46	-24.38 -13.70+1117.09 -13.70-1117.09	-52.56 -3.64 -8.00
1.5705	-773.04 -6.52+134.80 -6.52-134.80	-8.00 -5.58+10.82 -5.58-10.82	-24.96 -13.71+1116.95 -13.71-1116.95	-52.08 -7.85 -8.00
2.3557	-773.14 -6.46+134.81 -6.46-134.81	-8.00 -4.60+12.15 -4.60-12.15	-28.62 -13.72-1116.98 -13.72+1116.98	-50.43 -7.81 -8.00
3.1410	-773.06 -6.51+134.88 -6.51-134.88	-8.00 -4.51+11.33 -4.51-11.33	-28.16 -13.72-1116.96 -13.72+1116.96	-50.79 -8.08 -8.00
3.9262	-772.89 -6.60+134.93 -6.60-134.93	-8.00 -7.97+10.49 -7.97-10.49	-23.72 -13.70-1117.10 -13.70+1117.10	-52.82 -3.59 -8.00
4.7115	-772.89 -6.56+134.93 -6.56-134.93	-8.00 -7.51+10.33 -7.51-10.33	-23.93 -13.70-1117.08 -13.70+1117.08	-52.73 -4.48 -8.00
5.4967	-773.14 -3.99+12.28 -3.99-12.28	-8.00 -6.43-134.90 -6.43+134.90	-30.94 -13.73-1116.89 -13.73+1116.89	-7.99 -49.21 -8.00
6.2820	-772.96 -6.55+134.90 -6.55-134.90	-6.48 -13.71-1117.03 -13.71+1117.03	-7.99 -25.19 -8.01	-52.16 -4.28 -8.00

Table 6.14 Example 1b (continued)

b. Least-Squares Curve-Fits

A linear summation of weighted polynomials are fit in a least-squares sense to the computed gains

$$\text{Gain}_j = \sum_i a_{ji} F_i$$

where a_{ji} = i th coefficient of Gain_j
 F_i = i th polynomial

An IMSL routine (LINGF) is used to find the appropriate a_{ji} given a set of F_i . For the F_i , it is important to use orthogonal polynomials since polynomials of the form $F_k = \theta^{k-1}$ are very ill-conditioned for moderate to large problems. Legendre polynomials are used here, and are detailed in Appendix D. The same feedbacks and desired eigenvalues as requested for the discrete-gain case (section a) are requested here as well, to allow for comparison. The results are presented in Table 6.15. Again, only 16 of the 64 eigenvalue-sets is reported. The gain coefficients a_{ji} are given rather than gains themselves.

Number of feedbacks 8
 signals fed back θ , ST1, ST2, ST3, EST1, EST2, EST3, EST4
 desired eigenvalues $-5.0+i1.0$ $-13.71+i117.0$ $-6.52+i34.9$ $-26.5+i0.0$
 $-5.0-i1.0$ $-13.71-i117.0$ $-6.42-i34.9$ $-773.0+i0.0$

range of θ -5.4967 rad. \rightarrow 6.2820 rad.
 interval size 0.7852

gain coefficients a_{ji}

i								
j	1	2	3	4	5	6	7	8
1	17.968	-80.691	-17.219	4.323	63.145	188.407	137.033	62.810
2	-0.279	0.353	-1.047	0.173	-0.971	-1.989	-0.872	-1.380
3	9.764	26.990	0.173	15.901	-33.846	-58.783	-38.486	-40.355
4	-0.615	0.815	-2.395	0.431	-2.204	-4.503	-1.933	-3.158
5	-30.368	29.684	-105.711	14.903	-19.617	-93.419	35.434	-58.118
6	-0.952	1.227	-3.693	0.637	-3.347	-6.851	-2.903	-4.792
7	22.121	-29.293	84.404	-15.174	43.674	114.584	13.468	76.422
8	-1.258	1.652	-4.823	0.841	-4.475	-9.151	-3.976	-6.381

eigenvalues (as functions of θ)

θ	Set of 12 Eigenvalues			
-5.4967	-773.25	-8.00	-35.38	-46.21
	-6.35-134.90	-3.28+12.76	-13.74+i116.80	-8.00
	-6.35+134.90	-3.28-12.76	-13.74-i116.80	-8.00
-4.7115	-772.93	-6.38	-7.67	-52.35
	-6.54+i34.91	-13.70+i117.05	-24.81	-4.94
	-6.54-134.91	-13.70-i117.05	-8.00	-8.00
-3.9263	-772.79	-9.16	-8.25	-53.52
	-6.64+i34.99	-13.69+i117.18	-21.86	-3.32
	-6.63-134.99	-13.69-i117.18	-8.00	-8.00
-3.1410	-773.00	-8.00	-27.61	-51.15
	-6.52-134.93	-4.70+i0.99	-13.71-i117.01	-7.91
	-6.52+134.93	-4.70-i0.99	-13.71+i117.01	-8.00
-2.3558	-773.19	-8.00	-29.39	-49.92
	-6.45+i34.78	-4.45+i2.33	-13.73-i116.83	-7.81
	-6.45-134.78	-4.45-i2.33	-13.73+i116.83	-8.00
-1.5705	-773.17	-8.00	-26.82	-51.19
	-6.49-134.75	-5.19+i1.83	-13.72-i116.87	-7.63
	-6.49+134.75	-5.19-i1.83	-13.72+i116.87	-8.00

Table 6.15 Example 2b

θ	Sets of 12 Eigenvalues			
-0.7853	-772.93 -6.55+134.93 -6.55-134.93	-6.17 -13.70-1117.06 -13.70+1117.06	-8.76 -25.01 -8.00	-52.26 -3.94 -8.00
0.0000	-772.80 -6.57+135.08 -6.57-135.08	-8.00 -7.83+10.55 -7.83-10.55	-24.99 -13.69-1117.18 -13.69+1117.18	-52.56 -3.02 -8.00
0.7852	-772.89 -6.56-134.95 -6.56+134.95	-8.00 -7.73+10.94 -7.73-10.94	-24.56 -13.70-1117.09 -13.70+1117.09	-52.52 -3.62 -8.00
1.5705	-773.10 -6.50+134.77 -6.50-134.77	-8.00 -5.26+11.62 -5.26-11.62	-26.13 -13.71-1116.90 -13.71+1116.90	-51.52 -7.86 -8.00
2.3557	-773.19 -6.45+134.77 -6.45-134.77	-8.00 -4.40+12.32 -4.40-12.32	-29.31 -13.73-1116.83 -13.73+1116.83	-49.94 -7.97 -8.00
3.1410	-773.04 -6.51+134.89 -6.51-134.89	-8.00 -4.58+11.23 -4.58-11.23	-28.15 -13.72-1116.87 -13.72+1116.67	-50.81 -7.93 -8.00
3.9262	-772.82 -6.62+134.97 -6.62-134.97	-22.44 -8.47+10.49 -8.47-10.49	-8.00 -13.70+1117.15 -13.70-1117.15	-53.33 -3.38 -8.00
4.7115	-772.87 -6.57+134.93 -6.57-134.93	-8.00 -7.82+10.44 -7.82-10.44	-23.32 -13.69-1117.10 -13.69+1117.10	-52.97 -4.23 -8.00
5.4967	-773.20 -6.37+134.91 -6.37-134.91	-47.57 -3.52+12.66 -3.52-12.66	-8.00 -13.73+1116.83 -13.73-1116.83	-8.00 -33.54 -8.00
6.2820	-772.68 -6.73+134.90 -6.73-134.90	-8.00 -14.88+115.91 -14.88-115.91	-3.05 -13.68-1117.25 -13.68+1117.25	-55.24 -8.01 -8.00

Table 6.15 Example 2b (continued)

2. Simulation of System Response

Discussion in this section is limited to a few representative test-cases which show that simulation results are indeed realistic. Diagrams of initial beam configuration for these test-cases are included with the response plots. No control is used in the first three cases.

It appears that little explanation is required; system response is as one might expect. In Test Case 1, for instance, the beam is given an initial deflection of 1 radian and then released. The beam seeks the configuration relating to minimum potential energy, which occurs with $\theta = 0$ radians. This will be attained in an oscillatory fashion as the simulation plot suggests. The other five test-cases produce equally understandable results. It might be mentioned, as a side note, that the particular control values used for Test Case 5, subcase b, were determined by the pole-location program FFlex.

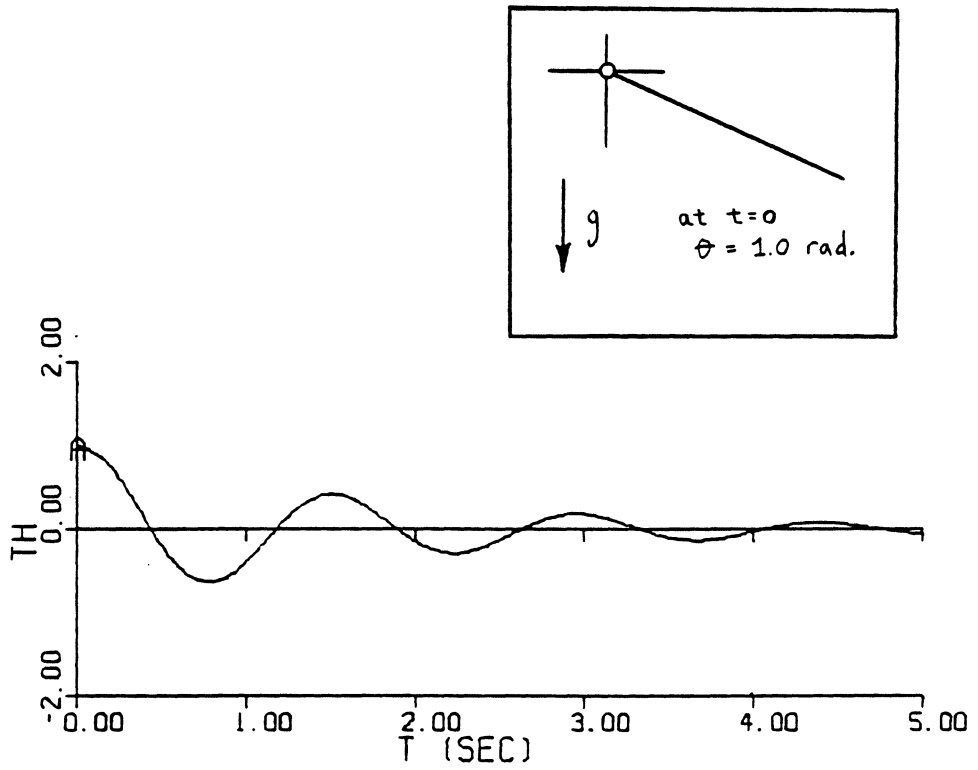


Figure 6.2 Simulation example 1; θ initially displaced

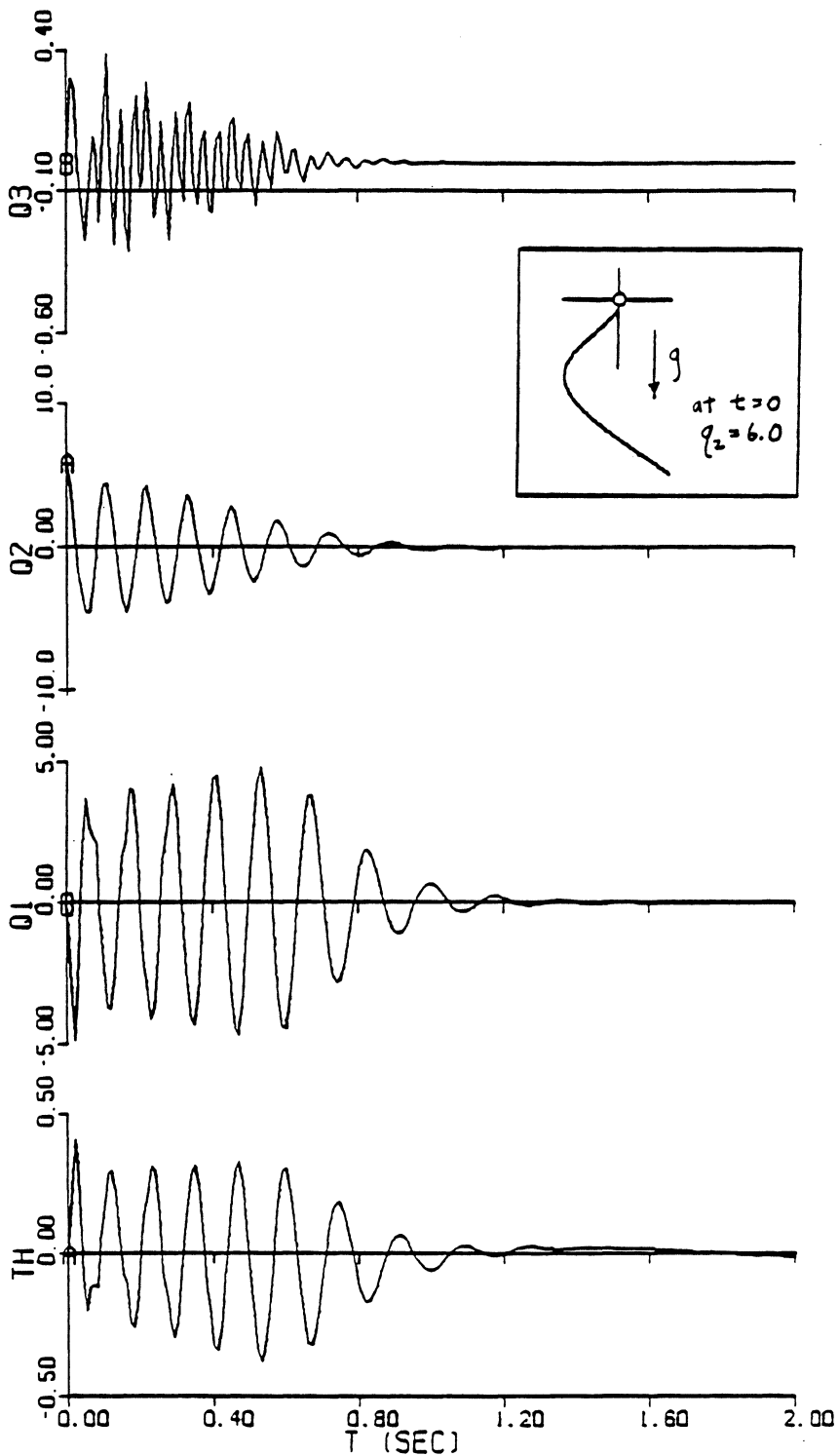


Figure 6.3 Simulation example 2; q_2 initially displaced

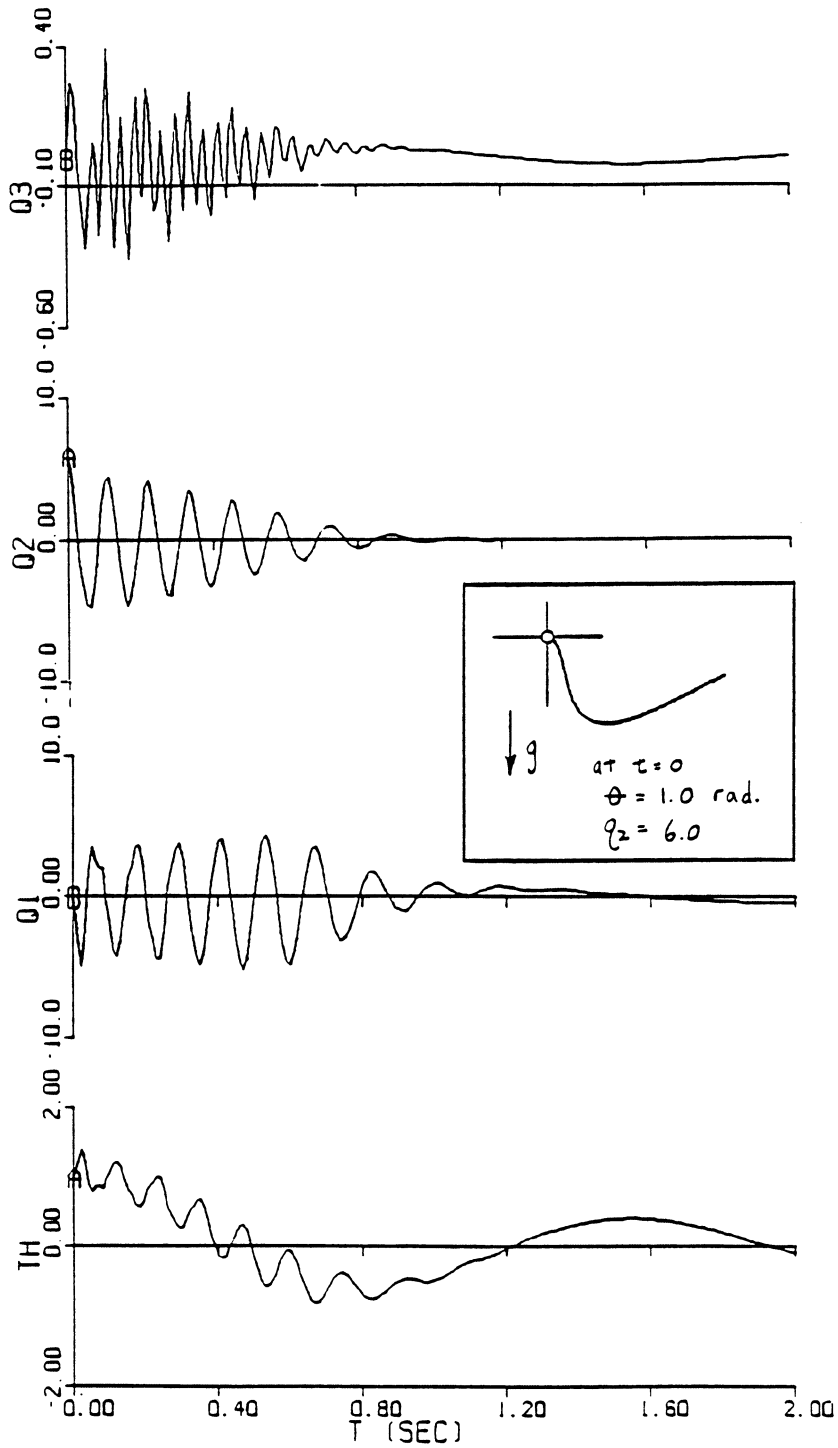


Figure 6.4 Simulation example 3; θ and q_2 initially displaced

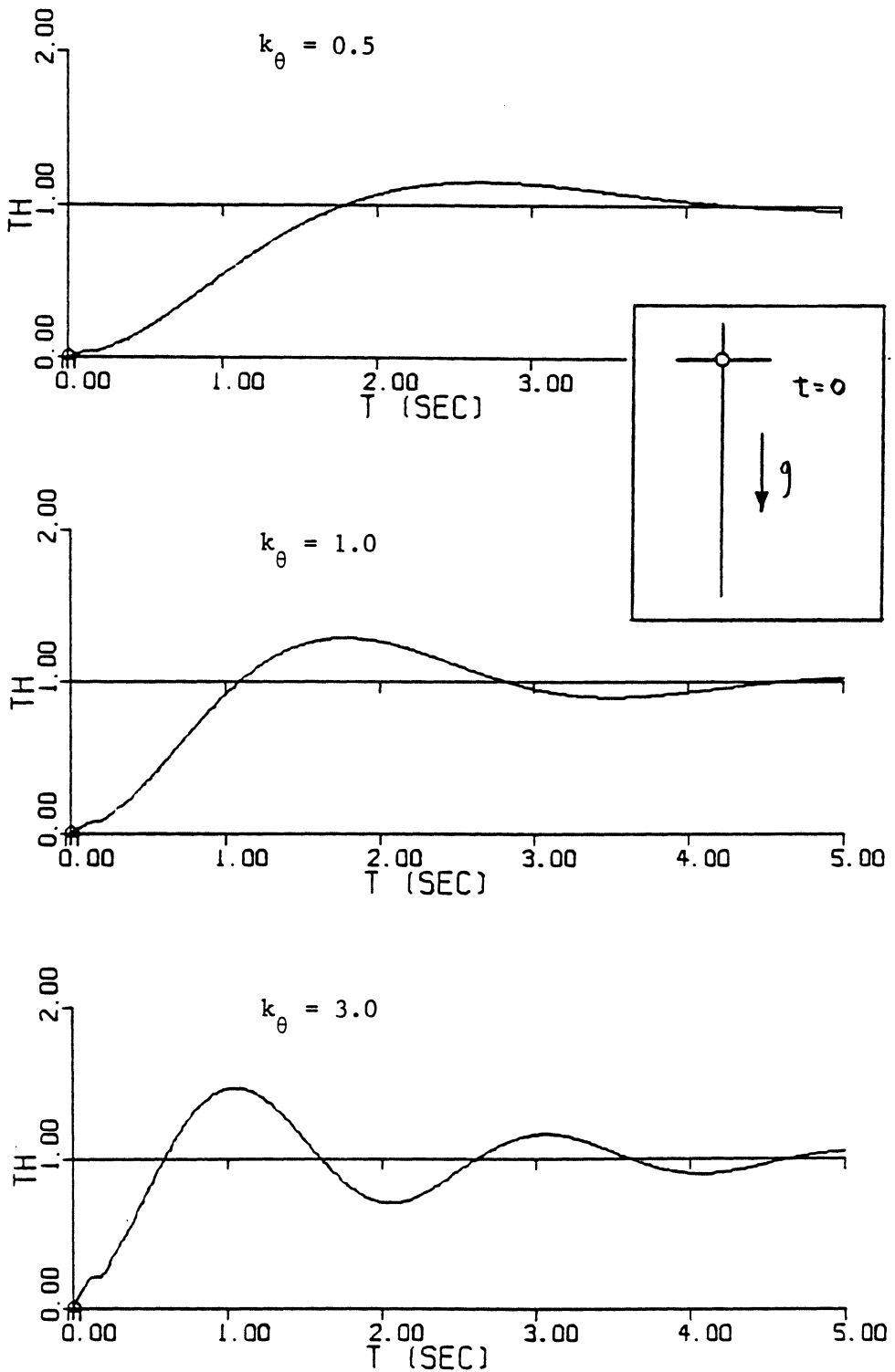


Figure 6.5 Simulation example 4; shoot for $R(t)=1$. rad. with three different feedbacks on θ

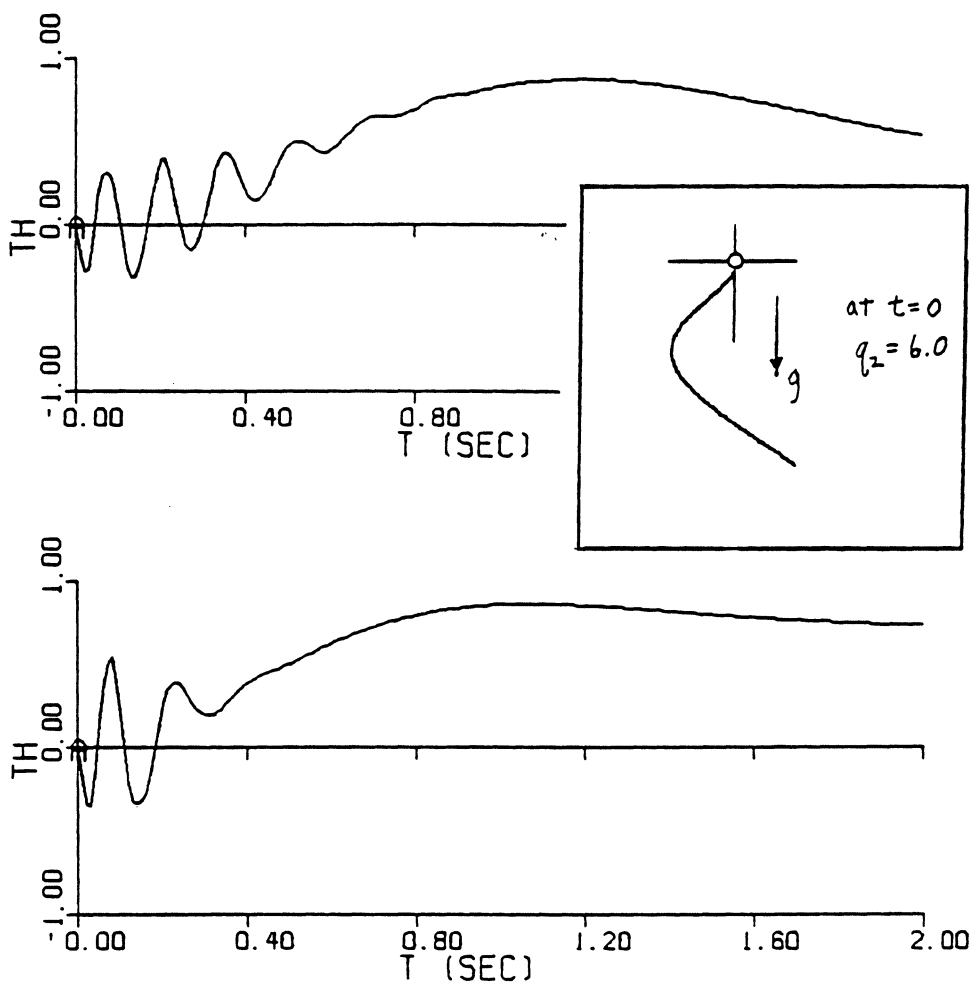


Figure 6.6 Simulation example 5; shoot for $R(t)=1$. rad., q_2 initially displaced

CHAPTER 7 EXPERIMENTAL RESULTS

The experimental system was tested, modified and retested as necessary until it functioned properly. Appropriate gains were then supplied to the controller, and experiments were conducted. The results of these experiments are presented and discussed in this section. Comparisons are also made with analytical results.

Experimental results include strip-chart records of system configuration as a function of time. System-related velocities are not shown because a) this information was not as readily available and because b) this is not absolutely necessary for the reader to understand (even if somewhat qualitatively) system response.

System velocities are not shown; however, they are used in the control algorithms. It therefore seems appropriate to show at least one response plot for the derivative estimator 'circuits' being used. Refer to Fig. 7.1.* The input signal comes from a sinewave generator, and its frequency is continuously increased. The estimator signal seems to correspond nicely to the magnitude and phase diagrams of Fig. 5.2. Additionally, the effect of using a digital computer to compute this estimate is obvious. At low frequencies the discrete nature of the output signal is not significant. At high frequencies, however, it becomes very significant, and eventually a point is reached when the controller does not receive signals fast enough to compute appropriate derivatives.

*Because of the volume of figures presented and discussed in this chapter, it seems most beneficial to the reader to group them all as a unit. They can be found at the end of the chapter.

For each experimental test, the following system-related information is provided:

1. initial conditions
2. plots of uncontrolled-system response
3. control gains to be used
4. computed eigenvalues of controlled system
5. plots of controlled-system response
6. plots of simulation response

Initial conditions could not be specified arbitrarily. Certain constraints existed. First of all, it was necessary to completely avoid initial velocities. These would have been difficult to determine, which in turn would have made it difficult to produce corresponding simulations. Secondly, it was necessary to fully exploit beam flexibility; otherwise the effectiveness of control to suppress vibrations would not be as evident.

System responses using the two control schemes will be reported one after the other.

