STATE-FEEDBACK CONTROL OF A RIGID ROTOR SUPPORTED ON SQUEEZE-FILM BEARINGS

by

H.Basri AKgul

B.S. in M.E., Bogazici University, 1988

Submitted to the Institute for Graduate Studies in Science and Engineering in Partial Fulfillment of the requirements for the degree of

Master of Science

in

Mechanical Engineering

Bogazici University

1988

STATE-FEEDBACK CONTROL OF A RIGID-ROTOR SUPPORTED ON SQUEEZE-FILM BEARINGS

H.BASRI AKGUL

APPROVED BY:

Yrd. Doc. Dr. Osman TURKAY ....

(Thesis Supervisor)

Doc. Dr. Ahmet KUZUCU ....

Yrd. Doc. Dr. Mehmet CAMURDAN ....

ACKNOWLEDGEMENTS

I would like to express my sincere gratitude to Yrd. Doc. Dr. Osman Seyhun Turkay for his guidance, invaluable suggestions and encouragement during my study. Valuable suggestions by Prof. Dr. Yorgo Istefanopulos is gratefully acknowledged.

ABSTRACT

In this thesis, pole assignment problem, one of the most commonly used control schemes, is considered and special emphasis is given on the application of available pole assingment algorithms in multi-input systems.

Various methods such as Ackermann's Procedure, Modal Control, Direct Design Procedure and Phase-Variable Canonical Form are explained for determining the required feedback gains for arbitrary pole assignment in single-input systems. Chapter II which includes also the methods developed for the multivariable systems.

In Chapter III, the squeeze film-bearing theory is presented. Squeeze-film bearing equations, oil-film coefficients and the coordinate transformations are derived.

In Chapter IV, the selected models and their characteristics are discussed. State feedback control is applied to a rotor bearing system and the computer program is developed.

In Chapter V, the general conclusions of thesis are given.

ÖZET

Bu tez çalışmasında , en çok kullanılan kontrol yöntemlerinden biri olan durum değişken geri beslemeli kutup yerleştirilmesi yöntemi incelenmiştir. Çalışmanın büyük bir kısmı kutup yerleştirilmesi yönteminin çok girdili sistemlere uygulanmasına ayrılmıştır.

Bölüm II' de tek girdili sistemlerin kutuplarının yerleştirilmesinde gerekli besleme kazançlarını belirlemek için kullanılan Ackermann's Yöntemi, Modal Kontrol, Doğrudan Dizayn Yöntemi ve Phase-Variable Canonical Form gibi kullanılan yöntemler incelendi ve genelleştirilerek çok girdili sistemlere de uygulanabilecek duruma getirildi.

Bölüm III' de squeeze-film bearing teorisi sunuldu. Squeeze-film bearing denklemleri, yağ-film katsayıları ve koordinat dönüşümleri çıkarıldı.

Bölüm IV' de seçilen modeller ve bunların özellikleri tartışıldı. Geri beslemeli kontrol rotor-yatak sistemine uygulandı ve bilgisayar programının açıklaması yapıldı.

Bölüm V' de tezin genel sonuçları verildi.

TABLE OF CONTENTS

Page

LIST OF FIGURES

Page

Page

LIST OF TABLES

Page

LIST OF SYSMBOLS

A	Plant dynamic matrix
a_n	Coefficients of the characteristic equation
B	Input matrix of dimension
B_n	Normalized input matrix
C	Output matrix
C_{xx}, C_{xy}..	Oil-film damping coefficients in fixed co-ord. axes
C_{rr}, C_{rs}..	Oil-film damping coefficients in local co-ord. axes
c	Clearance between the bearing housing and journal
F	Closed-loop matrix
F(s)	Characteristic polynomial of the open-loop system
f_x, f_y	Oil-film forces in fixed co-ord. axes
f_r, f_s	Oil-film forces in local co-ord. axes
G(s)	Open-loop transfer function matrix
H(s)	Characteristic polynomial of the closed-loop system
h	Last row of the controllability matrix
K	State-feedback matrix
K_o	Output-feedback matrix
L	Loading factor
l	Bearing land length
M	Modal matrix
m	Mass per land of the journal
P	Tansformed matrix from controllability matrix
p_i	Controllability indices
p	Row vector for transformation

Q	Controllability matrix
q	Column vector for transformation
R	Radius of bearing
T	Transformation matrix
T_{ij}	Transfer functions
$U(s)$	Laplace transform of the input vector
u	Input vector
v	External input vector
$Y(s)$	Laplace transform of the output vector
x	State-space vector
z	State-space vector for transformation
W	Angular frequency
Φ	Angular deflection
μ_c	Controllability index
Δ	Determinate of the characteristic equation
λ_i	Eigenvalues
Λ	Normalized system matrix
η	Oil viscosity
K_s	Retainer spring stiffness coefficients
ϵ_0	Static eccentricity ratio
$(\dot{\ })$	Denotes differentiation with respect to time
$(\ )^T$	Denotes transpose of matrix
$(\hat{\ })$	Denotes transformed matrix

I. INTRODUCTION

1.1 General Background

The study of rotor dynamics has in recent years become of increasing importance in the engineering design of power systems. With the increase in performance requirements of high-speed rotating machinery in various fields such as gas turbines, process equipment, auxiliary power machinery and space applications, the engineer is faced with the problem of designing a unit capable of smooth operation under various conditions of speed and load.

At the turn of the century, Jeffcott [1] developed the fundementals of dynamic response of the single-mass unbalanced rotor on a massless elastic shaft mounted on the rigid bearing supports. The Jeffcott analysis of the single-mass model showed that operating speeds above the first critical speed were possible.