1. System Response, Control Scheme 1 (Linearization About Discrete Angles)

Recall that this scheme uses least-squares polynomial curve-fits to the gains, and that the form of these curve-fits is

$$\text{Gain}_j = \sum_{i=1}^8 a_{ji} F_i$$

The basis polynomials F_i are given in Appendix D. Rather than presenting specific gains, which change with θ , the coefficients a_{ji} are

given.

Four experiments were run. The initial conditions for each of these is the same; these are shown as Configuration 1 in Fig. 7.2. The free response of the system to this initial configuration is shown in Fig. 7.3. This response is characterized by a fair bit of vibration, which dies out in approximately 3.5 seconds.

a. Experiment Number 1

The curve-fit coefficients are given in Table 7.1. Evidently, only two feedback signals are used: θ and ST2. Also included in Table 7.1 are the computed system eigenvalues for $\theta = 0$. radians. All are stable. The poles closest to the imaginary axis are those at $-3.07 \pm 0.84i$. The multiple poles at -8.00 are directly related to the estimator filters. If a comparison is made between these twelve poles and the poles of the uncontrolled system (Table 6.1), one can note how very similar they are. In fact, the program FFlex was used in this case only to move the least-negative poles out to ≈ -3.00 . All other pole movement comes as a consequence of this. If radical pole movements are attempted, as was done on occasion, two things generally occur: other poles are catapulted into the right-half region of the s-plane (the unstable region), or the computed gains are seen to be extremely high - indicating that unreasonable and unattainable motor torques would be needed.

The experimental results are presented in Fig. 7.4. The system is shown as it is being initially configured, and the arrow indicates the point at which it was released. Major vibrations damp out rather

quickly, in about .67 seconds (compare with uncontrolled system), although some small, high-frequency vibration is evident in ST1 for times past this. The system is obviously stable, which corresponds well to the general location of computed eigenvalues; however, at this point nothing definite can be said about how this response relates to the particular eigenvalues that are computed. The simulation plot of system response is shown in Fig. 7.5. The correlation between experiment and simulation is satisfactory, although experimental results for ST1 exhibit slightly more oscillation.

b. Experiment Number 2

Curve-fit coefficients and related eigenvalues ($\theta = 0$ rad.) are given in Table 7.2. An attempt has been made to move the least-negative poles associated with experiment Number 1 further to the left (in the s -plane). Experimental response is shown in Fig. 7.6. Oddly enough, the high-frequency oscillation of ST1 is even more prevalent than before.

A possible reason for this can be deduced from pole location. More negative poles correspond to faster system response. It is conceivable for beam vibration to include a contribution from the fourth natural mode. The control algorithm, though, is aware of only three mode shapes. This fourth-mode contribution will therefore go uncorrected. At times, with various gain-sets, the beam did vibrate entirely at the fourth mode (four nodes were seen), resulting in a neutrally stable condition. Nothing can be done about this unless the Ritz approximation to beam shape is rewritten to include the fourth mode shape. This is not done. Therefore, it is suggested that poles not be sent "too" far

to the left. This boundary is, at present, indeterminate. It depends upon such things as the number and the particular feedback signals used, the rate at which the controller outputs commands, etc.

Simulation response is shown in Fig. 7.7. Again, the basic form of these curves compares well with experimental results, but some oscillation is unpredicted.

c. Experiment Number 3

Curve-fit coefficients and related eigenvalues ($\theta = 0$. rad) are given in Table 7.3. Poles have been moved even further to the left, and sure enough, some undesirable oscillations are induced. Experimental response is shown in Fig. 7.8. Simulation response is shown in Fig. 7.9.

d. Experiment Number 4

Curve-fit coefficients and related eigenvalues ($\theta = 0$. rad.) are given in Table 7.4. Two eigenvalues are positive, indicating that the system is unstable. The experimental results, Fig. 7.10, confirm this instability. The experimental results actually suggest a neutral stability, but the system could easily (it was seen) become unstable if acted upon by a variety of external forces (a quick hit with the hand). The simulation results of Fig. 7.11 also reflect the positive eigenvalues.

2. System Response, Control Scheme 2 (Nonlinear Cancellation)

The results of six experiments are reported.

a. Experiment Number 5

The initial conditions are shown as configuration 1 in Fig. 7.2. The corresponding free-response is shown in Fig. 7.3. Control gains and computed eigenvalues are given in Table 7.5. Feedback signals used are θ , ST2, ST3, EST1 and EST3. All eigenvalues are stable. In fact, the least-negative ones are at $-7.61 \pm 1.95i$. System response is shown in Fig. 7.12, and it is fairly nice. It was said earlier (experiment no. 2) that one should not attempt to place eigenvalues too far to the left (s-plane); yet, here all eigenvalues are fairly negative and the response is beautiful. It seems likely that the boundary past which is unwise to place poles depends upon which eigenvalues are placed using which feedback signals. For instance, there is a pole at the origin for the uncontrolled system. This pole is primarily associated with θ motion. Any attempts to move this signal with strain gage signals will be marginally successful at best.

Simulation response is shown in Fig. 7.13. Agreement between experimental and simulation results is excellent.

b. Experiment Number 6

The initial conditions are shown as configuration 1 in Fig. 7.2. Free response is shown in Fig. 7.3. Control gains and computed eigenvalues are given in Table 7.6. System response is shown in Fig. 7.14. Simulation response is shown in Fig. 7.15. The eigenvalues are slightly less negative than those of the previous experiment, and, correspondingly, transient response is seen to die less quickly. However, response is still nice, and correlation with simulation response is, again, very

good.

c. Experiment Number 7

The initial conditions are the same as for the preceding 6 experiments: configuration 1 of Fig. 7.2; free response: Fig. 7.3. Control gains and computed eigenvalues are given in Table 7.7. Position feedback but no 'velocity' feedback is used. Response, Fig. 7.16, is good, but not as good as previously seen. Also, low-frequency θ oscillation is seen; this is due to the small value for k_{θ} . Comparison between experimental and simulated results (Fig. 7.17) is fairly good.

d. Experiment Number 8

A new initial configuration is used; this is shown as configuration 2 of Fig. 7.2. Uncontrolled system response is shown in Fig. 7.18. Control gains and computed eigenvalues are given in Table 7.8. Eigenvalues are very negative and system response is desirable; see Fig. 7.19. The simulation plot of Fig. 7.20 predicts these experimental results rather well.

e. Experiment Number 9

The initial conditions are shown as configuration 3 in Fig. 7.2. Uncontrolled system response is shown in Fig. 7.21. Control gains and computed eigenvalues are given in Table 7.9. Controlled-system response, Fig. 7.22, is a vast improvement over uncontrolled system response. The simulation, Fig. 7.23, shows the basic skeleton of actual response, but the fast, transient oscillation is not anticipated.

f. Experiment Number 10

A slewing maneuver is attempted. The initial conditions and target destination are indicated in configuration 4 of Fig. 7.2. At $t = 0$ the beam is in its natural static equilibrium shape. Control gains and computed eigenvalues are given in Table 7.10. Uncontrolled system response is shown in Fig. 7.24 for signals θ and ST1 only. Controlled-system response is shown in Fig. 7.25. A slight improvement is seen, although nothing outstanding. The gain set used, also used for experiment number 8, seems better suited to vibration suppression than suppression of θ -oscillations. Simulation response is shown in Fig. 7.26. The basic character of actual response is reflected here, although not perfectly.

curve-fit coefficients a_{ji}									
j	i	1	2	3	4	5	6	7	8
1		-.200E+01	0.00	0.428E+02	0.00	0.00	0.00	0.00	0.00
2		0.505E-02	0.00	0.236E-01	0.00	0.00	0.00	0.00	0.00
3		0.212E+00	0.00	0.722E+00	0.00	0.00	0.00	0.00	0.00
4		-.143E-02	0.00	-.670E-02	0.00	0.00	0.00	0.00	0.00
5		-.143E-01	0.00	-.522E-01	0.00	0.00	0.00	0.00	0.00
6		0.947E-04	0.00	0.445E-03	0.00	0.00	0.00	0.00	0.00
7		0.169E-03	0.00	0.652E-03	0.00	0.00	0.00	0.00	0.00
8		-.172E-05	0.00	-.810E-05	0.00	0.00	0.00	0.00	0.00
eigenvalues at $\theta = 0.0$									
		-8.00		-8.00		-8.00		-8.00	
		-22.76+i34.69		-25.07+i113.18		-3.07-i0.84		-777.08	
		-22.76-i34.69		-25.07-i113.18		-3.07+i0.84		-14.66	

Table 7.1 Control gains and corresponding system eigenvalues
experiment number 1

curve-fit coefficients a_{ji}									
j	i	1	2	3	4	5	6	7	8
1		-.182E+01	0.00	0.298E+02	0.00	0.455E+01	0.00	0.278E+02	0.00
2		0.167E-01	0.00	0.119E-01	0.00	0.747E-01	0.00	0.280E+00	0.00
3		0.491E+00	0.00	0.399E+00	0.00	0.642E+00	0.00	0.276E+01	0.00
4		-.401E-02	0.00	-.162E-02	0.00	-.202E-01	0.00	-.753E-01	0.00
5		-.386E-01	0.00	-.109E-01	0.00	-.623E-01	0.00	-.270E+00	0.00
6		0.264E-03	0.00	0.414E-04	0.00	0.135E-02	0.00	0.508E-02	0.00
7		0.514E-03	0.00	-.638E-05	0.00	0.104E-02	0.00	0.442E-02	0.00
8		-.488E-05	0.00	-.391E-07	0.00	-.248E-04	0.00	-.943E-04	0.00
eigenvalues at $\theta = 0.0$									
		-8.00		-8.00		-766.50		-13.55	
		-26.92+i109.77		-7.81+i4.97		-24.38-i31.03		-3.27	
		-26.92-i109.77		-7.81-i4.97		-24.38+i31.03		-8.01	

Table 7.2 Control gains and corresponding system eigenvalues,
experiment number 2

curve-fit coefficients a_{ji}									
j	i	1	2	3	4	5	6	7	8
1		-.166E+01	0.00	0.217E+02	0.386E+02	0.175E+01	0.00	0.108E+02	0.00
2		0.287E-01	0.00	0.797E-01	-.364E+00	0.114E+00	0.00	0.534E+00	0.00
3		0.399E+00	0.00	0.443E+00	-.205E+01	0.632E+00	0.00	0.304E+01	0.00
4		-.604E-02	0.00	-.119E-01	0.706E-01	-.258E-01	0.00	-.118E+00	0.00
5		-.315E-01	0.00	-.221E-01	0.187E+00	-.589E-01	0.00	-.285E+00	0.00
6		0.332E-03	0.00	0.464E-03	-.376E-02	0.154E-02	0.00	0.687E-02	0.00
7		0.409E-03	0.00	0.198E-03	-.271E-02	0.944E-03	0.00	0.446E-02	0.00
8		-.533E-05	0.00	-.514E-05	0.597E-04	-.263E-04	0.00	-.116E-03	0.00

eigenvalues at $\theta = 0.0$

-8.00	-8.00	-747.86	-75.49
-17.64-i110.92	-9.81-i3.44	-9.85+i7.09	-3.64
-17.64+i110.92	-9.81+i3.44	-9.85-i7.09	-8.00

Table 7.3 Control gains and corresponding system eigenvalues, experiment number 3

curve-fit coefficients a_{ji}									
j	i	1	2	3	4	5	6	7	8
1		-.289E+01	0.00	0.673E+02	-.809E+02	0.960E+00	0.00	0.00	0.00
2		0.107E-01	0.00	-.284E+00	0.785E+00	-.654E-02	0.00	0.00	0.00
3		0.306E+00	0.00	-.444E+01	0.131E+02	-.954E-01	0.00	0.00	0.00
4		-.300E-02	0.00	0.801E-01	-.221E+00	0.185E-02	0.00	0.00	0.00
5		-.209E-01	0.00	0.455E+00	-.125E+01	0.890E-02	0.00	0.00	0.00
6		0.199E-03	0.00	-.530E-02	0.146E-01	-.123E-03	0.00	0.00	0.00
7		0.258E-03	0.00	-.676E-02	0.182E-01	-.132E-03	0.00	0.00	0.00
8		-.362E-05	0.00	0.957E-04	-.265E-03	0.221E-05	0.00	0.00	0.00

eigenvalues at $\theta = 0.0$

-8.00	-8.00	-8.00	-820.82
-31.44+i125.58	-3.78-i1.96	6.97-i46.98	-22.75
-31.44-i125.58	-3.78+i1.96	6.97+i46.98	-1.50

Table 7.4 Control gains and corresponding system eigenvalues, experiment number 4

gains	k_{θ}	k_{ST1}	k_{ST2}	k_{ST3}	k_{EST1}	k_{EST2}	k_{EST3}	k_{EST4}
	1.6771	0.00	27.773	30.729	2.3216	0.00	15.456	0.00
eigenvalues								
	-8.00		-8.00		-744.66		-7.90	
	-19.21-i107.68		-7.61-i1.95		-8.96-i5.01		-76.29	
	-19.21+i107.68		-7.61+i1.95		-8.96+i5.01		-9.14	

Table 7.5 Control gains and corresponding system eigenvalues, experiment number 5

gains	k_{θ}	k_{ST1}	k_{ST2}	k_{ST3}	k_{EST1}	k_{EST2}	k_{EST3}	k_{EST4}
	.55477	0.00	46.515	0.00	.30825	0.00	0.00	0.00
eigenvalues								
	-8.00		-8.00		-8.00		-771.57	
	-25.41-i110.39		-3.00+i1.00		-23.16+i30.39		-22.83	
	-25.41+i110.39		-3.00+i1.00		-23.16-i30.39		-4.00	

Table 7.6 Control gains and corresponding system eigenvalues, experiment number 6

gains	k_{θ}	k_{ST1}	k_{ST2}	k_{ST3}	k_{EST1}	k_{EST2}	k_{EST3}	k_{EST4}
	.56869	9.5313	40.602	0.00	0.00	0.00	0.00	0.00
eigenvalues								
	-8.00		-8.00		-8.00		-8.00	
	-792.60		-23.91-i115.27		-4.05-i0.46		-20.08+i38.64	
	-4.92-i0.01		-23.91+i115.27		-4.05+i0.46		-20.08-i38.64	

Table 7.7 Control gains and corresponding system eigenvalues, experiment number 7

gains	k_{θ}	k_{ST1}	k_{ST2}	k_{ST3}	k_{EST1}	k_{EST2}	k_{EST3}	k_{EST4}
	2.465	0.00	37.048	0.00	4.544	0.00	30.240	0.00
eigenvalues								
	-8.00		-8.00		-757.88		-9.39	
	-28.02+i105.55		-27.40+i25.09		-8.50+i1.88		-7.22+i0.92	
	-28.02-i105.55		-27.40-i25.09		-8.50-i1.88		-7.22-i0.92	

Table 7.8 Control gains and corresponding system eigenvalues, experiment number 8

gains	k_{θ}	k_{ST1}	k_{ST2}	k_{ST3}	k_{EST1}	k_{EST2}	k_{EST3}	k_{EST4}
	1.0446	0.00	45.261	0.00	1.500	0.00	9.917	0.00
eigenvalues								
	-8.00		-8.00		-768.50		-6.04	
	-27.20+i108.87		-26.44-i32.14		-4.86+i0.98		-10.00	
	-27.20-i108.87		-26.44+i32.14		-4.86-i0.98		-8.01	

Table 7.9 Control gains and corresponding system eigenvalues, experiment number 9

gains	k_{θ}	k_{ST1}	k_{ST2}	k_{ST3}	k_{EST1}	k_{EST2}	k_{EST3}	k_{EST4}
	2.465	0.00	37.048	0.00	4.544	0.00	30.240	0.00
eigenvalues								
	-8.00		-8.00		-757.88		-9.39	
	-28.02+i105.55		-27.40+i25.09		-8.50+i1.88		-7.22+i0.92	
	-28.02-i105.55		-27.40-i25.09		-8.50-i1.88		-7.22-i0.92	

Table 7.10 Control gains and corresponding system eigenvalues, experiment number 10

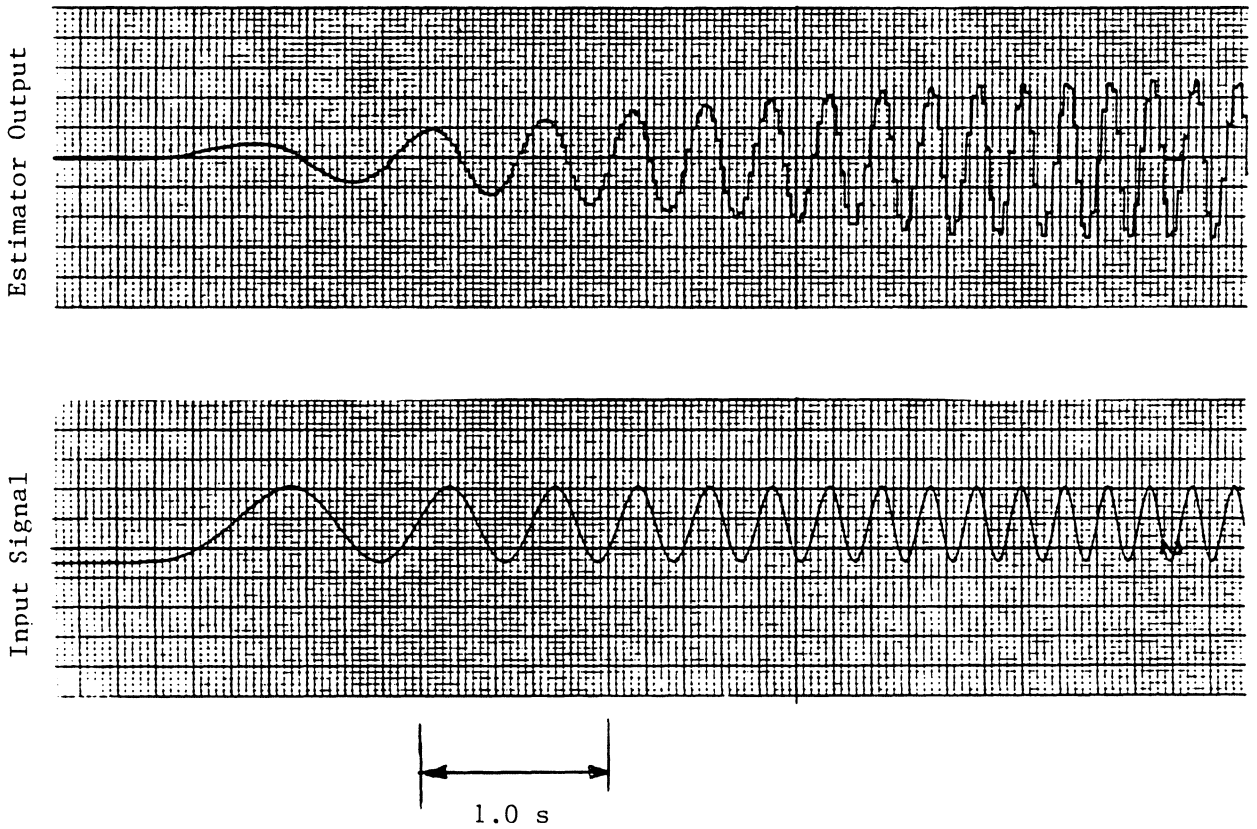


Figure 7.1 Estimator output as a function of frequency

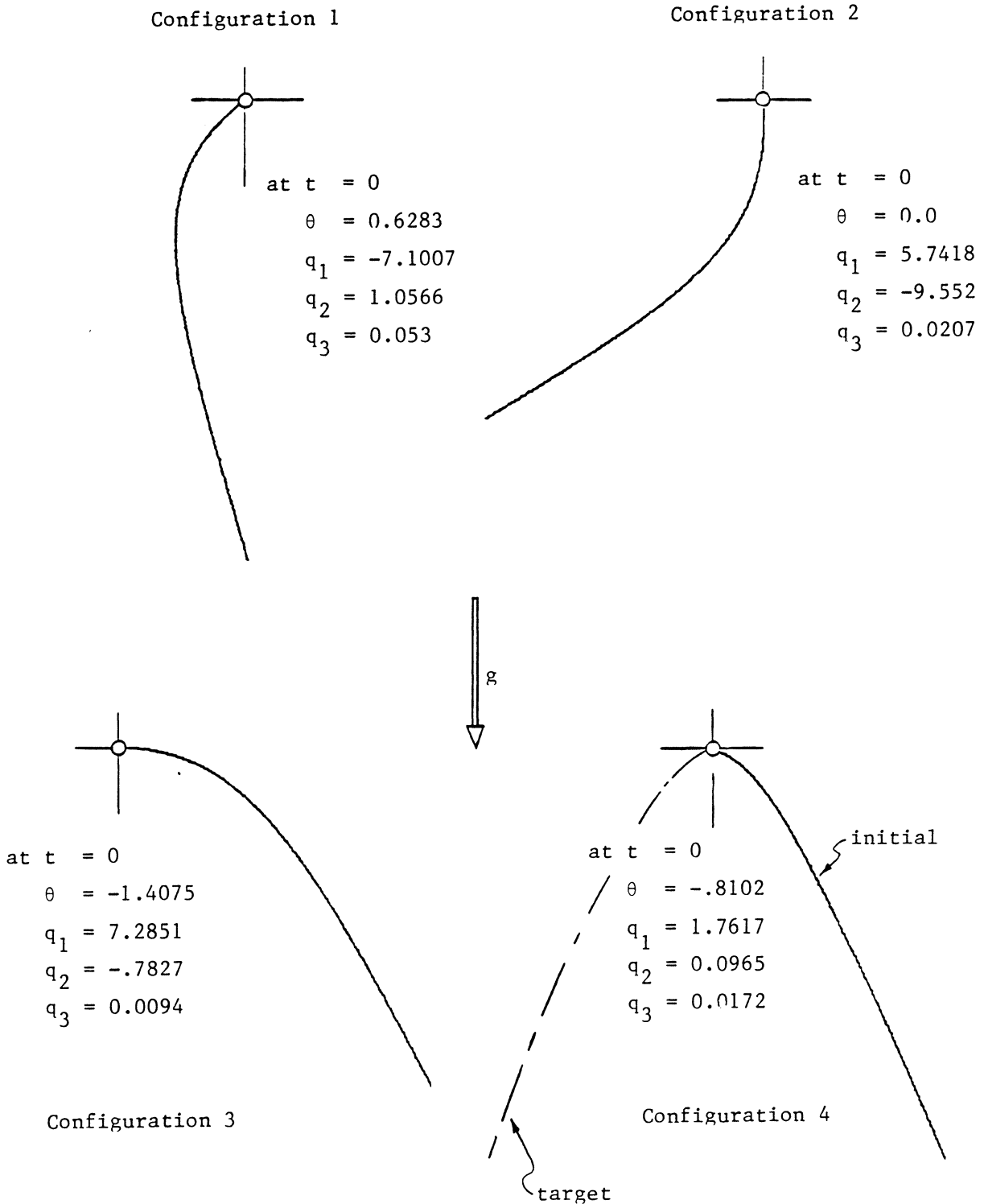


Figure 7.2 Initial configurations for experimental tests

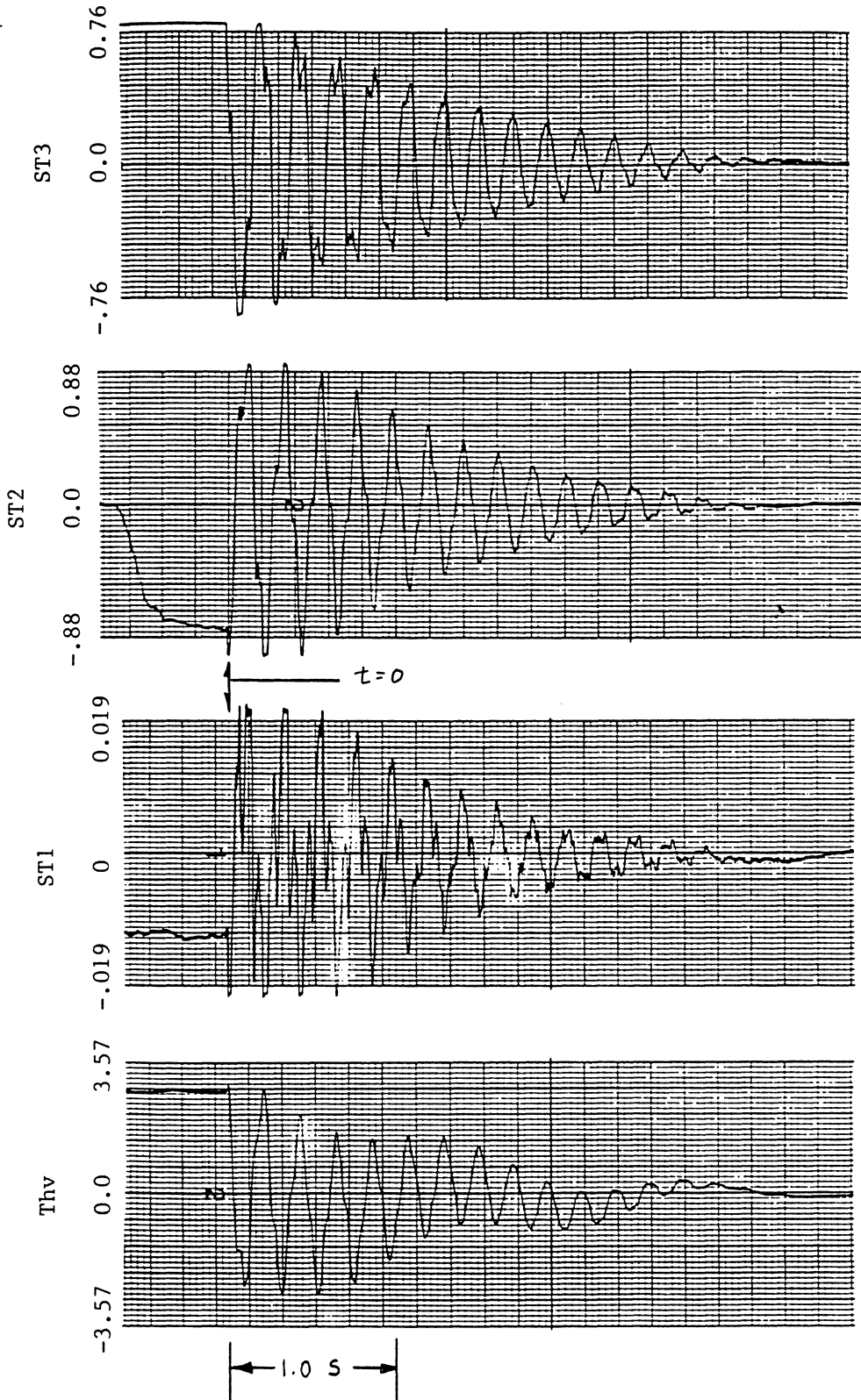


Figure 7.3 Uncontrolled-system response, initial configuration 1

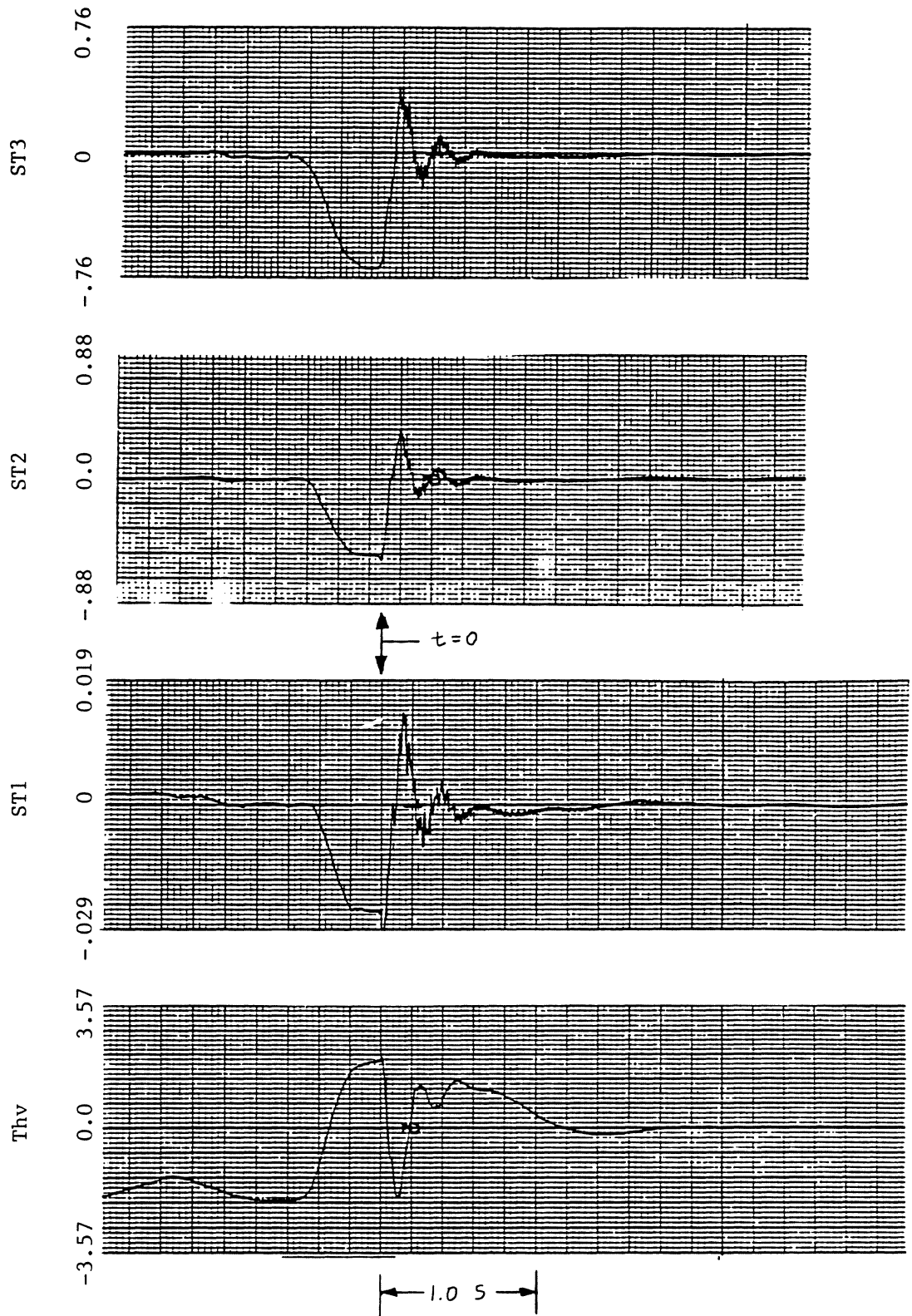


Figure 7.4 Experimental results, experiment number 1

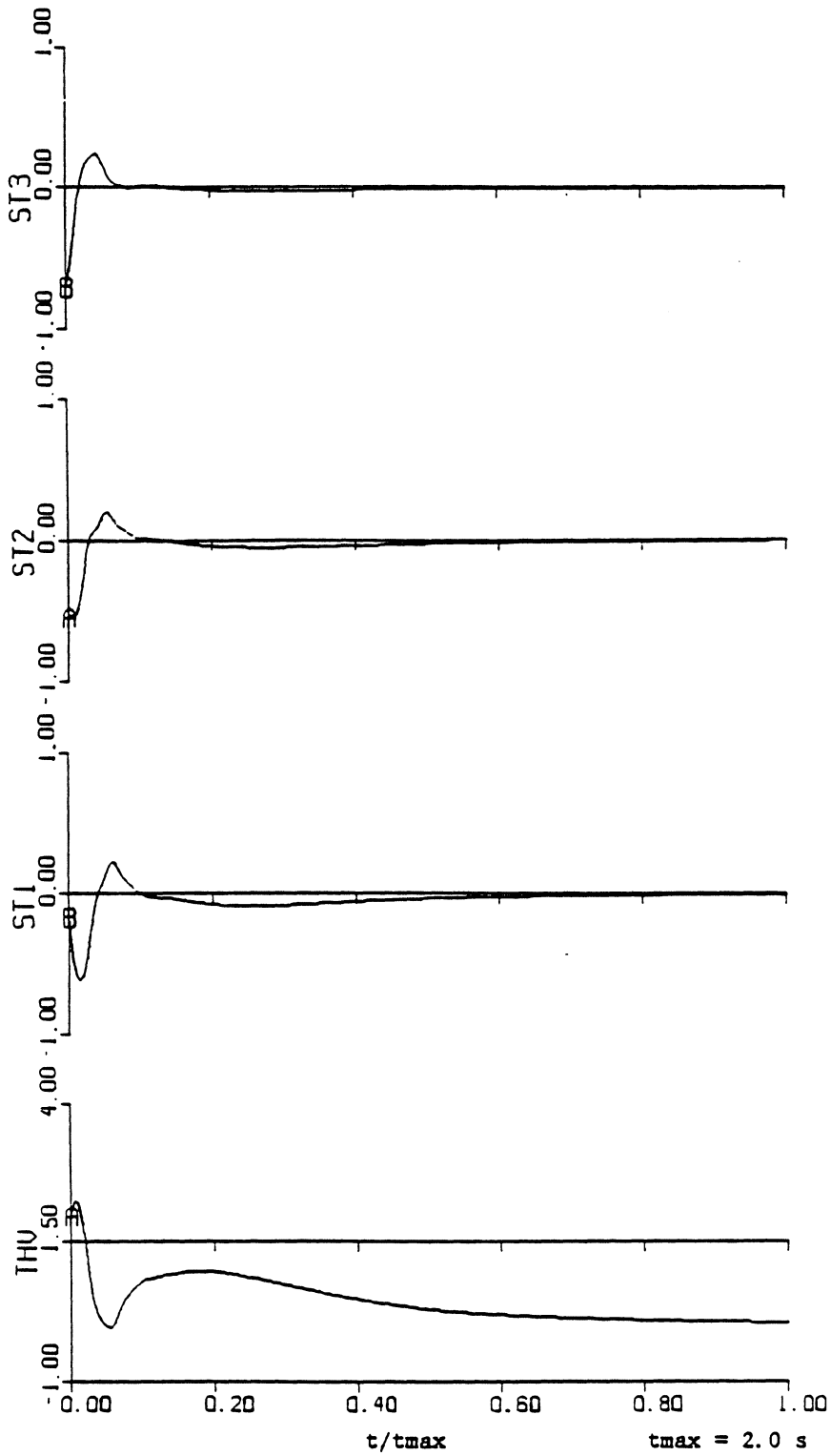


Figure 7.5 Simulation results, experiment number 1

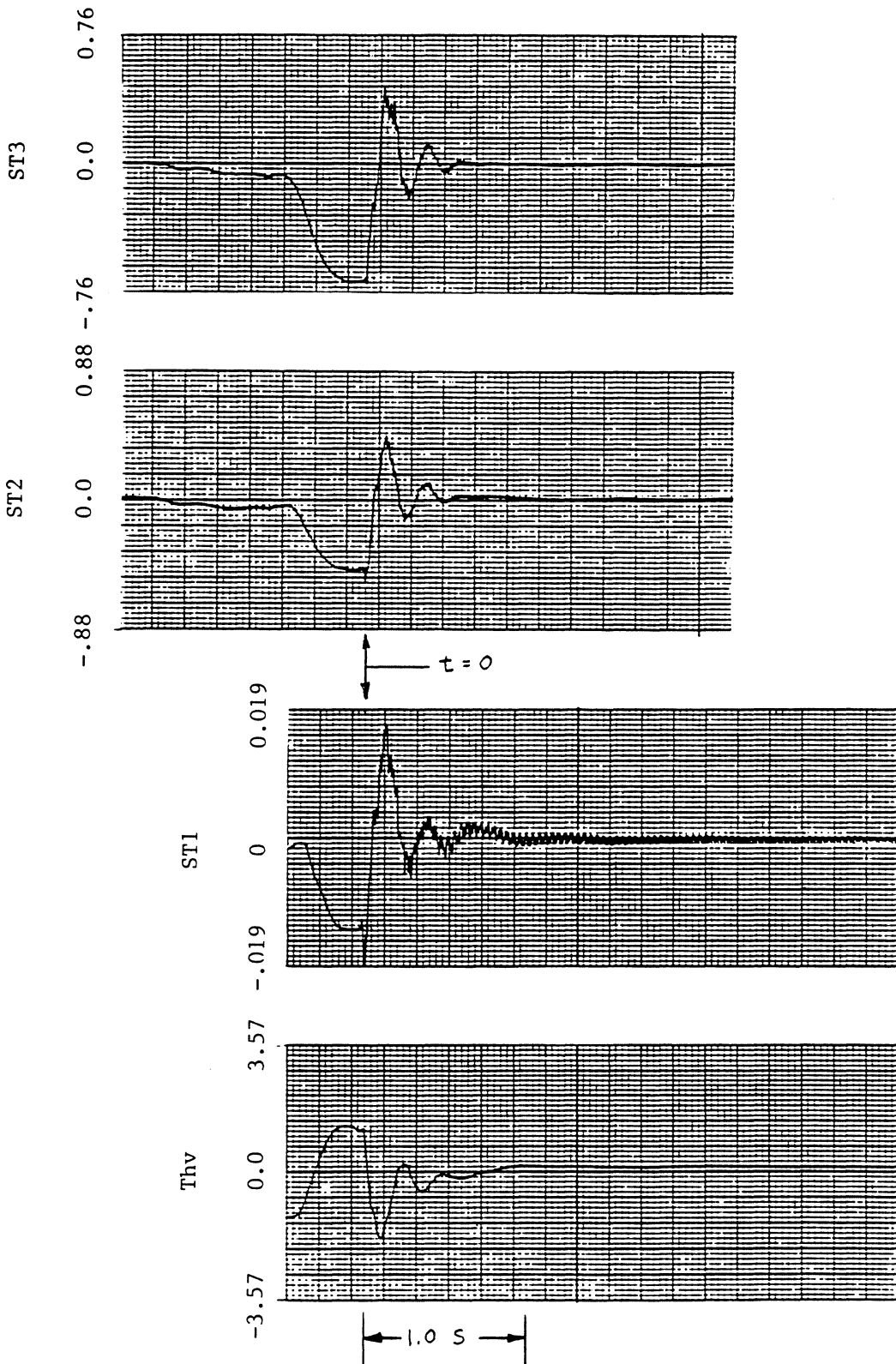


Figure 7.6 Experimental results, experiment number 2

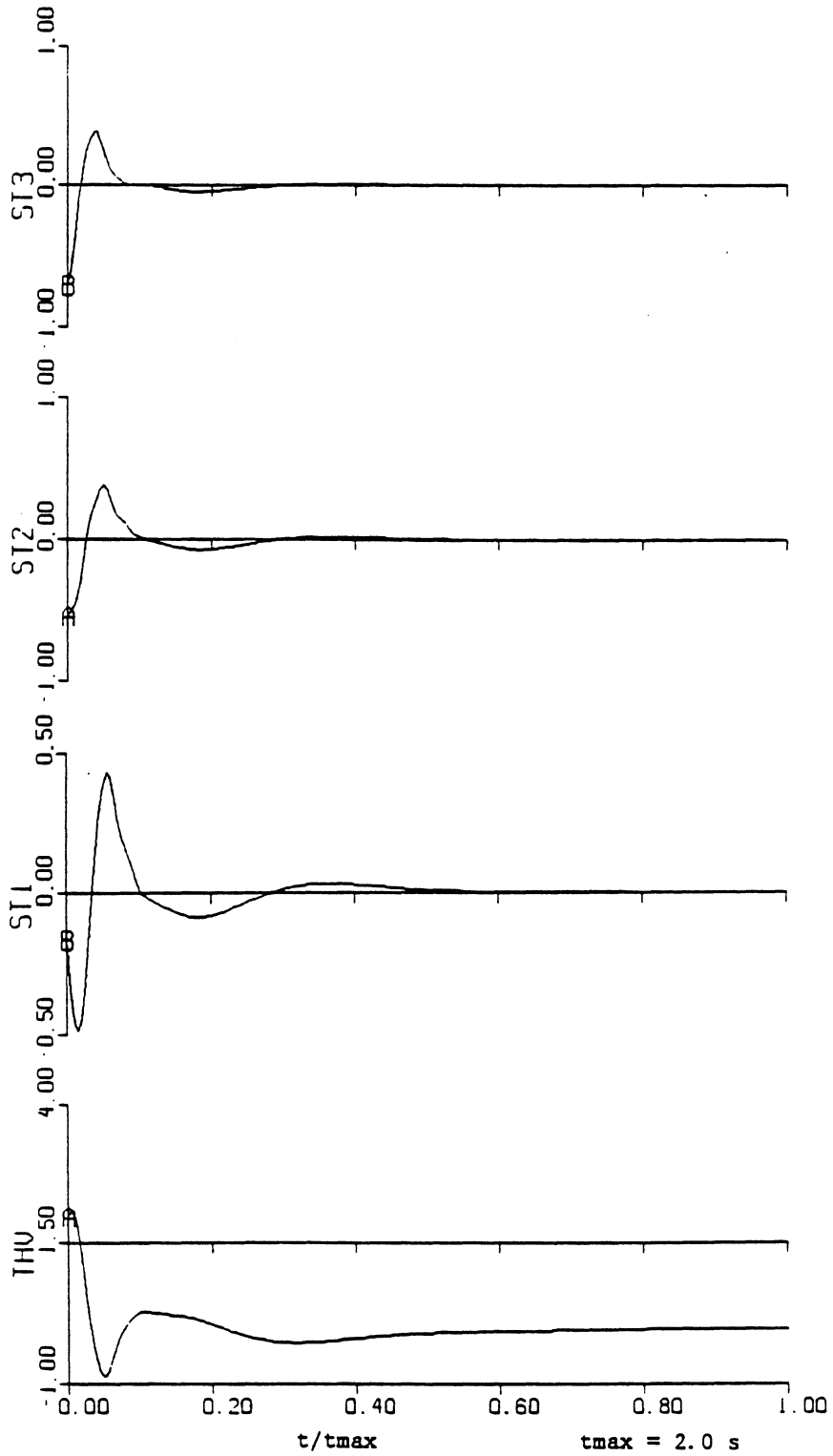


Figure 7.7 Simulation results, experiment number 2

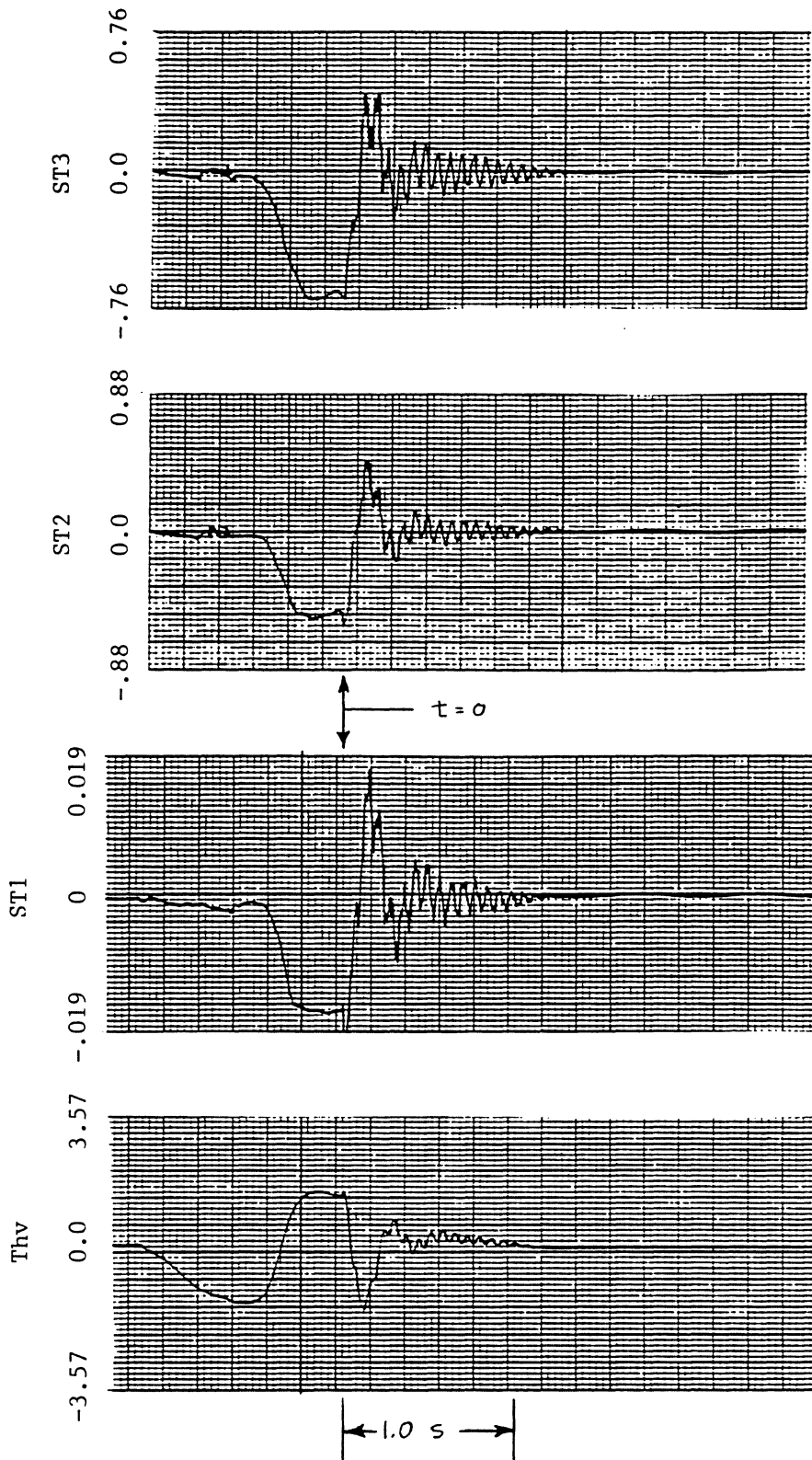


Figure 7.8 Experimental results, experiment number 3

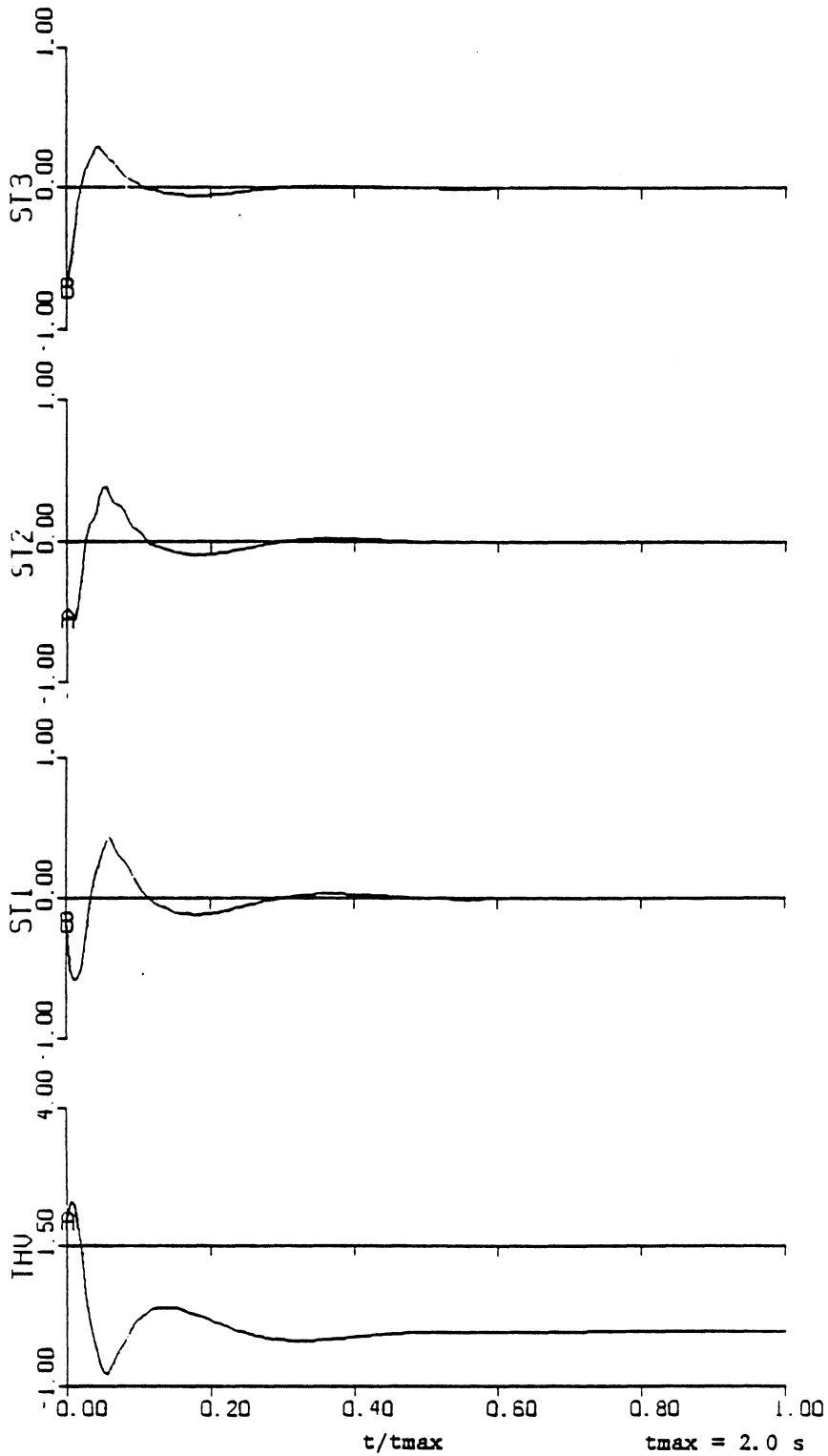


Figure 7.9 Simulation results, experiment number 3

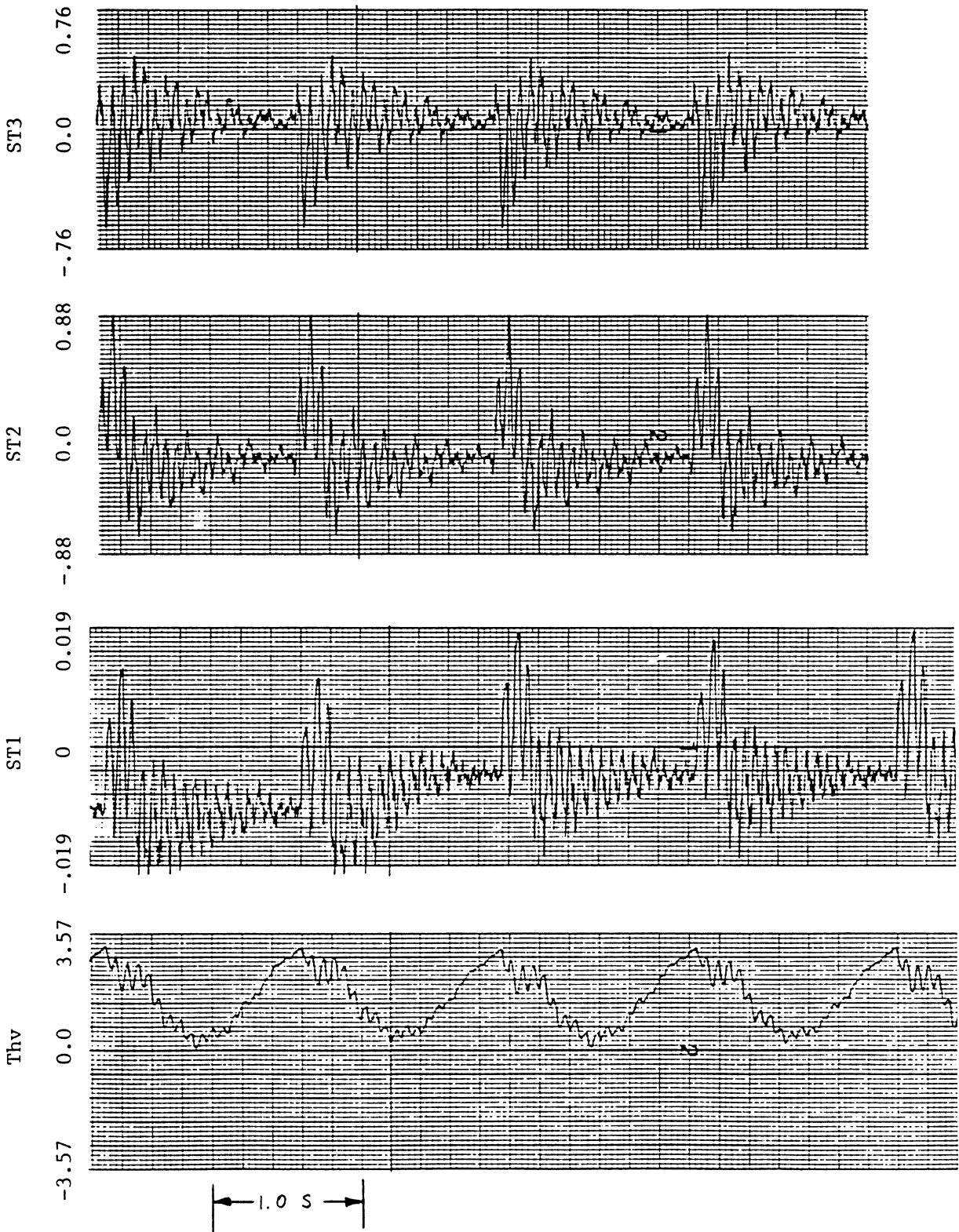


Figure 7.10 Experimental results, experiment number 4

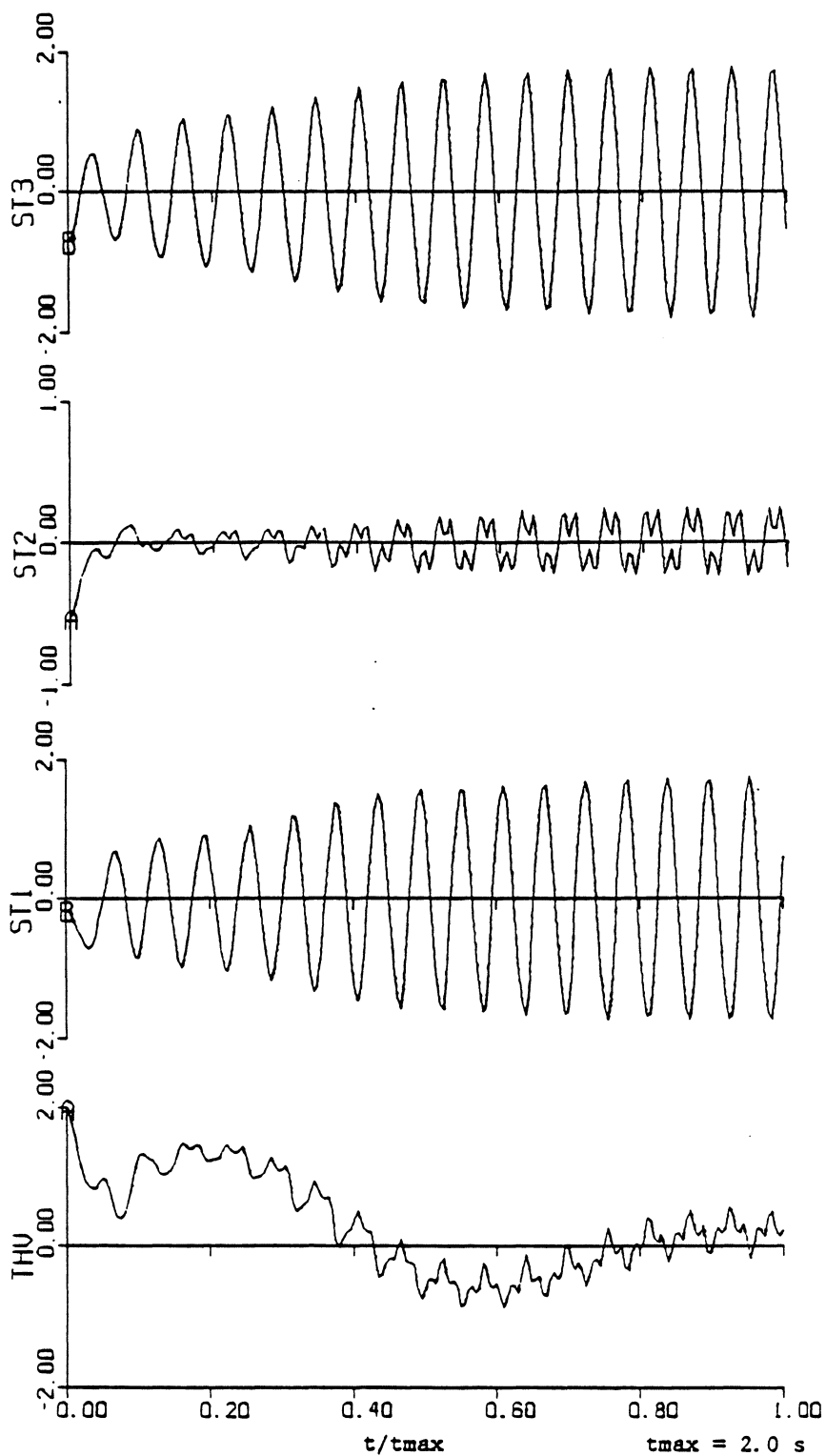


Figure 7.11 Simulation results, experiment number 4

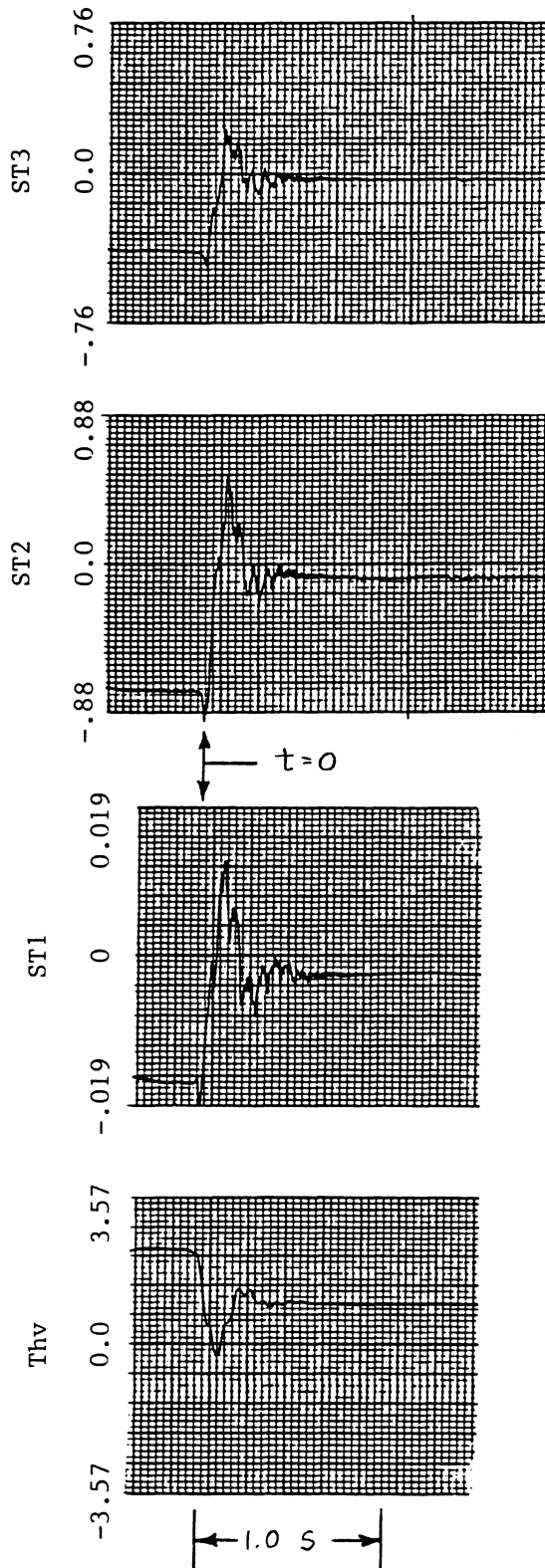


Figure 7.12 Experimental results, experiment number 5

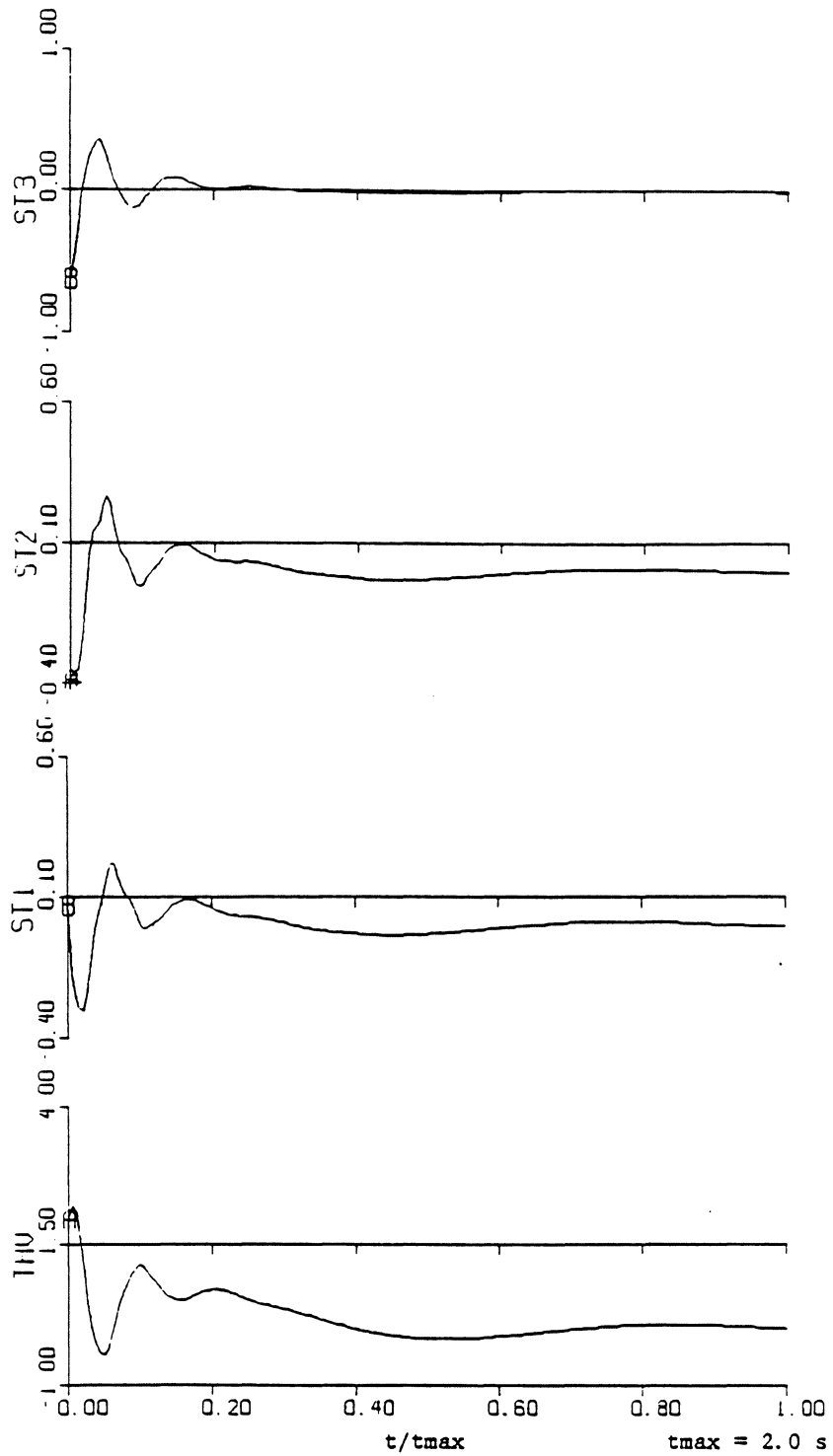


Figure 7.13 Simulation results, experiment number 5

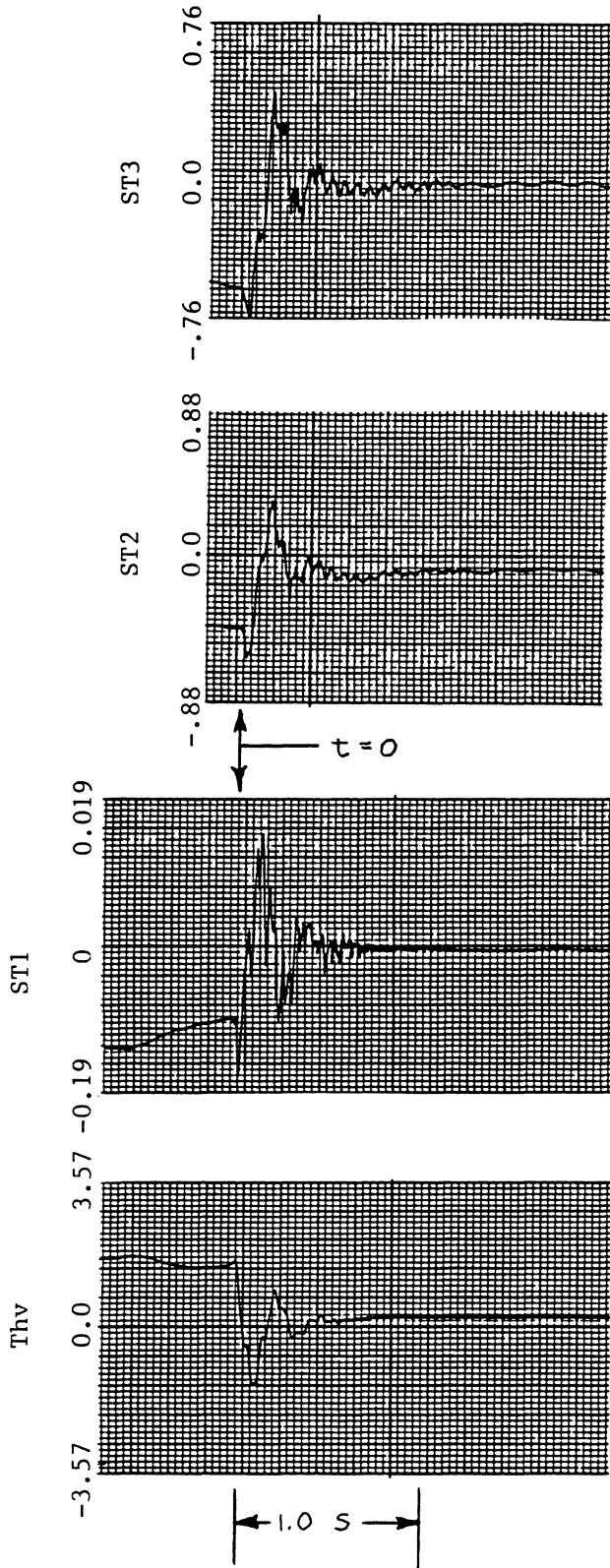


Figure 7.14 Experimental results, experiment number 6

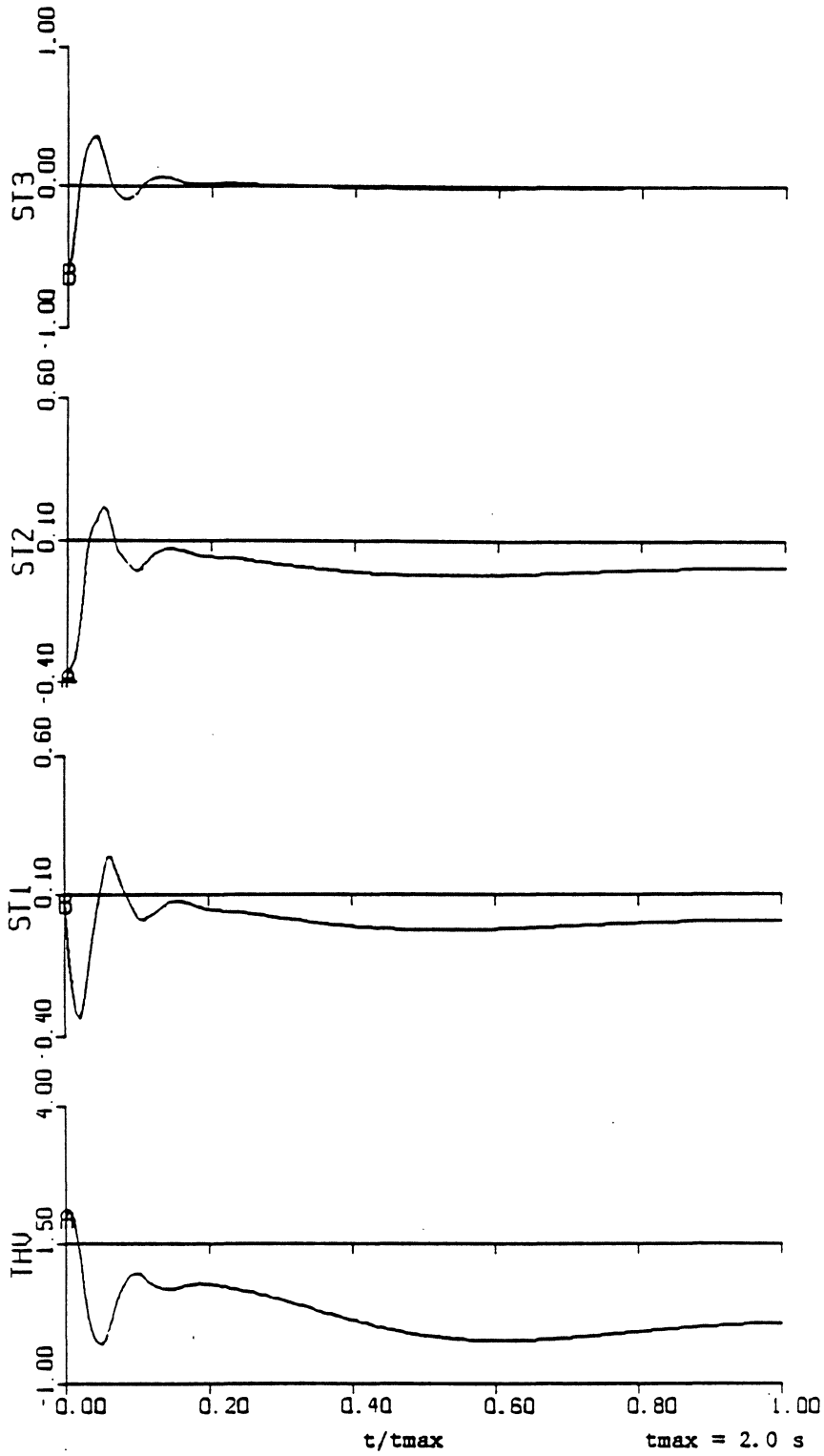


Figure 7.15 Simulation results, experiment number 6

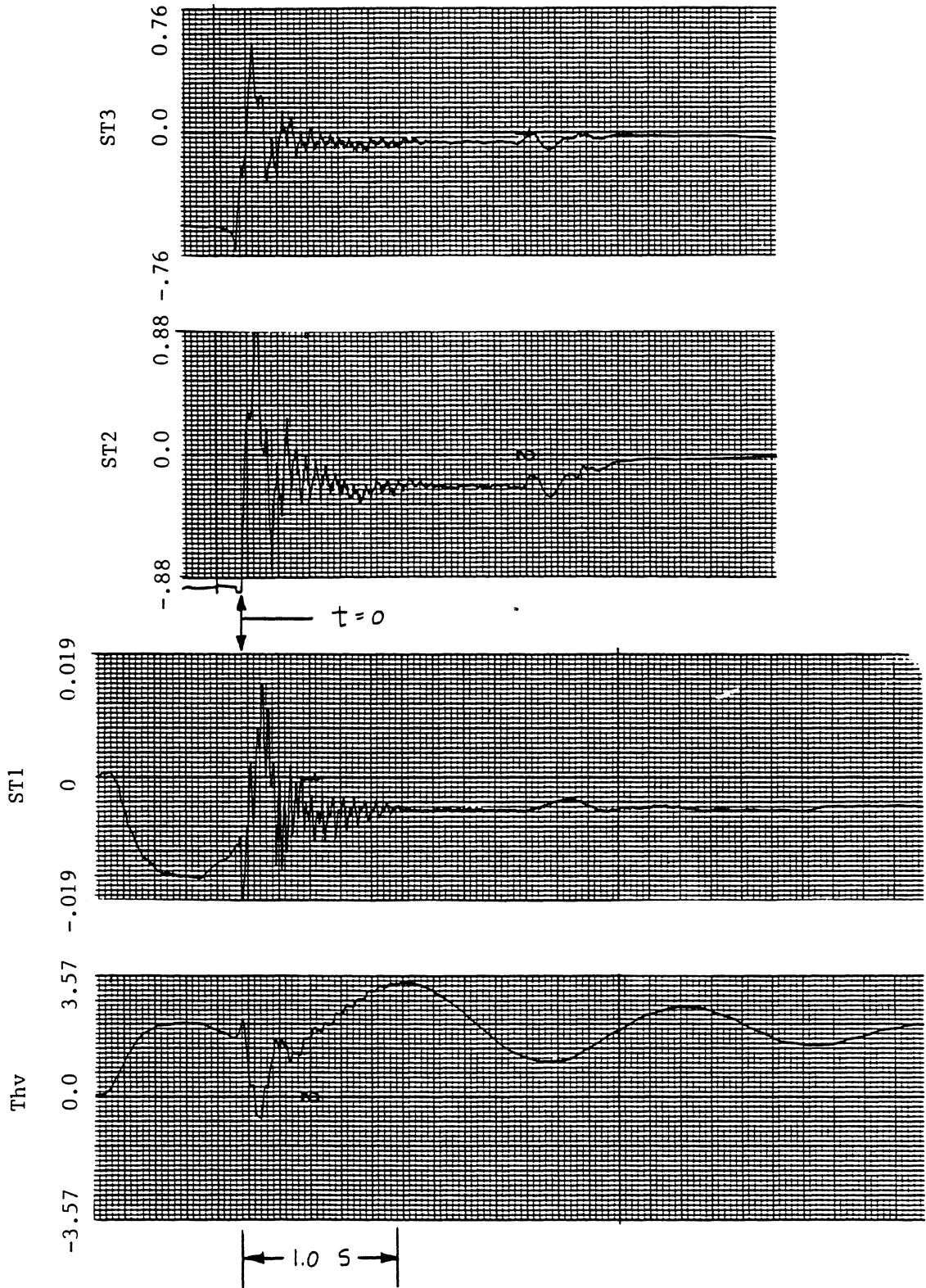


Figure 7.16 Experimental results, experiment number 7

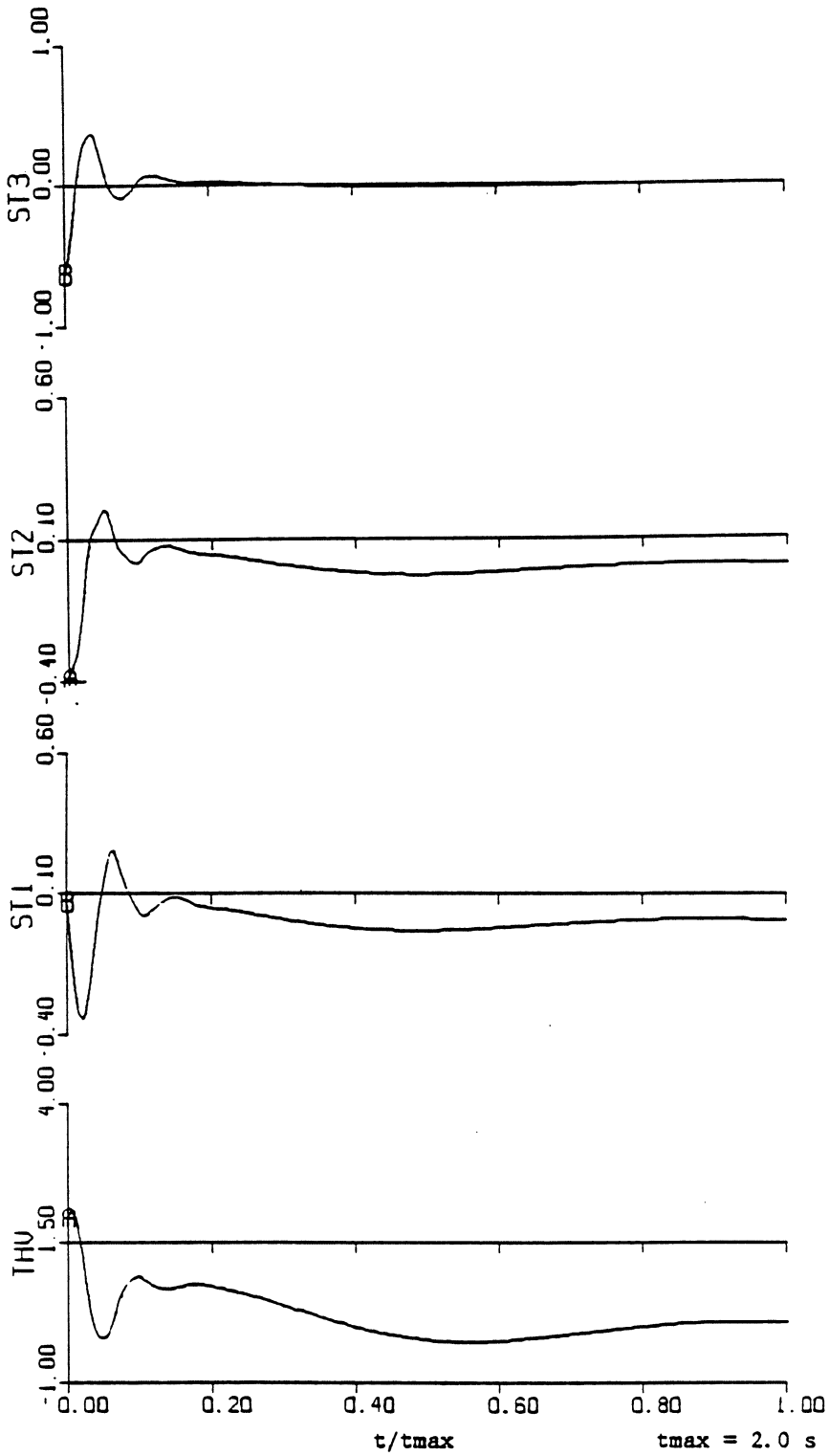


Figure 7.17 Simulation results, experiment number 7

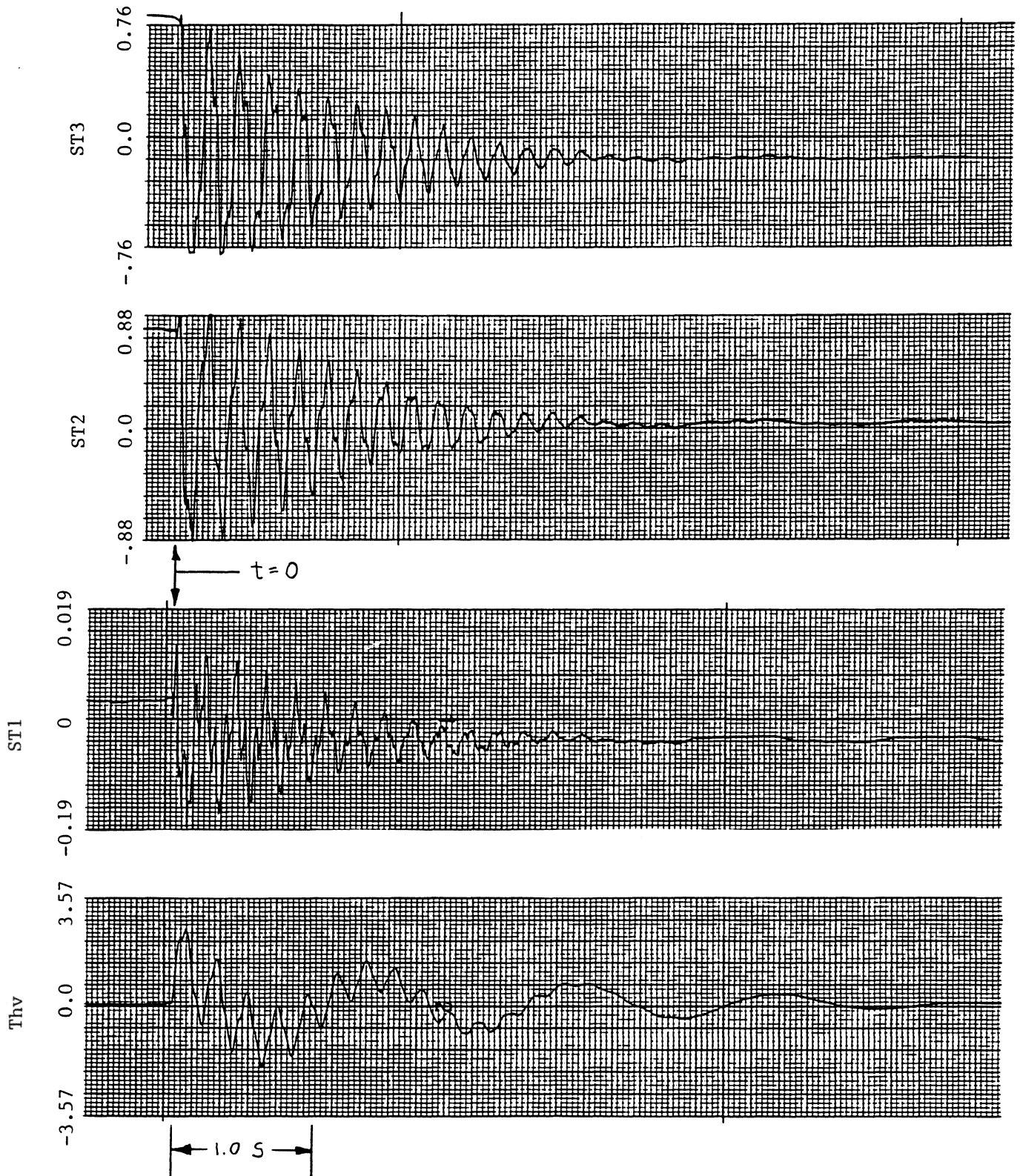


Figure 7.18 Uncontrolled-system response, initial configuration 2

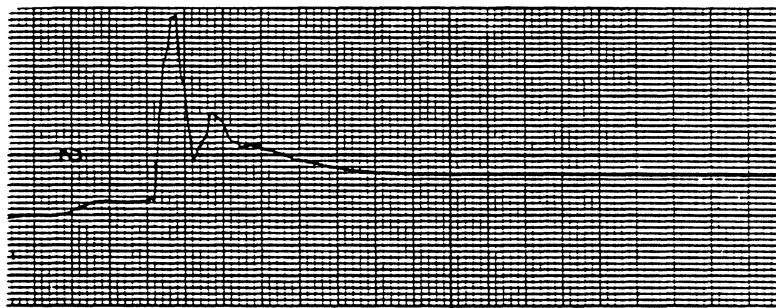
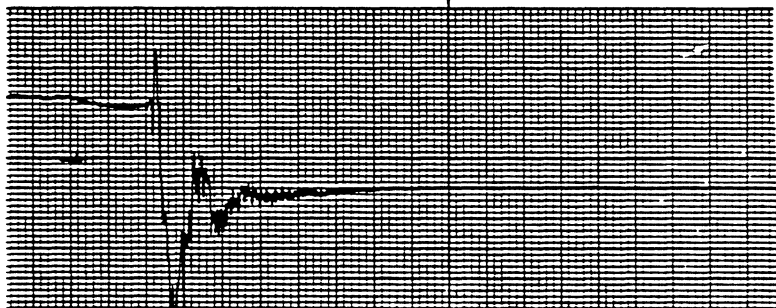
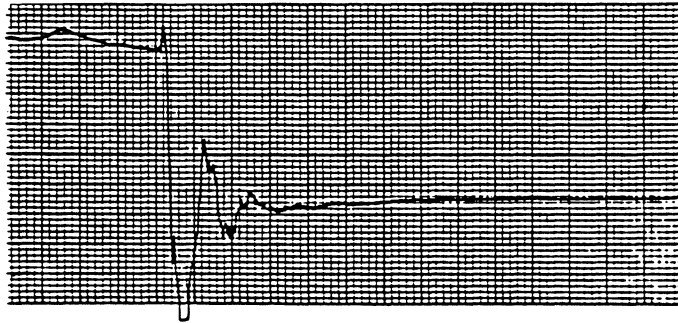
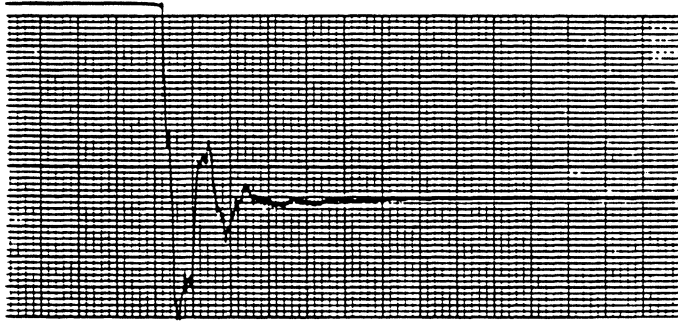


Figure 7.19 Experimental results, experiment number 8

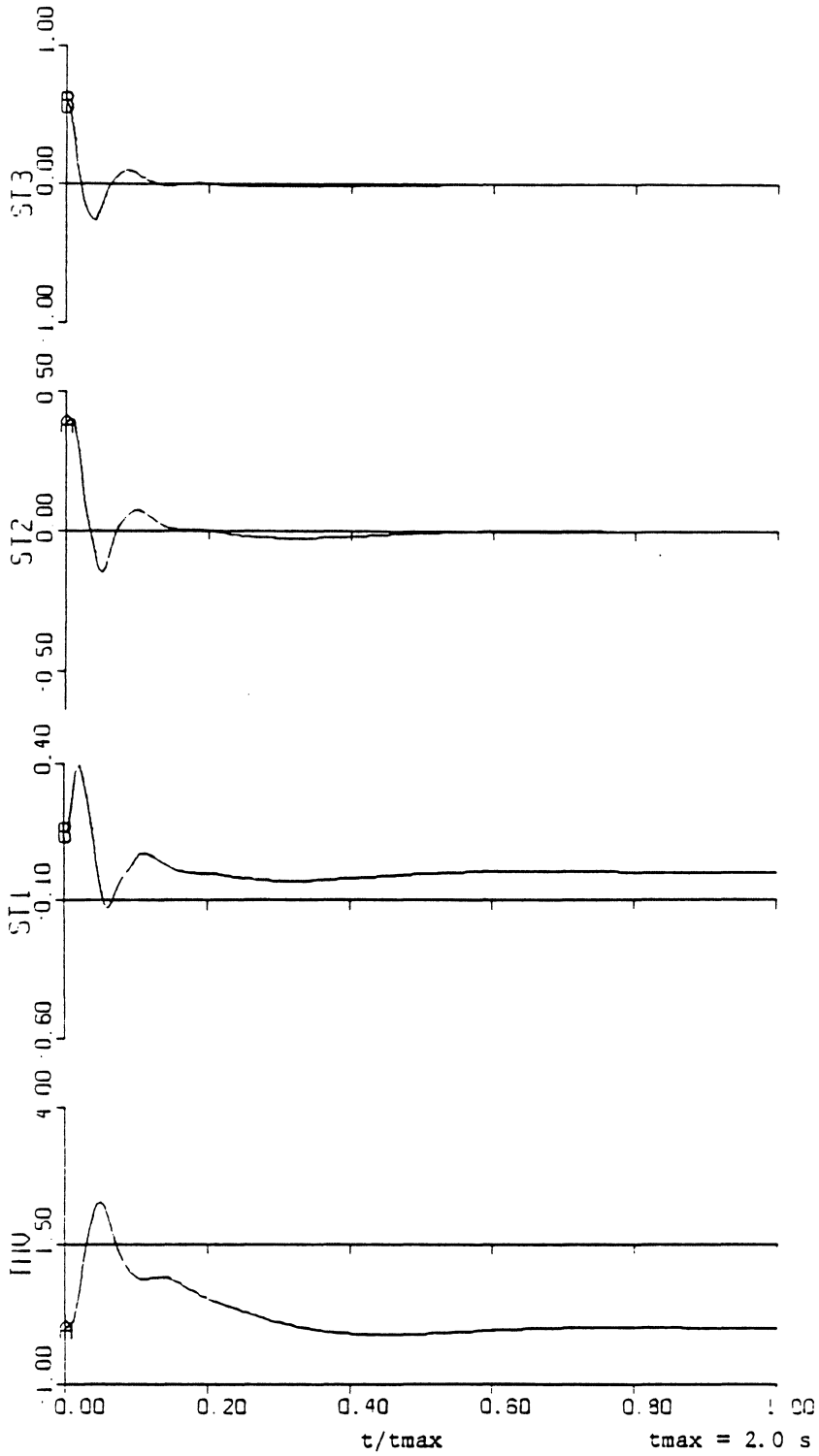


Figure 7.20 Simulation results, experiment number 8

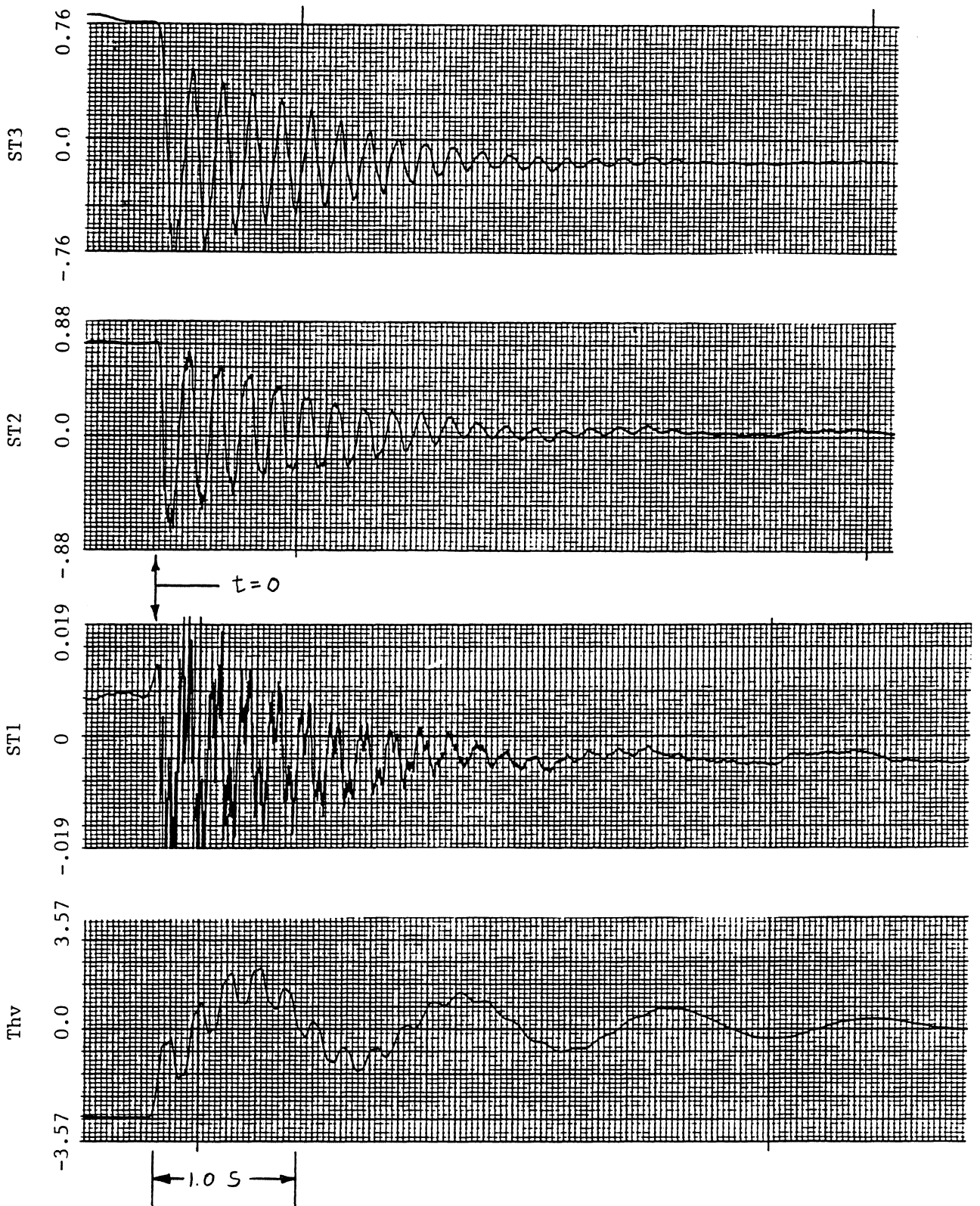


Figure 7.21 Uncontrolled-system response, initial configuration 3

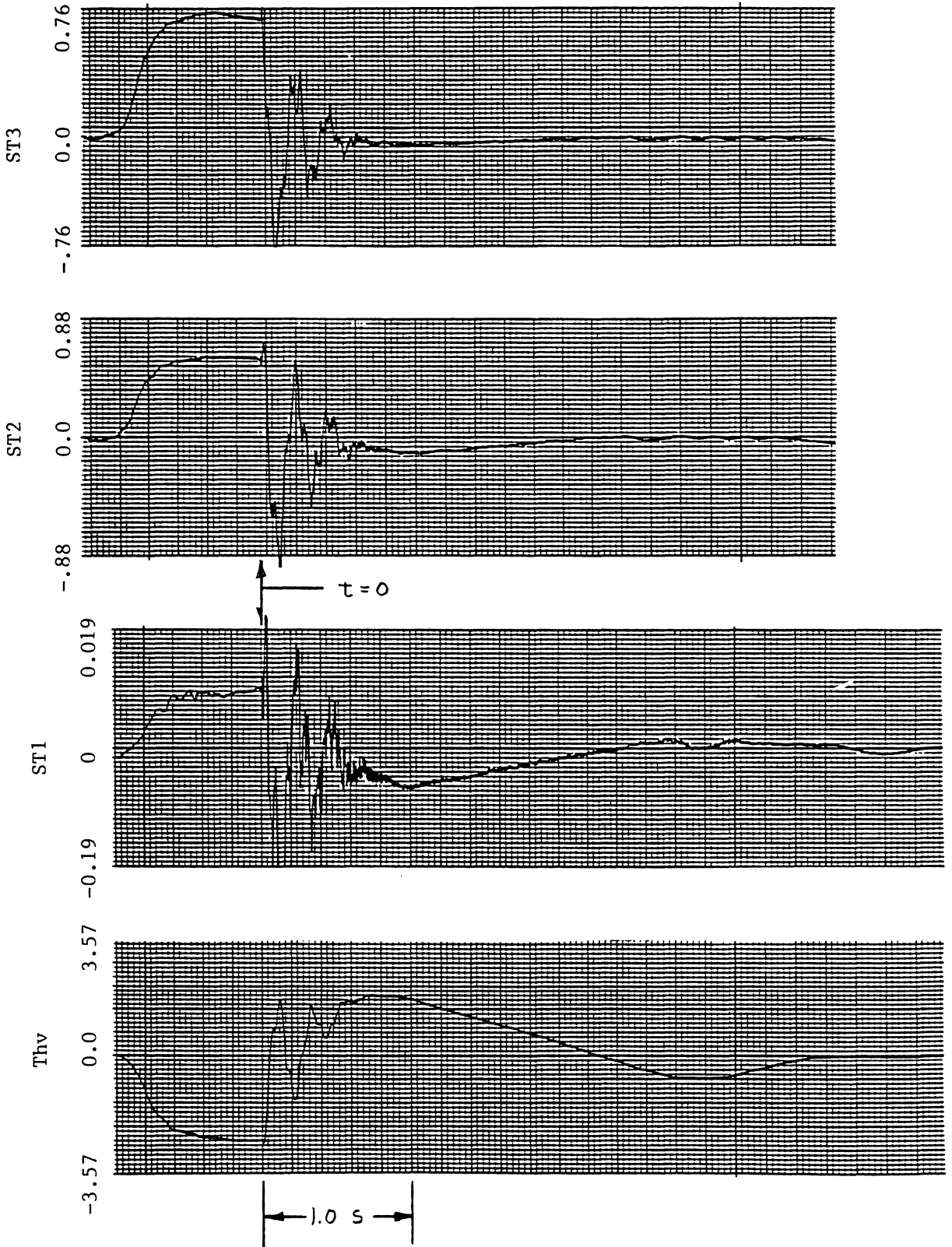


Figure 7.22 Experimental results, experiment number 9

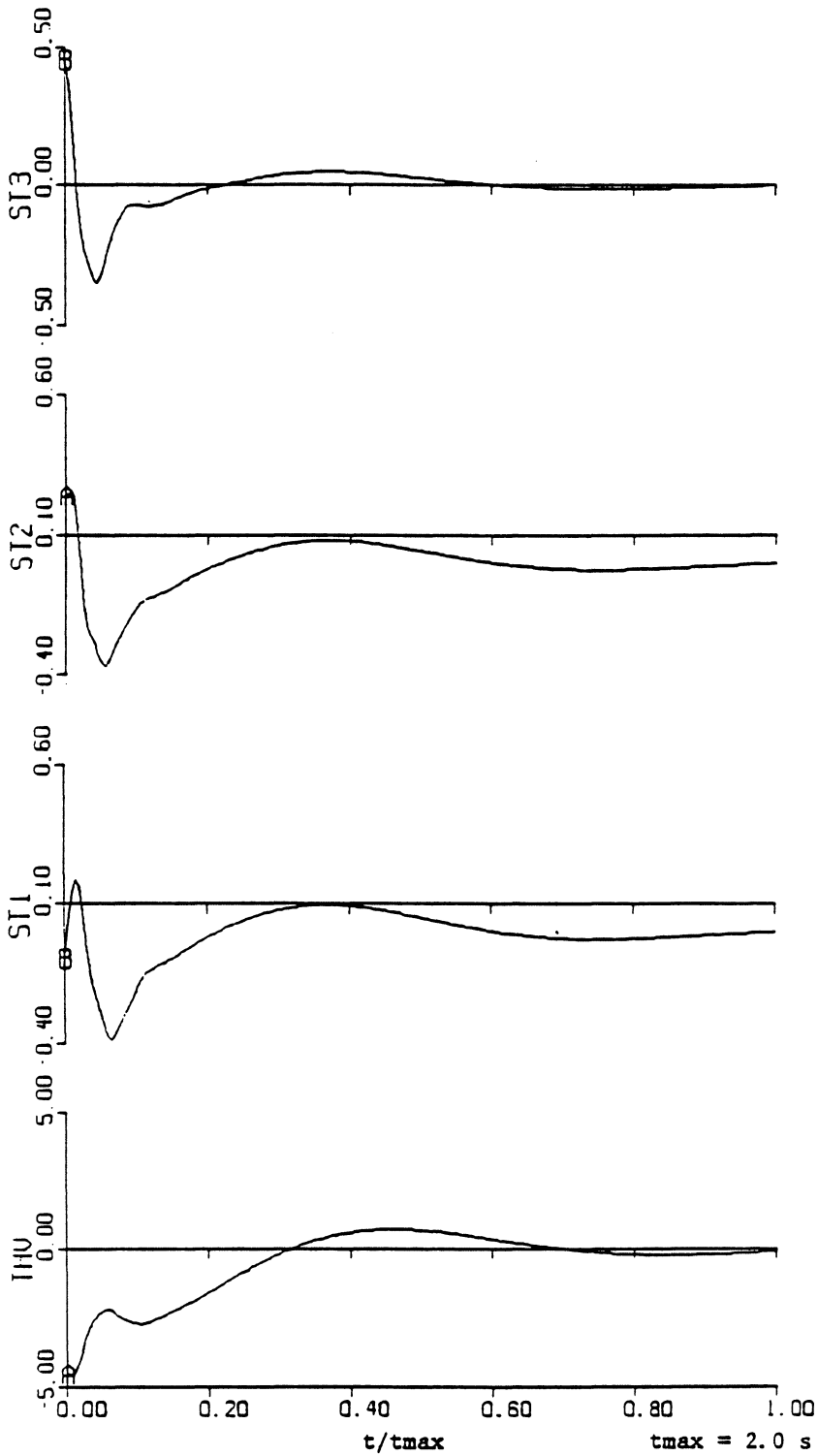


Figure 7.23 Simulation results, experiment number 9

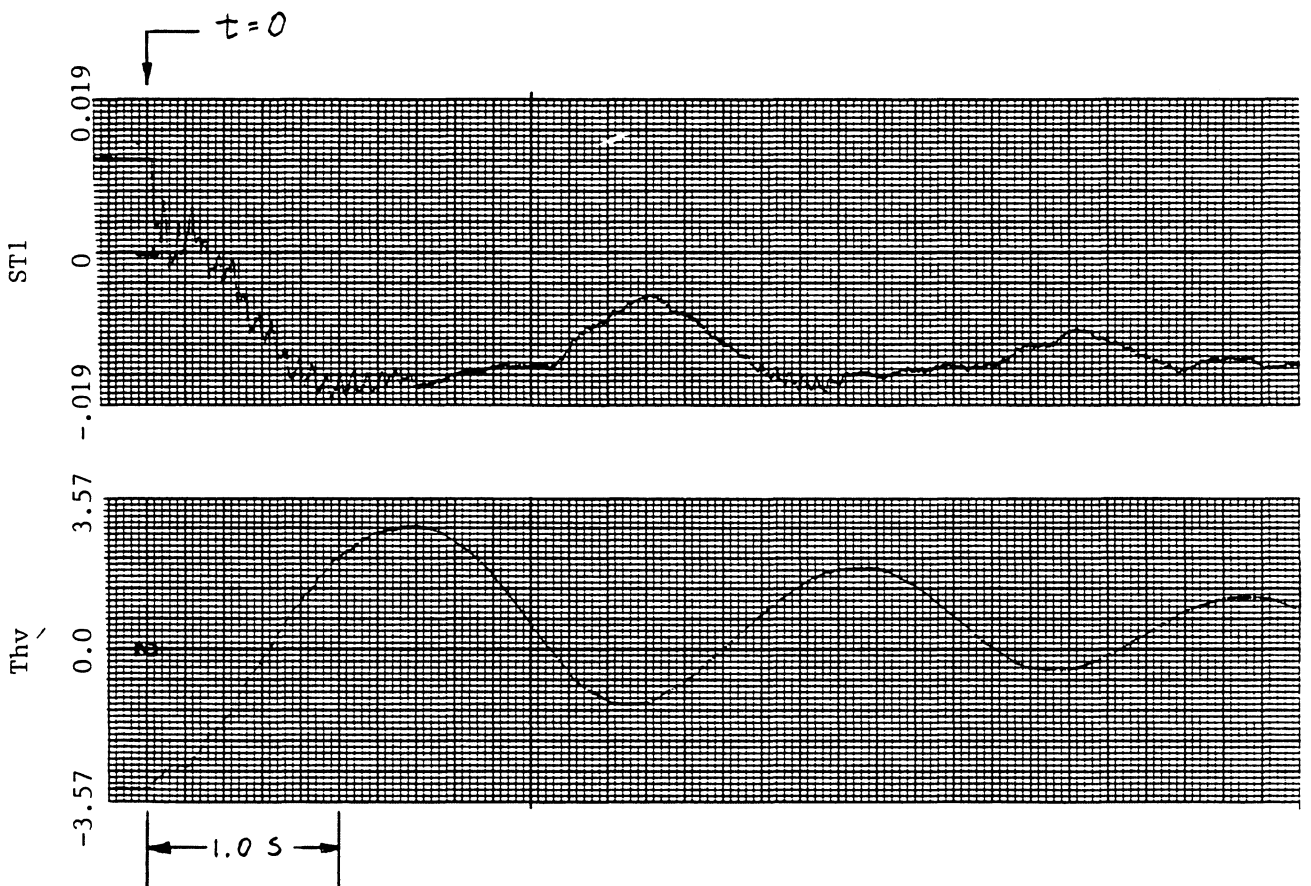


Figure 7.24 Slewing maneuver, $k = 1$, for comparison with results of experiment number 10

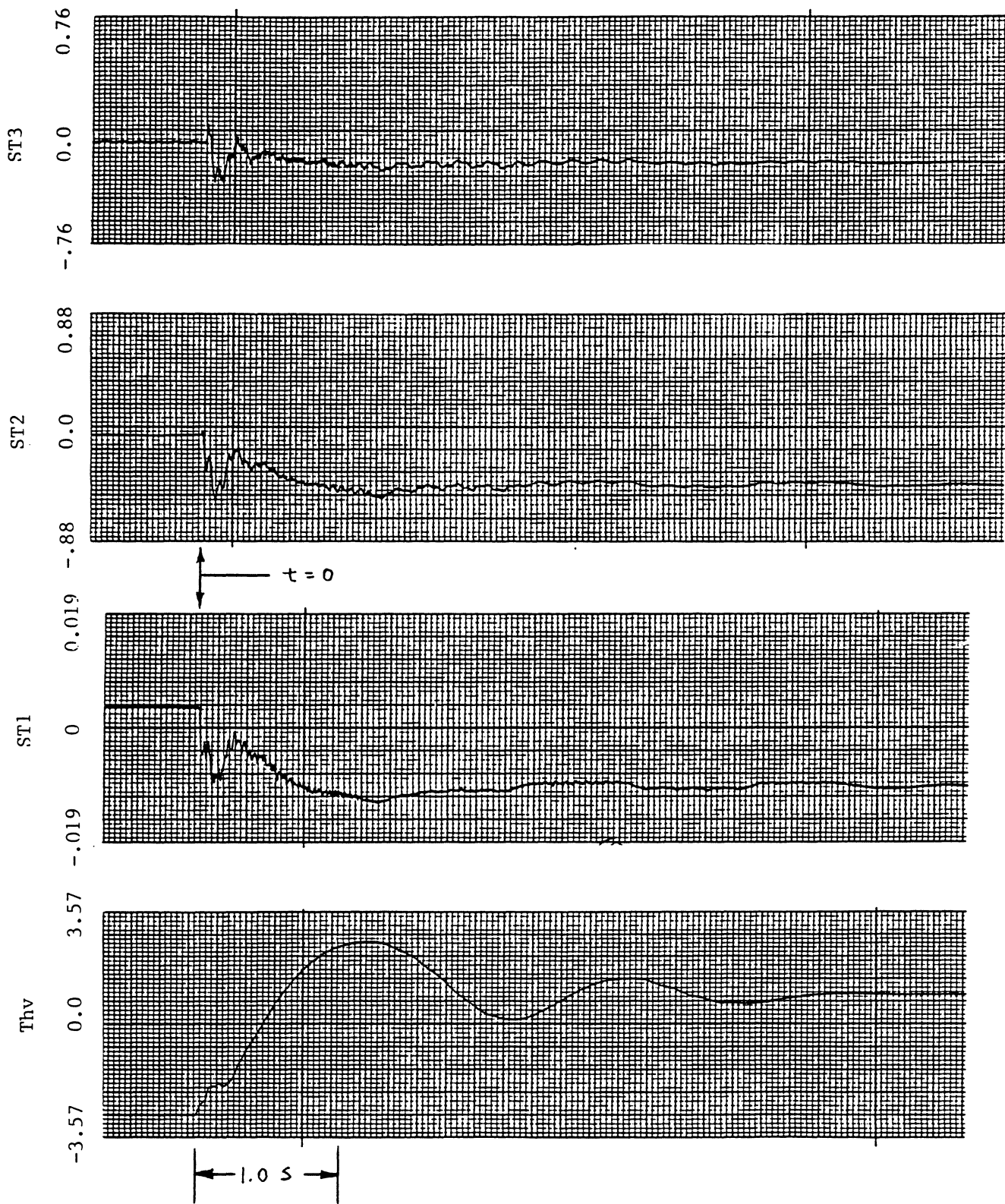


Figure 7.25 Experimental results, experiment number 10

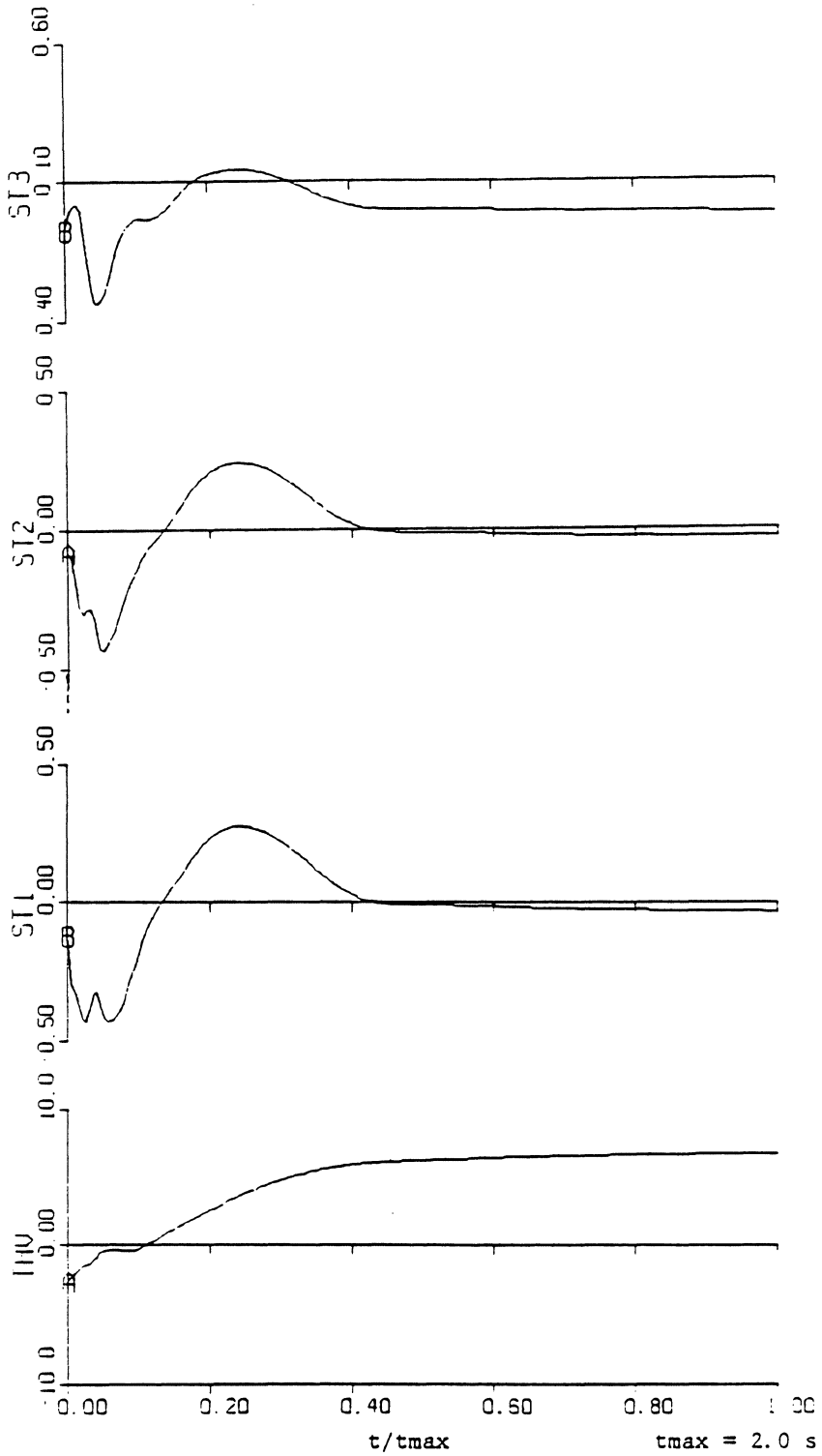


Figure 7.26 Simulation results, experiment number 10

CHAPTER 8 SUMMARY AND DISCUSSION

The equations of motion for the flexible beam system depicted in Fig. 2.1 were derived using Lagrange's energy-balance method. The equations were then linearized using two different schemes; one involved linearization about a sequence of discrete angles θ , the other involved direct cancellation of nonlinear terms through torque commands. Linear output feedback was added to the system. Gains were chosen so as to force the system to adopt prespecified eigenvalues, following a method discussed by Brogan [7]. It was desired to have these eigenvalues in the left-half region of the s-plane, and the number of eigenvalues which could be purposely and accurately located could be no greater than the number of independent outputs being monitored. Fortran-based computer programs were written for gain determination and for simulation of controlled-system response. The experimental system was then tested using the calculated gains; an IBM PC capable of A/D conversion served as the control device.

Results are respectable, although more so for vibration suppression than position control. Gains forcing the system-model to exhibit negative eigenvalues do, in general, lead to stable response of the experimental system. Correspondingly, gains relating to positive eigenvalues cause unstable experimental response. For gains chosen to push already-negative eigenvalues ever further to the left (s-plane), transient response-time decreases as it should. However, if pushed too far - past some indeterminate boundary - the response may begin to deteriorate. Three reasons are suggested for this.

1. The frequency of actual beam vibration may become higher than

that associated with the beam's third natural mode (this occurred on occasion). The control algorithms assume that beam shape can be described as a linear combination of the first three beam mode shapes, and if this is not true the algorithm's effectiveness decreases.

2. A digital computer is used as the control device. It monitors feedback signals and outputs motor (torque) commands at discrete time-intervals. The equations used for gain determination, however, were written for an entirely analog system. It is possible, of course, to recognize the sampled-data nature of the controller when formulating the control expression. If the sample rate is relatively fast as compared to system response, the assumption of a continuous-time controller is adequate. Pushing eigenvalues to the left increases system response speed...perhaps too much for our assumption to remain valid.
3. Pushing eigenvalues to the left usually entails large gains. As gains increase, the voltage output by the controller increases. The controller is limited, however, to ± 10 volts as indicated in Fig. 8.1. If the necessary voltage is outside of this range, the motor will not receive the proper commands. The ST1 oscillation seen in Fig. 7.6 is too mild to have been caused by this. However, wild and uncontrollable motion was induced on rare occasions, and the IBM PC output characteristics are thought to be responsible. The occasions were rare because overall system gain k_{sys} was set equal to the

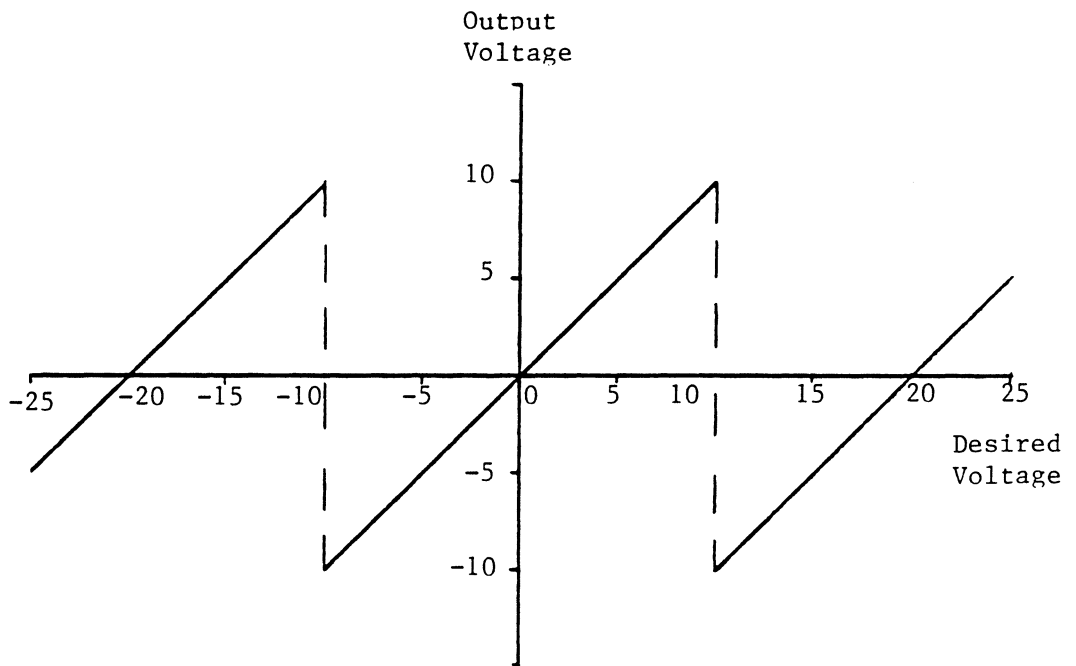


Figure 8.1 Characteristics of controller output voltage

relatively small value of 0.21.

These problems are unimportant if acceptable response can be achieved with eigenvalues which are not "too negative". The results presented in chapter 6 would indicate that this is the case.

In the experiments detailed in chapter 6, beam motion was confined to approximately $-70^\circ < \theta < 70^\circ$. Admittedly, this is quite a restriction. The motor was unable to supply enough continuous torque to hold the beam at angles beyond these. Upon realizing this, plans were initially made to boost this torque through a gearing system. This entails additional nonlinearities due to backlash, though, and it was thought best to concentrate on the current problem before increasing its complexity. Eventually, gears will be needed.

All things considered, this work has been a success. Extrapolation to larger systems composed of more than just one flexible beam looks promising. The dynamic equations will become correspondingly large, but the matrix approach of modern control theory is well-suited to this. The finite element method is based upon a foundation similar to the work presented here, and might be used to advantage in the search for appropriate control gains for these large, linearized systems.

REFERENCES

1. Balas, M. J., "Trends in Large Space Structure Control Theory: Fondest Hopes, Wildest Dreams," IEEE Transactions on Automatic Control, Vol. AC-27, No. 3, pp. 522-535, June 1982.
2. Meirovitch, L. (ed.), Proceedings of the 1st VPI&SU/AIAA Symposium on Dynamics and Control of Large Flexible Spacecraft, Blacksburg, VA, June 1977.
3. Meirovitch, L. (ed.), Proceedings of the 2nd VPI&SU/AIAA Symposium on Dynamics and Control of Large Flexible Spacecraft, Blacksburg, VA, June 1979.
4. Meirovitch, L. (ed.), Proceedings of the 3rd VPI&SU/AIAA Symposium on Dynamics and Control of Large Flexible Spacecraft, Blacksburg, VA, June 1981.
5. Meirovitch, L. (ed.), Proceedings of the 4th VPI&SU/AIAA Symposium on Dynamics and Control of Large Flexible Spacecraft, Blacksburg, VA, June 1983.
6. Meirovitch, L. (ed.), Proceedings of the 5th VPI&SU/AIAA Symposium on Dynamics and Control of Large Flexible Spacecraft, Blacksburg, VA, June 1985.
7. Brogan, W. L., Modern Control Theory, Prentice-Hall, Inc., Englewood Cliffs, NY, 1985.
8. Mitchell and Gauthier, Associates, Advanced Continuous Simulation Language, User Guide/Reference Manual, Concord, MA, 1981.

General

9. Patten, W. N., "Transient Response Characteristics of a Construction Crawler Crane--A Variational Approach," M.S. Thesis, Department of Mechanical Engineering, VPI&SU, Blacksburg, VA, May 1980.
10. Juston, J. M., "Theoretical and Experimental Study into the Dynamics and Control of a Flexible Beam with a DC-Servo Motor Actuator," M.S. Thesis, Department of Mechanical Engineering, VPI&SU, Blacksburg, VA, October 1985.
11. Merovitch, L., Methods of ANALYTICAL DYNAMICS, McGraw-Hill Book Company, 1970.
12. Holman, J. P., Experimental Methods for Engineers, McGraw-Hill Book Company, 1984.
13. Hale, F. J., Introduction to Control System Analysis and Design, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1973.

APPENDIX A

Gain-Determination Programs

1. FFLEX: controlling program
2. FLXRED: reads data files
3. KGETA: computes θ -dependent stiffness matrix
4. PEEPC2: calculates eigenvalues
5. KORDR1: re-orders gains
6. SUBRS: curves are fit to gains
7. AGETC: Legendre-polynomial coefficients are transformed into coefficients of the polynomials $f_k = \theta^{k-1}$ for IBM PC usage
8. GAINS: gains at discrete θ are determined using curve-fit equations
9. MATINV: matrix inversion


```

COMPLEX*8 A(20,20),AS2(20,20),B(20),BS2(20),M(20,20),MS2(20,20)
COMPLEX*8 LAMBDA(20),EIG(20),ASYS(20,20),Z(20,20),AS3(20,20)
COMPLEX*8 A11(20,20),WK(100),PR(20,20),BT(20,20)
COMPLEX*8 A12(20,20),T2(20,20),TINV(20,20),T1(20,20)

REAL*8 AR(20,20),AR2(20,20),BR(20),BR2(20),MR(20,20),MR2(20,20)
REAL*8 A1(20),A2(20),A3(20),A4(20),A5(20),A6(20)
REAL*8 A7(20),A8(20),K(20),ASYSR(20,20),KOUT(20)
INTEGER NPP,NEV,ICNT,INT,VM(20)

C
C INITILAIIZE GAIN-RELATED VECTORS
C
2 DATA K/20*0./,A1/20*0./,A2/20*0./,A3/20*0./,A4/20*0./
DATA A5/20*0./,A6/20*0./,A7/20*0./,A8/20*0./,KOUT/20*0./
C
C READ IN PERTINENT VARIABLES, MATRICES & VECTORS WHICH DETERMINE
C WHAT IS BEING SOLVED FOR
C
CALL REDFLX(A,B,M,T1,AS2,BS2,MS2,T2,NPP,NEV,LAMBDA,
$THA,THB,INT,ICURV,NPOLY,NVM,VM)
C
C BEGIN GAIN-DETERMINATION LOOP
C THIS INVOLVES TWO SUBROUTINES: 1. KGETA 2. PEEPC2
C GAINS ARE WRITTEN FROM PEEPC2 TO THE FILE 'FLEX OUT'
C
THINCR=(THB-THA)/FLOAT(INT)
DO 100 ICNT=1,INT
TH=THA+FLOAT(ICNT)*THINCR
CALL KGETA(NEV,TH,AS2,MS2,A11)
CALL PEEPC2(NEV,NVM,VM,LAMBDA,A11,BS2,TH,K,T1)
100 CONTINUE
C
C END GAIN DETERMINATION LOOP
C REWIND 'FLEX OUT'
C
REWIND(UNIT=16)
C
C THESE ABOVE-COMPUTED GAINS ARE NOW APPLIED TO A DYNAMIC SYSTEM OF
C SPECIFIED ORDER. THE GAINS ARE EITHER USED AS THEY APPEAR IN
C SGNRAD OUT, OR THESE GAINS ARE FIT TO CURVES--THIS DEPENDING UPON
C USER-SPECIFIED PARAMETER ICURV
C KORDRI REARRANGES GAINS SO THAT EACH GAIN IS COUPLES WITH THE
C APPROPRIATE STATE. OUTPUT WRITTEN TO FILE 'SGNRAD OUT'(UNIT 11)
C
CALL KORDRI(NVM,VM,NEV,INT)
REWIND(UNIT=11)
C
C DETERMINE WHETHER CURVE IS TO BE FIT
C IF SO, SUBRS FITS CURVE, AGETC CONVERTS RESULTS TO FORM SUITABLE
C FOR FASTEST POSSIBLE CONTROLLER IMPLEMENTATION
C
IF (ICURV.EQ.0) GOTO 200
CALL SUBRS(NEV,NPOLY,INT,A1,A2,A3,A4,A5,A6,A7,A8)
CALL AGETC

DO 212 I=1,8
READ(12,*) A1(I),A2(I),A3(I),A4(I)
READ(12,*) A5(I),A6(I),A7(I),A8(I)
212 CONTINUE
C
C BEGIN LOOP OF EIGENVALUE DETERMINATION
C GAINS ARE EITHER READ DIRECTLY FROM 'SGNRAD OUT' OR CALCULATED
C USING CURV-FIT COEFFICIENTS, AS APPROPRIATE
C
200 DO 300 ICNTR=1,INT
IF (ICURV.EQ.1) GOTO 210
READ(11,*) TH,(K(I),I=1,4)
READ(11,*) (K(I),I=5,8)
READ(11,*) (K(I),I=9,12)

```

```

      GOTO 225
210   TH=THA+FLOAT(ICNTR)*THINCR
211   FORMAT(1X,8(F7.3,1X))
      CALL GAINS(TH,K,A1,A2,A3,A4,A5,A6,A7,A8)
221   FORMAT(4(F9.4,2X))
225   CALL KGETA(NEV,TH,AS2,MS2,AS3)
C
C DETERMINE MATRIX ASYS=A2-B2*K*T, USED FOR EIGENVALUE DETERMINATION
C
      DO 240 I=1,NEV
        DO 230 J=1,NEV
          BT(I,J)=BS2(I)*K(J)
230   CONTINUE
231   FORMAT(1X,6(F10.4,1X))
240   CONTINUE

      CALL MULT(NEV,BT,T2,PR)

      DO 250 I=1,NEV
        DO 260 J=1,NEV
          ASYS(I,J)=AS3(I,J)-PR(I,J)
260   CONTINUE
250   CONTINUE
C
C DEFINE CERTAIN VARIABLES NEEDED BY IMSL SUBROUTINE 'EIGCC'
C
      MM=20
      MM2=20
      MI=0
C
C CALCULATE EIGENVALUES
C
      CALL EIGCC(ASYS,NEV,MM,MI,EIG,Z,MM2,WK,IER)

      WRITE(10,7) TH,(EIG(I),I=1,4)
      WRITE(10,6) (EIG(I),I=5,8)
      WRITE(10,6) (EIG(I),I=9,12)
      WRITE(10,*) ' '
6     FORMAT(7X,4(F7.2,' ',' ',F7.2))
7     FORMAT(F7.4,4(F7.2,' ',' ',F7.2))
300   CONTINUE

      STOP
      END

```



```

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
*
* 'FLXRED FORTRAN' IS ONE PROGRAM AMONGST A GROUP WHICH TOGETHER
* SOLVE FOR THE GAINS ASSOCIATED WITH A LINEAR FEEDBACK CONTROL
* SYSTEM.
*
* 'FLXRED FORTRAN' IS RESPONSIBLE FOR ALL COMMUNICATION WITH
* DATA FILES.
*
* THE GROUP IS CAPABLE OF PERFORMING A VARIETY OF RELATED-BUT-
* DIFFERENT CALCULATIONS ASSOCIATED WITH CONTROL SYSTEM DESIGN,
* AS DETAILED IN THE PRELIMINARY STATEMENTS INCLUDED IN PROGRAM
* 'FLEX FORTRAN'. THE DIRECTIVES WHICH DICTATE WHAT IS TO BE
* SOLVED FOR RESIDE IN DATA FILE 'FLEX DATA'. THEREFORE,
* 'FLXRED FORTRAN' FIRST READS THIS INFORMATION AND THEN GRABS
* THE APPROPRIATE SYSTEM DATA FILES.
*
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

```

```

SUBROUTINE FLXRED(A,B,M,T1,AS2,BS2,MS2,T2,NPP,NEV,LAMBDA,
$THA,THB,INT,ICURV,NPOLY,NVM,VM)

```

```

COMPLEX*8 A(20,20),AS2(20,20),B(20),BS2(20),M(20,20),MS2(20,20)
COMPLEX*8 LAMBDA(20),EIG(20),ASYS(20,20),Z(20),AS3(20,20),XXX
COMPLEX*8 A11(20,20),T(20,20),T2(20,20),TINV(20,20),T1(20,20)

```

```

REAL*8 AR(20,20),AR2(20,20),BR(20),BR2(20),MR(20,20),MR2(20,20)
REAL*8 A1(20),A2(20),A3(20),A4(20),A5(20),A6(20)
REAL*8 A7(20),A8(20),K(20),SG,T2R(20,20),TR(20,20)

```

```

INTEGER VM(20)

```

```

C
C INITILAIIZE GAIN-RELATED VECTORS
C

```

```

DATA K/20*0./,A1/20*0./,A2/20*0./,A3/20*0./,A4/20*0./
DATA A5/20*0./,A6/20*0./,A7/20*0./,A8/20*0./

```

```

C
C READ IN THE VARIOUS PARAMETERS WHICH DICTATE WHAT PARTICULAR SOLUTION
C IS DESIRED. THIS INFORMATION IS PLACED IN A DATA FILE RATHER THAN
C BEING INPUT INTERACTIVELY BECAUSE MUCH OF THE DATA INEVITABLY REMAINS
C THE SAME FROM RUN TO RUN, AND TO RETYPE THIS INFORMATION EACH TIME
C IS UNDESIREABLE.
C

```

```

C THIS PROGRAM IS WRITTEN SPECIFICALLY FOR THE THESIS-RELATED SYSTEM.
C USING DIFFERENT SIZED MODELS TO REPRESENT THE TRUE SYSTEM EFFECT-
C IVELY REFERS TO THE NUMBER OF MODES INCLUDED IN THE APPROXIMATION
C FOR BEAM LATERAL DISPLACEMENT: Y(X,T)
C

```

```

C THE DIFFERENT SYSTEM DATA FILES, THEREFORE, ARE MOST CONVENIENTLY
C DISTINGUISHED BY THE NUMBER OF MODES OF THE APPROXIMATION.
C

```

```

C READ THE NUMBER OF MODES ASSOCIATED WITH THE SYSTEM USED FOR
C FINDING GAINS.
C

```

```

READ(9,*) NPG
NPP=2*NPG+2

```

```

C
C ASSUMING THAT INCOMPLETE FEEDBACK IS USED, IT IS NECESSARY TO
C KNOW WHICH STATES ARE BEING MONITORED. LET TH=1,ST1=2,ST2=3,ST3=4,
C THD=5,ST1D=6,ST2D=7,ST3D=8. READ THE NUMBER OF VARIABLES BEING
C MEASURED, AND THEN THE NUMBERS CORRESPONDING TO THESE MEASUREMENTS,
C IN INCREASING SEQUENTIAL ORDER.
C

```

```

READ(9,*) NVM
READ(9,*) (VM(I),I=1,NVM)

```

```

C INPUT THE OVER-ALL SYSTEM GAIN
C

```

```

READ(9,*) SG

```

```

C
C INPUT THE COMPLEX EIGENVALUES THAT THE SYSTEM SHOULD BE FORCED
C TO ADOPT (SPECIFY NPP VALUES).
C
      IF (NVM.EQ.1.OR.NVM.EQ.3.OR.NVM.EQ.5) GOTO 4
      READ(9,*) (LAMBDA(I),I=1,NVM/2)
      READ(9,*) (LAMBDA(I),I=NVM/2+1,NVM)
      GOTO 5
4     READ(9,*) (LAMBDA(I),I=1,NVM/2+1)
      IF (NVM.NE.1) GOTO 6
      READ(9,*) XXX
      GOTO 5
6     READ(9,*) (LAMBDA(I),I=NVM/2+2,NVM)
C
C INPUT THE RANGE OF BEAM ANGLE IN RADIANS OVER WHICH THE SOLUTION IS
C TO BE DETERMINED (THA,THB).
      NOTE OF IMPORTANCE: CURRENTLY, 'FUNCTION FUNNC' WITHIN
      'SUBROUTINE SUBRS' ASSUMES THA=-6.282,
      THB=6.282. IF YOU PLAN TO CHANGE THIS
      RANGE, YOU MUST CHANGE THE CORRESPONDING
      RANGE IN FUNNC. THIS INFORMATION CANNOT
      BE PASSED TO FUNNC THROUGH ITS CALLING
      STATEMENT BECAUSE AN IMSL SUBROUTINE
      ALSO ACCESSES FUNNC, AND REQUIRES THE
      ARGUMENTS TO BE OF A PARTICULAR FORM.
5     READ(9,*) THA,THB
C
C INPUT THE NUMBER OF INTERVALS INTO WHICH THIS RANGE SHOULD BE
C SUBDIVIDED
      READ(9,*) INT
C
C SHOULD A CURVE-FIT BE APPLIED TO THE GAINS? (YES=1, NO=0)
      READ(9,*) ICURV
C
C HOW MANY POLYNOMIALS SHOULD BE USED FOR THESE CURVE-FITS?
      READ(9,*) NPOLY
C
C INPUT THE NUMBER OF MODES ASSOCIATED WITH THE SYSTEM (1,2,3) USED
C FOR EIGENVALUE DETERMINATION USING THE CALCULATED GAINS. THIS MUST
C BE LESS THAN OR EQUAL TO NPP
      READ(9,*) NEG
      NEV=2*NEG+2
*
* THIS MARKS THE END OF DIRECTIVES FROM 'FLEX DATA'
*
C READ THE APPROPRIATE SYSTEM DESCRIPTION FROM INDICATED DATA FILES
      FILEDEF 1=ONE-MODE SYSTEM
      FILEDEF 2=TWO-MODE SYSTEM
      FILEDEF 3=THREE-MODE SYSTEM
      FILEDEF 4=THREE-MODE SYSTEM + ESTIMATORS
C
C MATRIX A=SYSTEM MATRIX...DEVOID OF ANY TH-RELATED INFORMATION
C
      DO 10 I=1,NPP
        READ(NPG,*) (AR(I,J),J=1,NPP)
        DO 8 J=1,NPP
          A(I,J)=CMPLX(AR(I,J))
8         CONTINUE
10      CONTINUE
C
C VECTOR B PROVIDES HARDWARE-RELATED INFO
C
      READ(NPG,*) (BR(I),I=1,NPP)
      DO 11 I=1,NPP
        B(I)=CMPLX(BR(I))*SG

```

```

11      CONTINUE
C
C MATRIX M = INERTIA MATRIX ASSOCIATED WITH 2ND-ORDER EOM
C
      DO 9 I=1,NPP/2
        READ(NPG,*) (MR(I,J),J=1,NPP/2)
        DO 12 J=1,NPP/2
          M(I,J)=CMPLX(MR(I,J))
12      CONTINUE
9      CONTINUE
C
C MATRIX T IS THE MEASURED-VARIABLE/STATE TRANSFORMATION MATRIX
C
      DO 13 I=1,NPP
        READ(NPG,*) (TR(I,J),J=1,NPP)
        DO 13 J=1,NPP
          T(I,J)=CMPLX(TR(I,J))
13     CONTINUE
        CALL MATINV(NPP,T,TINV)

      REWIND(UNIT=NPG)
C
C READ SYSTEM INFORMATION FOR EIGENVALUE-DETERMINATION
C
      DO 50 I=1,NEV
        READ(NEG,*) (AR2(I,J),J=1,NEV)
        DO 49 J=1,NEV
          AS2(I,J)=CMPLX(AR2(I,J))
49     CONTINUE
50     CONTINUE

      READ(NEG,*) (BR2(I),I=1,NEV)
      DO 51 I=1,NEV
        BS2(I)=CMPLX(BR2(I))*SG
51     CONTINUE

      DO 53 I=1,NEV/2
        READ(NEG,*) (MR2(I,J),J=1,NEV/2)
        DO 54 J=1,NEV/2
          MS2(I,J)=CMPLX(MR2(I,J))
54     CONTINUE
53     CONTINUE
      DO 55 I=1,NEV
        READ(NEG,*) (T2R(I,J),J=1,NEV)
        DO 55 J=1,NEV
          T2(I,J)=CMPLX(T2R(I,J))
55     CONTINUE

      RETURN
      END

```

```

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
*
*   KGETA IS ONE PROGRAM AMONG A GROUP WHICH TOGETHER SOLVE FOR           *
*   THE GAINS ASSOCIATED WITH A FEEDBACK CONTROL SYSTEM.                 *
*
*   SPECIFICALLY, KGETA DEFINES THAT PART OF THE SYSTEM MATRIX         *
*   DEPENDENT UPON ANGLE TH.                                             *
*
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

```

```

SUBROUTINE KGETA(N,TH,AIN,M,AOUT)

```

```

COMPLEX*8 K(20,20),MINV(20,20),AIN(20,20),AOUT(20,20),M(20,20)
COMPLEX*8 P(20,20),W(20),Z(20,20),WK(40)
REAL*8 TH

```

```

IF(ABS(TH).LE.1.E-20) TH=1.E-20

```

```

C
C SAVE THE MATRIX A AS A1 SO THAT UPON RETURNING TO THE CALLING
C PROGRAM THE MATRIX A MAY REMAIN AS IT ORIGINALLY WAS
C

```

```

DO 10 I=1,N
  DO 20 J=1,N
    AOUT(I,J)=AIN(I,J)
  CONTINUE
20 CONTINUE
10 CONTINUE

```

```

C
C DETERMINE INVERSE OF M, INERTIAL MATRIX
C

```

```

CALL MATINV(N/2,M,MINV)

```

```

C
C DETERMINE ANGLE-DEPENDENT 'STIFFNESS' MATRIX
C

```

```

IF(TH.EQ.0.) D=0.
IF(TH.NE.0.) D=SIN(TH)/TH
E=COS(TH)

```

```

K(1,1)=CMPLX(11.939*D)
K(1,2)=CMPLX(.84901*E)
K(1,3)=CMPLX(.33913*E)
K(1,4)=CMPLX(.20254*E)
K(2,1)=CMPLX(.84901*D)
K(2,2)=CMPLX(.19449+.0774*E)
K(2,3)=CMPLX(-.014991*E)
K(2,4)=CMPLX(-.038786*E)
K(3,1)=CMPLX(.33913*D)
K(3,2)=CMPLX(-.014991*E)
K(3,3)=CMPLX(3.9676+.2212*E)
K(3,4)=CMPLX(.049166*E)
K(4,1)=CMPLX(.20254*D)
K(4,2)=CMPLX(-.038786*E)
K(4,3)=CMPLX(.049166*E)
K(4,4)=CMPLX(32.320+.6630*E)

```

```

*
* NOTE: FOR NON-LINEAR CANCELLATION SCHEME ALL K(I,J) ARE SET EQUAL
* TO 0.0 EXCEPT FOR K(2,2)= 0.19449, K(3,3)= 3.9676 AND
* K(4,4)= 32.320
*

```

```

C
C THE APPROPRIATE QUADRANT OF SYSTEM MATRIX A=MINV*K
C

```

```

CALL MULT(N/2,MINV,K,P)

```

```

DO 39 IP=N/2+1,N
  DO 41 JP=1,N/2
    AOUT(IP,JP)=-P(IP-N/2,JP)
41 CONTINUE
39 CONTINUE

```

```

RETURN
END

```

```

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
*
*   'PEEPC2 FORTRAN' IS ONE PROGRAM AMONG A GROUP WHICH TOGETHER
*   SOLVE FOR THE GAINS ASSOCIATED WITH FEEDBACK CONTROL SYSTEM.
*
*   'PEEPC2 FORTRAN' IS RESPONSIBLE FOR COMPUTING THE GAINS WHICH
*   FORCE THE SYSTEM TO ADOPT THE SPECIFIED EIGENVALUES. THE LOGIC
*   OF THIS PROGRAM FOLLOWS THAT OUTLINED BY BROGAN IN HIS TEXT
*   'MODERN CONTROL THEORY'.
*
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

```

```

SUBROUTINE PEEPC2(N,M,LAMBDA,A,B,TH1,K,T2)

```

```

COMPLEX*8 PHI(20),PHI2(20,20),KTEMP,PTEMP,LAMBDA(20),IDD(20,20)
COMPLEX*8 ID(20,20),A(20,20),B(20),PSI(20,20),PSI2(20,20)
COMPLEX*8 WKAREA(40),PSS(20,20),PSI3(20,20),WK(40),S(40),G(20,20)
COMPLEX*8 TH,MOD,PG(100,100),PP(100,100),T2(20,20),TINV(20,20)
REAL*8 K(20),TH1,K2(20)

```

C BEGIN THE ACTUAL COMPUTATIONAL ALGORITHM. EACH POLE WILL BE C DEALT WITH SEQUENTIALLY. BECAUSE OUTPUT FEEDBACK IS ASSUMED, C THE NUMBER OF SPECIFIED POLES IS M, A NUMBER THAT IS LESS THAN C OR EQUAL TO SYSTEM MODEL SIZE N.

C THE PURPOSE OF THE LOOP DEFINED IMMEDIATELY BELOW (LINE 100) IS C TO SEQUENTIALLY BUILD THE MATRIX G. THE SIZE OF G IS M*M: EACH C COLUMN IS RELATED TO A SPECIFIC EIGENVALUE

```

DO 100 ILAM=1,M

```

C FIRST WE DEFINE THE MATRIX (LAMBDA * IDENTITY)

```

DO 30 ID1=1,N
DO 20 ID2=1,N
IF(ID1.EQ.ID2) ID(ID1,ID2)=LAMBDA(ILAM)
IF(ID1.NE.ID2) ID(ID1,ID2)=0.0
20 CONTINUE
30 CONTINUE

```

C NOW WE DETERMINE THE MATRIX PSI=ID-A

```

DO 60 I1=1,N
DO 50 I2=1,N
PSI(I1,I2)=ID(I1,I2)-A(I1,I2)
PSS(I1,I2)=PSI(I1,I2)
50 CONTINUE
60 CONTINUE

```

C DETERMINE THE INVERSE OF THIS MATRIX PSI

```

CALL MATINV(N,PSI,PSI2)

```

C CALCULATE THE MATRIX PHI=PSI2 * B

```

DO 80 J1=1,N
PHI(J1)=(0.0,0.0)
DO 70 J2=1,N
PTEMP=PSI2(J1,J2)*B(J2)
PHI(J1)=PHI(J1)+PTEMP
70 CONTINUE
80 CONTINUE

```

C DETERMINE THE MATRIX G, DEFINED TO BE T2*PHI(J)

```

DO 90 I=1,M
G(I,ILAM)=(0.,0.)
DO 89 J=1,N
G(I,ILAM)=G(I,ILAM)+T2(I,J)*PHI(J)
89 CONTINUE
90 CONTINUE

```

```
100 CONTINUE
C DETERMINE INVERSE OF THIS MATRIX G
  CALL MATINV(M,G,PHI2)
C FINALLY, COMPUTE GAIN VECTOR K
  DO 120 I=1,M
    K(I)=0.0
  DO 110 J=1,M
    KTEMP=-PHI2(J,I)
    K(I)=K(I)+KTEMP
110 CONTINUE
120 CONTINUE

  WRITE(16,130) TH1,(K(I),I=1,N/2)
  WRITE(16,131) (K(I),I=N/2+1,N)
130 FORMAT(F7.4,2X,4(E12.6,2X))
131 FORMAT(9X,4(E12.6,2X))
3   FORMAT(8F9.4)

  RETURN
  END
```

```

*****
*
*   'KORDR1 FORTRAN' IS ONE WITHIN A SET OF PROGRAMS WHICH TOGETHER
*   SOLVE FOR THE GAINS ASSOCIATED WITH A FEEDBACK CONTROL SYSTEM.
*
*   IF THE SYSTEM USED FOR POLEPICKING IS DIFFERENT IN SIZE FROM
*   THE SYSTEM USED FOR EIGENVALUE DETERMINATION, THE GAINS AS
*   CALCULATED WILL NECESSARILY CORRESPOND TO DIFFERENT OUTPUT
*   VARIABLES. THIS PROGRAM RE-ORDERS THE GAINS SO AS TO
*   CIRCUMVENT THIS DIFFICULTY.
*
*****

```

```

      SUBROUTINE KORDR1(NVM,VM,NEV,INT)
      REAL*8 KIN(20),KOUT(20)
      INTEGER VM(20)
C
C BEGIN COMPUTATIONAL SEQUENCE. EACH LOOP CORRESPONDS TO A DIFFERENT
C VALUE OF BEAM-ROOT ANGLE,TH
C
      DO 100 ICT=1,INT
C
C READ THE INCORRECTLY-ORDERED GAINS
C
      READ(16,*) TH,(KIN(I),I=1,NEV/2)
      READ(16,*) (KIN(I),I=NEV/2+1,NEV)
C
C INITIALIZE ALL VALUES OF KOUT TO BE ZERO. KOUT WILL SOON BE
C TRANSFORMED INTO THE CORRECTLY-ORDERED GAIN VECTOR.
C
      DO 10 I=1,NEV
          KOUT(I)=0.
10    CONTINUE
C
C DEFINE THE VALUES OF KOUT USING THE DIRECTING VECTOR VM
C
      DO 20 I=1,NVM
          KOUT(VM(I))=KIN(I)
20    CONTINUE
100  CONTINUE

      RETURN
      END

```

```

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
*
* 'SUBRS FORTRAN' IS ONE PROGRAM WITHIN A GROUP WHICH TOGETHER
* SOLVE FOR THE GAINS ASSOCIATED WITH A FEEDBACK CONTROL SYSTEM.
*
* 'SUBRS FORTRAN' TRANSFORMS THE TABLE OF DISCRETE GAINS INTO
* A SET OF M POLYNOMIALS. M IS THE NUMBER OF VARIABLES BEING
* SENSED, AND THEREFORE THE NUMBER OF GAINS BEING SPECIFIED.
*
* DEFINED IN THE FUNCTION 'FUNCT', BELOW, ARE THE ORTHOGONAL
* POLYNOMIALS (LEGENDRE-TYPE) USED FOR CURVE FITTING. THE NUMBER
* OF DIFFERENT POLYNOMIALS TO BE USED IS AT YOUR DISCRETION.
* IN GENERAL, MORE POLYNOMIALS MEAN A BETTER DATA FIT. ON THE
* OTHER HAND, MORE POLYNOMIALS MEAN MORE COMPUTATION TIME FOR
* THE CONTROLLER, WHICH CAN POTENTIALLY CAUSE PROBLEMS.
* JUDGEMENT MUST BE EXERCISED.
*
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

```

```

SUBROUTINE SUBRS(NPP,NPOLY,INT,A1,A2,A3,A4,A5,A6,A7,A8)

```

```

REAL X(100),WK(1000),A1(100),A2(100),A3(100),A4(100)
REAL A5(100),A6(100),A7(100),A8(100),R(20),Y(100,100)
REAL Y1(100),Y2(100),Y3(100),Y4(100),Y5(100),Y6(100),Y7(100)
REAL Y8(100)
EXTERNAL FUNNC

```

```

DATA Y/10000*0./

```

```

C
C READ DISCRETE GAIN DATA, CURRENTLY IN TABULAR FORM
C

```

```

DO 10 I=1,INT
READ(11,*) X(I),(Y(I,J),J=1,NPP/2)
READ(11,*) (Y(I,J),J=NPP/2+1,NPP)
10 CONTINUE
98 FORMAT(F7.4,2X,4(E12.6,2X))
99 FORMAT(9X,4(E12.6,2X))

```

```

C
C THE FOLLOWING ASSIGNMENTS ARE NECESSARY BECAUSE THE IMSL ROUTINE
C 'IFLSQ' REQUIRES THE GAINS TO BE IN VECTOR FORM RATHER THAN IN
C MATRIX FORM.
C

```

```

DO 20 I=1,INT
Y1(I)=Y(I,1)
Y2(I)=Y(I,2)
Y3(I)=Y(I,3)
Y4(I)=Y(I,4)
Y5(I)=Y(I,5)
Y6(I)=Y(I,6)
Y7(I)=Y(I,7)
Y8(I)=Y(I,8)
20 CONTINUE

```

```

C
C DETERMINE LEAST-SQUARES POLYNOMIAL CURVE-FIT. EACH 'Y-' CORRESPONDS TO
C A DIFFERENT GAIN, AND EACH VECTOR 'A-' CONTAINS THE RELATED CURVE-FIT
C COEFFICIENTS.
C

```

```

CALL IFLSQ(FUNNC,X,Y1,INT,A1,NPOLY,WK,IER)
CALL IFLSQ(FUNNC,X,Y2,INT,A2,NPOLY,WK,IER)
CALL IFLSQ(FUNNC,X,Y3,INT,A3,NPOLY,WK,IER)
CALL IFLSQ(FUNNC,X,Y4,INT,A4,NPOLY,WK,IER)
CALL IFLSQ(FUNNC,X,Y5,INT,A5,NPOLY,WK,IER)
CALL IFLSQ(FUNNC,X,Y6,INT,A6,NPOLY,WK,IER)
CALL IFLSQ(FUNNC,X,Y7,INT,A7,NPOLY,WK,IER)
CALL IFLSQ(FUNNC,X,Y8,INT,A8,NPOLY,WK,IER)

```

```

DO 9 I=1,NPOLY
WRITE(12,8) A1(I),A2(I),A3(I),A4(I)
WRITE(12,8) A5(I),A6(I),A7(I),A8(I)
8 FORMAT(4(E12.6,2X))

```



```

9  CONTINUE
   PRINT*, ' '
   REWIND(UNIT=12)
   RETURN
   END

```

```

*****
*
*   THE FOLLOWING FUNCTION WHICH DEFINES THE POLYNOMIALS USED FOR
*   CURVE-FITTING WAS TAKEN FROM 'IFLSQ' SUPPORTING DOCUMENTATION
*   PROVIDED BY IMSL.
*
*****

```

```

FUNCTION FUNNC(K,X)
REAL X,XA,XB,T,PKM1,PKM2,RI
INTEGER K,I
IF(ABS(X).LT.1.E-77) X=0.

XA=-6.282
XB=6.282
T=(2.*X-XA-XB)/(XB-XA)
FUNNC=1.
IF(K.EQ.1) RETURN
FUNNC=T
IF(K.EQ.2) RETURN
PKM2=1.
PKM1=T
DO 5 I=3,K
  RI=I
  FUNNC=((2.*RI-3.)*T*PKM1-(RI-2.)*PKM2)/(RI-1.)
  PKM2=PKM1
  PKM1=FUNNC
5  CONTINUE

RETURN
END

```

```

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
*
*   'AGETC FORTRAN' IS ONE PROGRAM AMONG A GROUP WHICH TOGETHER
*   SOLVE FOR THE GAINS ASSOCIATED WITH A FEEDBACK CONTROL SYSTEM.
*
*   THIS ROUTINE IS SPECIFICALLY GEARED TOWARDS IMPLEMENTATION
*   INTO THE EXPERIMENTAL SYSTEM. IT IS BEST THAT THE GAIN-
*   CALCULATION ALGORITHM BE AS FAST AS POSSIBLE. RATHER THAN
*   COMPUTING THE LEGENDRE POLYNOMIALS EACH TIME THAT SYSTEM
*   VARIABLES ARE MEASURED, IT WOULD BE SIMPLER TO USE THE
*   POLYNOMIALS (1.,TH,TH $\phi$ 2,TH $\phi$ 3,...,TH $\phi$ 7). OBVIOUSLY, THE
*   COEFFICIENTS ASSOCIATED WITH THIS NEW POLYNOMIAL SET ARE
*   DIFFERENT THAN THE CURRENT COEFFICIENTS. 'AGETC FORTRAN'
*   CALCULATES THESE NEW COEFFICIENTS.
*
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

```

```

SUBROUTINE AGETC

```

```

REAL*8 A(20,20),C(20,20)

```

```

REWIND(UNIT=12)

```

```

DO 10 I=1,8
  READ(12,*) (A(I,J),J=1,4)
  READ(12,*) (A(I,J),J=5,8)

```

```

10 CONTINUE

```

```

C
C THE FOLLOWING EQUATIONS USED TO DETERMINE THE NEW COEFFICIENTS
C NECESSARILY CORRESPOND TO LEGENDRE POLYNOMIALS
C

```

```

DO 20 I=1,8
  C(1,I)=A(1,I)-.5*A(3,I)+.375*A(5,I)-.3125*A(7,I)
  C(2,I)=.15918*A(2,I)-.2388*A(4,I)+.2985*A(6,I)-.3482*A(8,I)
  C(3,I)=3.801E-2*A(3,I)-.1513E-2*A(5,I)+.1663*A(7,I)
  C(4,I)=1.008E-2*A(4,I)-3.530E-2*A(6,I)+7.941E-2*A(8,I)
  C(5,I)=2.809E-3*A(5,I)-1.264E-2*A(7,I)
  C(6,I)=8.049E-4*A(6,I)-4.427E-3*A(8,I)
  C(7,I)=2.349E-4*A(7,I)
  C(8,I)=6.945E-5*A(8,I)

```

```

20 CONTINUE

```

```

DO 40 I=1,8
  WRITE(14,50) (C(I,J),J=1,4)
  WRITE(14,50) (C(I,J),J=5,8)

```

```

40 CONTINUE

```

```

50 FORMAT(4(E12.6,2X))

```

```

REWIND(UNIT=12)

```

```

RETURN
END

```

```

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
*
*   'GAINS FORTRAN' IS ONE PROGRAM AMONG A GROUP WHICH TOGETHER
*   SOLVE FOR THE GAINS ASSOCIATED WITH A FEEDBACK CONTROL SYSTEM.
*
*   'GAINS FORTRAN' IS USED ONLY AFTER CURVE-FITTING THE DISCRETE
*   GAINS TO POLYNOMIALS, AS DISCUSSED IN 'SUBRS FORTRAN'.
*
*   THE OUTPUT OF 'SUBRS FORTRAN' IS A MATRIX OF COEFFICIENTS
*   CORRESPONDING TO THE LEGENDRE POLYNOMIALS.
*
*   WHEN ACTUALLY RUNNING THE EXPERIMENTAL SYSTEM, OR WHEN
*   COMPUTING EIGENVALUES IN THIS SET OF PROGRAMS, IT IS
*   NECESSARY TO COMPUTE THE GAINS--WHICH ARE FUNCTIONS OF
*   THESE COEFFICIENTS, THE POLYNOMIALS, AND ANGLE TH.
*   THIS IS THE PURPOSE OF 'GAINS FORTRAN'.
*
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

```

```

SUBROUTINE GAINS(TH,K,A1,A2,A3,A4,A5,A6,A7,A8)

```

```

REAL*8 A1(20),A2(20),A3(20),A4(20),F(20),K(20)
REAL*8 A5(20),A6(20),A7(20),A8(20)

```

```

C THE LEGENDRE ORTHOGONAL POLYNOMIALS ARE DEFINED BELOW
C

```

```

THA=-6.282
THB=6.282
T1=(2.*TH-THA-THB)/(THB-THA)
F(1)=1.
F(2)=T1
DO 10 KK=3,8
PKM2=1.
PKM1=T1
DO 20 I=3, KK
RI=I
F(I)=((2.*RI-3.)*T1*PKM1-(RI-2.)*PKM2)/(RI-1.)
PKM2=PKM1
PKM1=F(I)
20 CONTINUE
F(KK)=F(I)
10 CONTINUE

```

```

C INITIALIZE GAIN VECTOR
C

```

```

15 DO 15 I=1,8
K(I)=0.

```

```

C COMPUTE GAINS
C

```

```

DO 30 I=1,8
K(1)=K(1)+A1(I)*F(I)
K(2)=K(2)+A2(I)*F(I)
K(3)=K(3)+A3(I)*F(I)
K(4)=K(4)+A4(I)*F(I)
K(5)=K(5)+A5(I)*F(I)
K(6)=K(6)+A6(I)*F(I)
K(7)=K(7)+A7(I)*F(I)
K(8)=K(8)+A8(I)*F(I)
30 CONTINUE

```

```

RETURN
END

```

```

XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX
*
*   'MATINV FORTRAN' PERFORMS MATRIX INVERSION ON COMPLEX MATRICES.
*
*   THIS IS ACCOMPLISHED USING ELEMENTARY MATRIX OPERATIONS,
*   GAUSSIAN ELIMINATION AND PARTIAL PIVOTING.
*
XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXXX

```

```

SUBROUTINE MATINV(N,A,B)

```

```

C
C MATRIX INVERSION USING ELEMENTARY OPERATIONS
C

```

```

COMPLEX*8 A(20,20),B(20,20),INTER(20,2),TEMPA(1,20),TEMPB(1,20)
COMPLEX*8 XMULT,INV(20,20),A1(20,20)
INTEGER I,J,D,N,FF,MQB,MQE,MQI,S,G,H

```

```

C
C TAKE NECESSARY STEPS SO THAT MATRIX 'A' MAY BE RETURNED TO THE
C CALLING PROGRAM AS IT WAS RECEIVED
C

```

```

DO 10 I=1,N
DO 8 J=1,N
A1(I,J)=A(I,J)
8 CONTINUE
10 CONTINUE

```

```

C
C DEFINE THE IDENTITY MATRIX B WHICH WILL EVENTUALLY BE
C CONVERTED INTO THE SOLUTION MATRIX
C

```

```

15 DO 30 I=1,N
DO 20 J=1,N
IF (I.EQ.J) THEN
B(I,J)=(1.0,0.0)
ELSE
B(I,J)=(0.0,0.0)
END IF
20 CONTINUE
30 CONTINUE

```

```

C
C USE GAUSSIAN ELIMINATION TO ACHIEVE ZEROS ABOVE, BELOW PIVOT
C

```

```

DO 130 FF=1,N
IF(FF.EQ.N) GOTO 125
70 MQB=FF+1
MQE=N
MQI=1
75 DO 120 I=MQB,MQE,MQI

XMULT=A(I,FF)/A(FF,FF)
DO 110 J=1,N
A(I,J)=A(I,J)-XMULT*A(FF,J)
B(I,J)=B(I,J)-XMULT*B(FF,J)
110 CONTINUE
A(I,FF)=0.0
120 CONTINUE
IF(FF.EQ.1) GOTO 130
IF(MQI.EQ.(-1)) GOTO 130
125 MQB=FF-1
MQE=1
MQI=-1
GOTO 75
130 CONTINUE

```

```

C
C DIVIDE EACH DIAGONAL ELEMENT OF 'A' MATRIX BY ITSELF TO FORM
C IDENTITY, THEREBY ALSO TRANSFORMING B MATRIX INTO NEARLY
C FINISHED ARRANGEMENT
C

```

```

DO 150 I=1,N
DO 140 J=1,N

```

```
      B(I,J)=B(I,J)/A(I,I)
140  CONTINUE
150  CONTINUE
C
C REINSTATE 'A' TO ITS INITIAL STATUS
C
      DO 220 I=1,N
        DO 210 J=1,N
          A(I,J)=A1(I,J)
210   CONTINUE
220  CONTINUE

      RETURN
      END
```

APPENDIX B

Simulation Program, Control Scheme 1

PROGRAM SIMULATION OF SYSTEM RESPONSE, CONTROL SCHEME 1

C
 C PURPOSE: DYNAMIC SIMULATION OF A FLEXIBLE BEAM CANTILEVERED FROM
 C A SIMPLE SUPPORT (SHAFT WITH BEARINGS) IN THE GRAVITATIONAL
 C FIELD. A MOTOR IS COUPLED TO THE SHAFT, AND IS USED TO
 C CONTROL BOTH THE ANGULAR POSITION AND LATERAL VIBRATION
 C OF THE BEAM. SENSORS ARE ATTACHED TO THE BEAM SO THAT THE
 C THE BEAM CONFIGURATION AT ANY INSTANT OF TIME MAY BE
 C KNOWN. THESE SENSORS INCLUDE A POTENTIOMETER FOR ANGULAR
 C POSITION, AND THREE STRAIN GAGE GROUPS ALONG THE BEAM LENGTH
 C FOR BEAM SHAPE. THE OUTPUT OF THESE SENSORS IS SENT TO A
 C CONTROLLER (DIGITAL OR ANALOG) SO THAT MOTOR COMMAND SIGNALS
 C MAY BE DETERMINED. THE DEVELOPMENT OF AN ALGORITHM TO
 C DETERMINE SUCH COMMANDS IS THE EMPHASIS OF WORK FOR WHICH
 C THIS PROGRAM HAS BEEN WRITTEN. THE CONTROLLER RECEIVES
 C A MAXIMUM OF FOUR INCOMING SIGNALS (FROM THE SENSORS
 C MENTIONED ABOVE) AND USES THESE AS WELL AS THE TIME-
 C DERIVATIVES OF THESE SIGNALS (ESTIMATED BY THE CONTROLLER
 C ITSELF) IN ADDITION TO THE POSITION COMMAND OF THE SYSTEM
 C OPERATOR TO CALCULATE THE APPROPRIATE TORQUE TO BE APPLIED
 C BY THE MOTOR. THIS TORQUE COMMAND IS STRENGTHENED BY A
 C CURRENT-AMPLIFIER AND THEN SENT TO THE MOTOR. BEAM
 C CONFIGURATION IS ONCE AGAIN SENSED AND THE CONTROL PROCESS
 C IS REPEATED.

C PROGRAM: THIS CODE IS WRITTEN SPECIFICALLY FOR ACSL (ADVANCED
 C CONTINUOUS SIMULATION LANGUAGE), DISTRIBUTED BY MITCHELL
 C & GAUTHIER, ASSOC., INC. IT IS A LANGUAGE DESIGNED FOR
 C MODELLING AND EVALUATING THE PERFORMANCE OF SYSTEMS
 C DESCRIBED BY TIME-DEPENDENT, NONLINEAR DIFFERENTIAL
 C EQUATIONS. IT IS A FORTRAN-BASED CODE, AND WHEN NOT USING
 C ACSL-SPECIFIC COMMANDS THE USER MUST WRITE IN FORTRAN.

C DEFINE THE PORTION OF CODE TO BE EXECUTED ONLY AT THE BEGINING OF
 C THE SIMULATION RUN WITH THE DELIMITERS 'INITIAL' AND 'END'.
 C NOTE: THERE A FEW DIFFERENT TYPES OF BLOCKS WHICH CAN BE DEFINED,
 C 'INITIAL' BEING ONE, AND THE END OF EACH MUST BE DENOTED WITH AN
 C 'END' CARD. BLOCKS CAN BE NESTED WITHIN OTHER BLOCKS, AND IT IS
 C THEREFORE ADVANTAGEOUS TO LABEL THE DIFFERENT 'END' STATEMENTS AS
 C BELONGING TO A PARTICULAR BLOCK.

INITIAL

C DEFINE THE INTEGRATION ALGORITHM TO BE USED. THE LANGUAGE PROVIDES
 C THE USER WITH A CHOICE OF SIX (RUNGE-KUTTA SCHEMES, ETC.) OR THE
 C USER CAN WRITE HIS OWN. IALG=1 REFERS TO ADAM'S MOULTON, VARIABLE
 C STEP, VARIABLE ORDER METHOD.

ALGORITHM IALG=1

ARRAY A(4,4),B(4),X(4),AIN(8,8)

C DEFINE COMMUNICATION INTERVAL (TIME IN SECONDS) OVER WHICH INTEGRATION
 C SCHEME ERRORS ARE COMPUTED. DECREASING THIS TIME-STEP INCREASES
 C ACCURACY BUT ALSO INCREASES TOTAL SOLUTION TIME.

CINTERVAL CINT=.1

INTEGER N,MA,I

C DEFINE PARAMETERS HAVING CONSTANT VALUES.

CONSTANT I1=0, C1=.7, C2=1.632E-5
 CONSTANT M=2.0599E-3,EI=228.88,L=30.,NN=1.,N=4,TINC=1

CONSTANT MM=1.1592E-4,G=386.4,MB=0.,R=0.,KT=.900,KB=.04297
 CONSTANT RA=1.,CT=0,MA=2,IJOB=0,MGL=23.878,KG=20.

C DEFINE SENSOR TRANSFORMATION MATRIX

```

CONSTANT TM11=3.183, TM12=0., TM13=0., TM14=0.
CONSTANT TM22=-.0891, TM23=-.3699, TM24=-.9703
CONSTANT TM32=.0845, TM33=-.0155, TM34=-.7761
CONSTANT TM42=.0536, TM43=-.4851, TM44=-.0835

```

C INITIAL CONDITIONS, TIME-SPAN FOR WHICH SOLUTION IS TO BE CALCUCATED,
C AND POSITION COMMAND

```

CONSTANT THIC=0., Q1IC=0., Q2IC=0., Q3IC=0.
CONSTANT THDIC=0., Q1DIC=0., Q2DIC=0., Q3DIC=0.
CONSTANT TSTOP=5., RT=0.

```

C
C BEAM CURVATURE RELATED CONSTANTS
C

```

CONSTANT DS1=-697.42, DS2=-80.221, DS3=-29.128

```

```

CONSTANT D1=-32.0, D2=-12.782, D3=-76.338

```

```

CONSTANT DA1=.84975E-3, DA2=.017335, DA3=.14121

```

```

CONSTANT D11=55.677, D22=28.921, D33=30.047

```

```

CONSTANT D1111=.28666E-5, D1112=.33631E-5, D1113=-.46029E-7
CONSTANT D1122=.10430E-4, D1123=.11970E-4, D1133=.36272E-4
CONSTANT D1211=-.13822E-4, D1212=.63659E-4, D1213=.37213E-4
CONSTANT D1222=-.89578E-5, D1223=.22339E-4, D1233=-.50859E-4
CONSTANT D1311=-.43893E-5, D1312=-.57336E-4, D1313=.12297E-4
CONSTANT D1322=-.57702E-5, D1323=.79755E-5, D1333=-.19324E-5

```

```

CONSTANT D2211=.14708E-3, D2212=-.10661E-3, D2213=-.24463E-3
CONSTANT D2222=.25662E-3, D2223=.85279E-4, D2233=.10846E-2
CONSTANT D2311=-.20042E-3, D2312=.53914E-3, D2313=-.10291E-3
CONSTANT D2322=-.71209E-3, D2323=.13549E-3, D2333=-.37244E-3

```

```

CONSTANT D3311=.13161E-2, D3312=-.15096E-2, D3313=.39258E-3
CONSTANT D3322=.44914E-2, D3323=-.10824E-2, D3333=.54498E-2

```

```

CONSTANT DD11=2.9154, DD12=-.56503, DD13=-1.4619
CONSTANT DD22=8.336, DD23=1.8531, DD33=24.998

```

C
C DEFINE SYSTEM INERTIA MATRIX (CONSTANT)
C

```

A(1,1)=M*L*L/3.  $ A(1,2)=-M*DS1/L  $ A(1,3)=-M*DS2/L
A(1,4)=-M*DS3/L  $ A(2,1)=A(1,2)  $ A(2,2)=M*D11/L
A(2,3)=0.0  $ A(2,4)=0.0  $ A(3,1)=A(1,3)
A(3,2)=0.0  $ A(3,3)=M*D22/L  $ A(3,4)=0.0
A(4,1)=A(1,4)  $ A(4,2)=0.0  $ A(4,3)=0.0
A(4,4)=M*D33/L

```

C
C FEED IN GAIN COEFFICIENTS
C

```

CALL AINIT(AIN)

```

C
C DETERMINE TARGET ANGLE CORRESPONDING TO INPUT RT
C

```

RTV=RT*TM11

```

```

DO JP1 I=1,8
K(I)=AIN(1,I)
DO JP1 J=2,8
K(I)=K(I)*RTV+AIN(J,I)

```

JP1..CONTINUE

C
C THE FOLLOWING RELATE TO BEAM EQUILIBRIUM SHAPE AS A FUNCTION OF ANGLE
C

```

QT1=-3.2303*SIN(RT)
QT2=-.08547*SIN(RT)
QT3=-.00627*SIN(RT)
STT1=TM22*QT1+TM23*QT2+TM24*QT3
STT2=TM32*QT1+TM33*QT2+TM34*QT3
STT3=TM42*QT1+TM43*QT2+TM44*QT3

```



```
TEMP=(11.939*SIN(RT)+.84901*COS(RT)*QT1+.33913*COS(RT)*...
      QT2+.20254*COS(RT)*QT3)/1.
```

```
RTV=(K(1)*RTV+K(2)*STT1+K(3)*STT2+K(4)*STT3)/.6
```

```
END $"OF INITIAL"
```

```
C BEGIN DYNAMIC PORTION OF RUN, WHICH IS DENOTED BY 'DERIVATIVE' CARD.
C THE CODE IN THIS SECTION IS, WHEN TRANSLATED TO FORTRAN, REORDERED
C AS NECESSARY. THE 'PROCEDURAL' BLOCK IS USED TO SPECIFY
C PARTICULAR CODE WITHIN THE 'DERIVATIVE' BLOCK WHICH IS NOT TO BE
C REORDERED.
C THE DERIVATIVE BLOCK CODE IS RE-EXECUTED FOR EACH TIME STEP.
```

```
DERIVATIVE
PROCEDURAL
S1=Q1*Q1*Q1
S2=Q1*Q1*Q2
S3=Q1*Q1*Q3
S4=Q1*Q2*Q2
S5=Q1*Q2*Q3
S6=Q1*Q3*Q3
S7=Q2*Q2*Q2
S8=Q2*Q2*Q3
S9=Q2*Q3*Q3
S10=Q3*Q3*Q3
```

```
CQ1=2.*D1111*S1+(3.*D1112+3.*D1211)*S2+(3.*D1113+3.*D1311)*S3+...
      (D1122+D2211+4.*D1212)*S4+(2.*(D1123+D2311)+4.*(D1213+...
      D1312))*S5+(D1133+D3311+4.*D1313)*S6+(D1222+D2212)*S7+...
      (2.*(D1223+D2312)+D1322+D2213)*S8+(D1233+D3312+2.*...
      (D1323+D2313))*S9+(D1333+D3313)*S10
```

```
CQ2=(D1211+D1112)*S1+(4.*D1212+D2211+D1122)*S2+(2.*(D1213+...
      D1312)+D2311+D1123)*S3+3.*(D1222+D2212)*S4+(4.*(D1223+...
      D2312)+2.*(D2213+D1322))*S5+(D1233+D3312+2.*(D2313+...
      D1323))*S6+2.*D2222*S7+3.*(D2223+D2322)*S8+(D2233+...
      D3322+4.*D2323)*S9+(D2333+D3323)*S10
```

```
CQ3=(D1311+D1113)*S1+(2.*(D1312+D1213)+D2311+D1123)*S2+...
      (4.*D1313+D3311+D1133)*S3+(D1322+D2213+2.*(D2313+D1223))...
      *S4+(4.*(D1323+D2313)+2.*(D3312+D1233))*S5+3.*(D1333+...
      D3313)*S6+(D2322+D2223)*S7+(4.*D2323+D3322+D2233)*S8+...
      3.*(D2333+D3323)*S9+2.*D3333*S10
```

```
CTH1=Q1*D1+Q2*D2+Q3*D3
```

```
CDD1=Q1*DD11+Q2*DD12+Q3*DD13
CDD2=Q1*DD12+Q2*DD22+Q3*DD23
CDD3=Q1*DD13+Q2*DD23+Q3*DD33
CDD=Q1*CDD1+Q2*CDD2+Q3*CDD3
```

```
C DETERMINE APPROPRIATE GAINS FOR THE CURRENT VALUE OF TH.
```

```
C SO AS TO SIMULATE THE ACTUAL SYSTEM, IT IS SUPPOSED THAT SENSOR
C INPUT IS RECEIVED BY THE CONTROLLER RATHER THAN ACTUAL STATE
C INFORMATION. THE RELATIONSHIP BETWEEN STATES AND SENSORS IS AS
C DEFINED HERE.
```

```
THV=TM11*TH
```

```
C
C DETERMINE APPROPRIATE GAINS
```

```
C
      DO JP2 I=1,8
      K(I)=AIN(1,I)
      DO JP2 J=2,8
      K(I)=K(I)*THV+AIN(J,I)
JP2..CONTINUE
```

```
C
C DETERMINE STRAINS
```

```
C
```

```

ST1=TM22*Q1+TM23*Q2+TM24*Q3
st2=TM32*q1+TM33*q2+TM34*q3
st3=TM42*q1+TM43*q2+TM44*q3

```

```

C
C CALCULATE CONTROLLER OUTPUT VOLTAGE
C
  KP=.6*(RTV-K(1)*THV-K(2)*ST1-K(3)*ST2-K(4)*ST3-...
      K(5)*EST1-K(6)*EST2-K(7)*EST3-K(8)*EST4)

```

```

C
C DETERMINE B OF AX=B. B INCLUDES MOTOR TORQUE
C

```

```

  B(1)=-M*G*L*SIN(TH)/2.+M*G*COS(TH)*CTH1/L+KP+...
      M*G*SIN(TH)*CDD/(L*2.)-C1*THD
  B(2)=-EI*(Q1*da1+CQ1)-M*G*CDD1*COS(TH)/L+M*G*SIN(TH)*D1/L...
      -C2*Q1D*D11
  B(3)=-EI*(Q2*da2+CQ2)-M*G*CDD2*COS(TH)/L+M*G*SIN(TH)*D2/L...
      -C2*Q2D*D22
  B(4)=-EI*(Q3*da3+CQ3)-M*G*CDD3*COS(TH)/L+M*G*SIN(TH)*D3/L...
      -C2*Q3D*D33

```

```

C
C SOLVE SIMULTANEOUS LINEAR EQUATIONS USING GAUSSIAN ELIMINATION/
C BACK-SUBSTITUTION
C

```

```

  CALL GAUSS(N,A,B,X,I1)

  THDD=X(1)
  Q1DD=X(2)
  Q2DD=X(3)
  Q3DD=X(4)

```

```

  END $'OF PROCEDURAL'

```

```

C
C INTEGRATE ACCELERATIONS
C

```

```

  THD=INTEG(THDD,THDIC)
  Q1D=INTEG(Q1DD,Q1DIC)
  Q2D=INTEG(Q2DD,Q2DIC)
  Q3D=INTEG(Q3DD,Q3DIC)

```

```

  PROCEDURAL

```

```

C
C DETERMINE STRAIN RATES
C

```

```

  THVD = TM11 *THD
  ST1D = TM22*Q1D+TM23*Q2D+TM24*Q3D
  ST2D = TM32*Q1D+TM33*Q2D+TM34*Q3D
  ST3D = TM42*Q1D+TM43*Q2D+TM44*Q3D

```

```

C
C DETERMINE TIME-DERIVATIVES OF ESTIMATORS
C

```

```

  EST1D = G0*THVD - G0*G1*EST1
  EST2D = G0*ST1D - G0*G1*EST2
  EST3D = G0*ST2D - G0*G1*EST3
  EST4D = G0*ST3D - G0*G1*EST4
  END $"OF PROCEDURAL"

```

```

C
C INTEGRATE VELOCITIES
C

```

```

  TH=INTEG(THD,THIC)
  Q1=INTEG(Q1D,Q1IC)
  Q2=INTEG(Q2D,Q2IC)
  Q3=INTEG(Q3D,Q3IC)

```

```

C
C DETERMINE ESTIMATOR OUTPUTS
C

```

```

  EST1 = INTEG(EST1D,EST1IC)
  EST2 = INTEG(EST2D,EST2IC)
  EST3 = INTEG(EST3D,EST3IC)
  EST4 = INTEG(EST4D,EST4IC)

```

```

  TERMT(T.GE.TSTOP)

```

END \$"OF DERIVATIVE"

END \$"OF PROGRAM"

SUBROUTINE GAUSS(N,A,B,X,I1)

C SOLUTION OF SIMULTANEOUS EQUATIONS BY GAUSSIAN ELIMINATION

IMPLICIT REAL(A-F,0-Z)

REAL A(4,4),B(4),X(4),TEMPA(4),BIG

INTEGER FF,S,G,N,H

DATA TEMPA/4*0.0/

2 FORMAT(4(E15.8,3X))

C SEARCH FOR LARGEST PIVOT BELOW DIAGONAL

C (FF IS THE PIVOT INDEX)

```

C
DO 130 FF=1,(N-1)
    S=FF
    BIG=A(FF,FF)
    DO 40 G=(FF+1),N
        IF (ABS(A(G,FF)).LE.ABS(BIG)) GOTO 40
        S=G
        BIG=A(G,FF)
    40     CONTINUE
        IF (S.EQ.FF) GOTO 90

```

C SWITCH ROWS FOR LARGEST PIVOT

C FIRST, TEMPORARILY ASSIGN INITIAL PIVOT ROW TO 'TEMP'

```

C
DO 60 H=FF,N
    TEMPA(H)=A(FF,H)
    CONTINUE

```

```

60
C
DO 70 H=FF,N
    A(FF,H)=A(S,H)
    CONTINUE

```

```

70
C
DO 80 H=FF,N
    A(S,H)=TEMPA(H)
    CONTINUE

```

```

80
C
BTEMP=B(FF)
B(FF)=B(S)
B(S)=BTEMP

```

C WE NOW HAVE THE LARGEST ELEMENT IN THE PIVOT POSITION AND WILL

C THEREFORE BEGIN ELIMINATION SEQUENCE

```

90     DO 120 I=(FF+1),N
        XMULT= A(I,FF)/A(FF,FF)
        DO 110 J=(FF+1),N
            A(I,J)=A(I,J)-XMULT*A(FF,J)
        110     CONTINUE
            B(I)=B(I)-XMULT*B(FF)
            A(I,FF)=0.0
    115     FORMAT(1X,4F10.5)
    120     CONTINUE

```

130 CONTINUE

IF (A(N,N).EQ.0.0) GOTO 220

X(N)=B(N)/A(N,N)

DO 200 I=N-1,1,-1

 X(I)=B(I)

 DO 190 J=I+1,N

 X(I)=X(I)-A(I,J)*X(J)

190 CONTINUE

```

          X(I)=X(I)/A(I,I)
200  CONTINUE
      GOTO 230
220  WRITE(6,*) 'ENTRY A(N,N) = 0.0'
230  RETURN
      END

```

C DTGAIN DETERMINES THE GAINS AS A FUNCTION OF TH AND THE COEFFICIENT
 C MATRIX A, WHICH CORRESPONDS TO A LEAST-SQUARES CURVE FIT TO GAINS
 C CALCULATED AT DISCRETE VALUES OF TH. THE BASIS FUNCTIONS USED FOR
 C THIS CURVE FIT ARE THE LEGENDRE ORTHOGONAL POLYNOMIALS.

```

      SUBROUTINE DTGAIN(TH,F)
      REAL TH,F(20)

```

C THE VALUE 12.564 ASSUMES THAT GAINS CALCULATED BY THE 'FLEX' SET
 C OF PROGRAMS WERE FOR A RANGE OF ANGLES SPANNING -6.282 RAD,
 C +6.282 RADIANS.

```

      T1=(2.*TH)/12.564
      F(1)=1.
      F(2)=T1
      DO 10 K=3,8
          PKM2=1.
          PKM1=T1
          DO 20 I=3,K
              RI=I
              F(I)=((2.*RI-3.)*T1*PKM1-(RI-2.)*PKM2)/(RI-1.)
              PKM2=PKM1
              PKM1=F(I)
20      CONTINUE
      F(K)=F(I)
10      CONTINUE
      RETURN
      END

```

C SUBROUTINE AINIT READS THE GAIN COEFFICIENTS FROM DATA FILE 7

```

      ARRAY A(4,4),b(4),X(4),AIN(8,8),K(8)
      SUBROUTINE AINIT(AIN)
      REAL AIN(8,8)
      DO 10 I=8,1,-1
          READ(7,*) (AIN(I,J),J=1,8)
10      CONTINUE

      REWIND(UNIT=7)
      RETURN
      END

```

APPENDIX C

IBM PC programs

1. control scheme 1
2. control scheme 2

\$STORAGE:2
\$NOFLOATCALLS

```

C      PROGRAM DIG058
C
C      This program is for flexible beam control using an IBM PC equipped
C      with a DT2801 analog/digital converter card. The control algorithm
C      is as indicated in CONTROL SCHEME 1.
C
C      INTEGER*2 STATUS,GAIN,CHAN,ADV(12),STCH,ENCH,TIM,NOV,DACS
C      INTEGER*2 XSA,XSCD,XDV,XBAD
C      INTEGER*4 CLOCKD,NAV,FREQ
C      REAL AVE(8),EST(8),DV,G(2),G0,K(8),A(8,8),THD,RT,oldx(8),x(8)
C      REAL G1,k0
C
C      Read in gain-coefficient matrix
C
C      open (10,file='a:input.dat')
C      do 4 i=8,1,-1
C      read(10,*) (a(i,j),j=1,8)
C      continue
C
C      Set A/D and D/A constants, and read in interactively supplied
C      control directives
C
C      STATUS = 0
C      TIM = 0
C      GAIN = 1
C
C      A maximum of 8 system-related signals can be fed to the PC
C      through provided channels. All channels between and including
C      STCH and ENCH will be read
C
C      WRITE(*,10)
C      FORMAT(1X,'Enter starting and ending channel numbers')
C      READ(*,*)STCH,ENCH
C      STATUS = XSA(TIM,STCH,ENCH,GAIN)
C
C      Frequency at which the channels are to be read in tandem may
C      be as high as 2000 hz
C
C      WRITE(*,20)
C      FORMAT(1X,'Enter sample frequency ')
C      READ(*,*)FREQ
C      CLOCKD = (400000/FREQ)-.5
C      STATUS = XSCD(CLOCKD)
C      NC=(ENCH-STCH)+1
C      NOV = 4
C
C      The target command refers to the desired beam-root angle
C
C      WRITE(*,30)
C      FORMAT(1X,'Enter target THD ')
C      READ(*,*)THD
C      DO 35 I=1,2
C      G(I)=A(1,i)
C      DO 35 J=2,8
C      G(I)=G(I)*THD+A(j,i)
C      CONTINUE
C
C      Overall gain constant k0 refers to overall system gain, independent
C      of individual feedback signal gains.
C
C      WRITE(*,37)
C      FORMAT(1X,'Enter overall gain k0')
C
C      Because the beam is curved at equilibrium positions due to
C      gravity, the strains will not equal zero when the desired
C      target is achieved (unless = 0 or 180 ), and these strains
C      must be accounted for when specifying the target command rt
C
C      READ(*,*)k0

```

```
RT= 1/k0*(13.266-3.0809*COS(THD))*SIN(THD)+G(1)*THD*3.183
```

```
The two estimator constants can be set interactively
```

```
write(*,38)
format(lx,'enter filter constants g0,g1')
read(*,*) g0,g1
t=1/50.
```

```
Begin real-time calculation loops
```

```
AVE(I) are the incoming positon signals, and EST(I) are the
derivative estimates.
```

```
40 STATUS = XBAD(NOV,ADV(0))
   status = xwad(adv(nov-1))
   DO 50 I=STCH,ENCH
   AVE(I)=ADV(I)*.004882-10.
   x(i)=(oldx(i)+g0*g1*t*ave(i))/(1+g0*g1*t)
   EST(I)=g0*(ave(i)-x(i))
   oldx(i)=x(i)
50 CONTINUE
```

```
The gains are functions of  $\theta$  and the provided coefficients
(matrix A)
```

```
DO 70 I=1,8
K(I)=A(1,i)
DO 70 J=2,8
K(I)=K(I)*AVE(0)+A(j,i)
70 CONTINUE
```

```
DV is the control voltage to be sent through the power amp to
the servo motor.
```

```
DV=k0*(RT-(K(1)*AVE(0)+K(2)*AVE(1)+K(3)*ave(2)+K(4)*ave(3)+
1      k(5)*est(0)+k(6)*est(1)+k(7)*est(2)+k(8)*est(3)))
```

```
The remaining code refers to D/A conversion
```

```
DACS=0
NAV=(DV+10.)*204.8
STATUS = XDV(DACS,NAV)
GOTO 40
STOP
END
```

\$STORAGE:2
 \$NOFLOATCALLS

PROGRAM D0G085

C This program is for flexible beam control using an IBM PC equipped
 C with a DT2801 analog/digital converter card. The control algorithm
 C is as indicated in CONTROL SCHEME 2.

C INTEGER*2 STATUS,GAIN,CHAN,ADV(1000),STCH,ENCH,TIM,NOV,DACS
 C INTEGER*2 XSA,XSCD,XDV,XBAD,XWAD,XESC,XDSC,nc
 C INTEGER*4 CLOCKD,NAV,FREQ
 C REAL AVE(8),EST(8),DV,G0,K(8),THD,RT,T,X(9),dd(6),MULT,sinth
 C REAL QT1,QT2,QT3,QT1P,QT2P,QT3P,TH,SB1,SB2,SB3,SP1,SP2,SP3
 C REAL SUM1,SUM2,SUM3,SUM4,T2,term,OLDX(8),SIG(8),K0,G1,t1

data dd/2.9154,-.56503,-1.4619,8.336,1.8531,24.998/

C Set A/D and D/A constants, and read in interactively-supplied
 C control directives

C STATUS = 0
 C TIM = 0
 C GAIN = 1

C A maximum of 8 system-related signals can be fed to the PC through
 C the provided channels. All channels between and including STCH
 C and ENCH will be read.

C WRITE(*,10)
 10 FORMAT(1X,'Enter starting and ending channel numbers')
 C READ(*,*)STCH,ENCH
 C STATUS = XSA(TIM,STCH,ENCH,GAIN)

C Frequency at which the channels are read in tandem may be as high
 C as 2000 hz.

C WRITE(*,20)
 20 FORMAT(1X,'Enter sample frequency ')
 C READ(*,*)FREQ
 C CLOCKD = (400000/FREQ)
 C STATUS = XSCD(CLOCKD)
 C NC=(ENCH-STCH)+1

C T=NC/FREQ
 C WRITE(*,29)
 29 FORMAT(1X,'ENTER NOV')
 C READ(*,*) NOV

C This work required that only four channels be read (numbers 0
 C through 3). A time-varying target command from, say, a sinewave
 C generator, may input through channel 4 if desired.

C WRITE(*,*) 'TARGET FROM CHANNEL 4? (1=YES,0=NO)'
 C READ(*,*) CHAR
 C IF(CHAR.EQ.0) GOTO 28
 C THD=0.
 C GOTO 31

C Or, a constant target may simply be specified.

C WRITE(*,30)
 28
 30 FORMAT(1X,'Enter target THD ')
 C READ(*,*)THD

C In this control scheme only one gain-set, consisting of a maximum
 C of 8 numbers, is required. These are to be input interactively.

C WRITE(*,*)'ENTER 8 GAINS'
 31


```

READ(*,*) (K(I),I=1,8)
WRITE(*,37)
C
C The overall system gain is input here. For all work reported in
C the thesis, a constant value of k0=0.21 was used. This number
C should not be changed at will since the calculated gains depend
C upon this.
37 FORMAT(1X,'Enter overall gain K0')
READ(*,*)K0
C
C The gains associated with the estimators must be input
C
write(*,36)
36 format(1x,'enter filter gain constants g0,g1')
read(*,*) g0,g1
C
C The frequency at which commands are output by the PC using this
C algorithm was experimentally determined to be 38.5 hz.
C
t=1/38.5
C
C The target command is a function not only of the desired shaft
C angle, but also of the non-zero strain gage readings
C
35 RT= K(1)*THD*3.183-k(2)*.40464*sin(thd)-k(3)*.25701*sin(thd)-
1 k(4)*.14606*sin(thd)
WRITE(*,*) 'RT:',RT
dacs = 0
DO 48 I=1,NC
48 OLDX(I)=0.
C
C Begin real-time calculation loop
C
C AVE(I) are the incoming position signals, and EST(I) are the
C derivative-estimates
C
40 status = xbad(nov,adv(1))
C
DO 50 j=1,NOV
STATUS = XWAD(ADV(j))
SIG(J)=ADV(j)*.004882-10.
X(J)=(OLDX(J)+g0*g1*T*SIG(J))/(1+g0*g1*T)
EST(J)=g0*(SIG(J)-x(j))
50 OLDx(J)=x(j)
CONTINUE
C
C The following calculations are to determine the appropriate
C nonlinear terms to add to the control command so as to counter-act
C the nonlinear terms in EOM( ).
C
51 TH=SIG(1)*.3142
C
aa=sig(2)*sig(2)*79.4218+sig(3)*sig(3)*155.001+
1 sig(4)*sig(4)*98.5296+sig(2)*sig(3)*171.1267+
1 sig(2)*sig(4)*113.7509+sig(3)*sig(4)*121.1524
C
sinth = sin(th)
term=(-.013266*aa+11.939)*sinth
SUM4=.026532*COS(TH)*(sig(2)*201.0854+sig(3)*81.2219+
1 sig(4)*176.4095)
T2=TERM+SUM4
C
c AVE2=SIG(2)+.40464*SINTH
c AVE3=SIG(3)+.25701*SINTH
c AVE4=SIG(4)+.14606*SINTH
C
if(nov.eq.4) sig(5)=1.
C

```

C DV is the control command signal in volts
C

```
DV=K0*(RT-(K(1)*SIG(1)+K(2)*sig(2)+K(3)*sig(3)+K(4)*sig(4)+
1 sig(5)*(K(5)*EST(1)+K(6)*EST(2)+K(7)*EST(3)
1 +K(8)*EST(4))))+T2*.38138
```

C
C The remaining code refers to D/A conversion.
C

```
NAV=(DV+10.)*204.8
STATUS = XDV(DACS,NAV)
GOTO 40
STOP
END
```

APPENDIX D

Legendre polynomials

CURVE-FITS TO GAINS

IFLSQ (an imsl routine) finds coefficients c_i , $i = 1, 2, \dots, n$ such that

$$\sum_{j=1}^m [c_1 F_1(x_j) + \dots + c_n F_n(x_n) - y_j]^2$$

is minimized.

PROGRAMMING NOTES

If polynomial approximation is desired, an orthogonal polynomial basis should normally be used, since the basis set $F(K,X) = X^{*(K-1)}$ leads to a very ill-conditioned linear system for moderate or large N . The following Fortran function defines a set of Legendre polynomials which are orthogonal over the interval (XA, XB) (set $XA = X(1)$, $XB = X(M)$ normally).

```

REAL FUNCTION F(K,X)
INTEGER K,I
REAL, X,XA,XB,T,PKM2,PKM1,RI
DATA XA,XB/.../
T = (2.*X-XA-XB)/(XB-XA)
F = 1.0
IF(K.EQ.1) RETURN
PKM2 = 1.0
PKM1 = T
DO 5 I = 3,K
RI = I
F = ((2.*RI-3.)*T*PKM1-(RI-2.)*PKM2)/(RI-1.)
PKM2 = PKM1
PKM1 = F
5 CONTINUE
RETURN
END

```

APPENDIX E

Experimental apparatus

EXPERIMENTAL APPARATUS

1. Flexible beam
Steel, density = $.283 \text{ lbs/in}^3$
Dimensions: 30' * 3" * (1/32)"
2. Strain gages
Number: 12
3. Potentiometer
Manufacturer: Maurey Instrument Corporation
Model: LI2-P19
Serial number: S/N 1031
4. Bridge amplifiers
Number: 3
Manufacturer-model: Vishay/Ellis-11
Serial numbers:
 1. S/N 032466
 2. S/N 027485
 3. S/N 027483
5. Strip-chart recorder
Manufacturer-model: Gould-Brush 220
VPI & SU inventory number 6375
6. Power amplifier
Manufacturer: PMI Motors, division of Kollmorgen
Type: servo switching amplifier
Model: 00-88018-003
Serial number: 410003
7. Motor
Manufacturer: PMI Motors
Model: 08-01209-035
Type: U12M4T
Serial number: 3620816
8. Controller
IBM PC with Intel 8087 math coprocessor and Data Translation DT2801
analog/digital converter card

The vita has been removed
from the scanned document