Any increase in the rotational speed causes the build-up of the vibration amplitude until the system fails. This new insight led to the need to consider the effects of the oil-film bearings on the rotor dynamic. Stodola [2] was the first to attribute stiffness coefficients to the oil-film bearings but neglected the damping properties. Since then a number of researchers have investigated the dynamics of the oil-film bearings. The most usual approach is to represent the oil filn journal bearing by eight linearised stiffness and damping coefficients.

In 1963 Cooper [3] was given the patent for the design of the squeeze-film bearing which is a special type of oil-film journal bearing. In the squeeze-film bearing applications the shaft is usually mounted in a roller-bearing whose outer race is prevented from rotating. The static load is supported by retainer springs, figure 1.1.1. The clearance region between the bearing housing and the outer race of the roller-bearing is filled with oil. In the case of squeeze- film bearing the four stiffness coefficients disappear and the bearing is characterised by the four damping coefficients which are a function of the bearing dimensions (l, R, c), the oil viscosity (η), journal static eccentricity ratio (ϵ_0) and the film extend along the journal. However, the effect of the oil supply pressure on the linearized coefficients does not appear in the theoretical relationships.

Oil-film journal bearings are frequently employed in turbomachinery. The stiffness and damping properties of the oil-film was examined by Smith [4] in 1969. This properties were used to provide an effective method for the passive control of vibration by correct selection of bearing parameters by Morrison [5] in 1976. However, these bearings may also cause rotor instability. The various types of unstable vibrations excited in the bearings are discussed by

Smith [4].

Bearing-induced instabilty can often be remedied by introducing a different design for the bearing, but Smith has noted that no single design provides a universal solution to the problem. This limitation is common to all passive forms of vibration control and has led to interesting techniques for the active control of rotor vibrations. The instability associated with oil-film bearings can be avoided if they are replaced by magnetic bearings. These elements can be used for the active control of vibrations and this is particularly significant in machines which are required to operate in excess of one or more critical speeds. These are examined by Schweitzer [6] in 1975.

The characteristics of a magnetic bearing for the active control of rotor vibrations were examined by Schweitzer and Lange [7] in 1976, who derived a multi-variable representation for these elements relating the output control force vector to the input vector. Bleuler and Schweitzer [8] in 1983 examined the use of two magnetic bearings to support a rigid shaft.

Stanway and Burrows [9] have evaluated the relative merits of various passive and active schemes for controlling the lateral vibrations of flexible rotor. The work was extented by Burrows and SahinKaya [10] to consider the open-loop control of multi-mode rotor-bearing systems. They highlighted the problems of designing close-loop control systems for multi-mode rotor-bearing.

1.2 Object Of The Work And Presentation Of The Thesis

The purpose of the work is to investigate possibility of being a state-feedback manner for multi-input systems which could be applicable to rigid rotor supported on squeeze-film bearing while the shaft rotates at a constant angular velocity and computer simulation program of selected manner.

The thesis consist of two parts.

In the first part (chapters II and III), the pole-placement problem of multi-variable systems in state-space representation is discussed in detail and basic methods developed in this field are introduced.

The second part of the thesis (chapter III and IV) deals with the squeeze-film bearing dynamics theoretically and simulation of state-feedback control to a rigid rotor supported on squeeze-film bearings.

II. STATE-FEEDBACK CONTROL

2.1 Introduction

One of the most popular techniques for altering the response characteristics of a control system is the application of linear state feedback. In the past decade, considerable effort has been made to understand exactly what feedback has offer and what its limitations are.

The fact that, one can use state feedback to assign the closed- loop system any desired self conjugate set of eigen values, provided that the open-loop system is controllable, is a well known and commonly used result [11]. For single-input system, this result is simple to derive and has been known for some time. Eigenvalue placement in multi-input systems was studied by Lagenhop [12], Wonham [13], Simon and Mitter [14], and Brunovsky [15]. Wonham was the first to prove the property of state- feedback and he applied to controllable multi-input systems.

Numerous eigenvalue-assignment algorithms have been devised for controllable multi-input time invariant linear systems. However, most of these algorithms proceed by reducing multi-input systems to equivalent single-input systems in the interest of computational tractability but thus unfortunately introduce diffuculties (such as the need to consider the cyclicity of plant matrices [16]) not associated with the original multi-input system. It is accordingly the purpose to present an assignment algorithm which deals directly with multi-input systems and which also relates eigenvalue-assignment directly to the fundamental structural properties of controllable multi-input time-invariant linear systems [17].

2.2 Some Aspects Of State-Feedback Control

2.2.1. Definitions

System : A system is a combination of components that act together and perform a certain objective. A system is not limited to physical ones. The concept of the system can be applied to abstract, dynamic phenomena such as those encountered in economics. The word system should therefore, be interpreted to imply physical, bioliogical, economics, etc., systems.

Disturbance : A disturbance is a signal which tends to adversely affect the value of the output system. If a disturbance is generated within the system, it is called internal; while an external disturbance is genarated outside the system and is an input.

State : The state of a dynamic system is the smallest set of variables (called state variables) such that knowledge of these variables at $t=t_0$, together with an input for $t \geq t_0$, completely determines the behavior of the system for any time $t \geq t_0$.

Thus, the state of a dynamic system at time t is uniquely determined by the state at time t_0 and the input for $t \geq t_0$ and it is independent of the state and input before t_0.

State Variables : The state variables of a dynamic system are the smallest set of variables which determine the state of the dynamic system. If at least n variables $x_1(t), x_2(t), \ldots, x_n(t)$ are needed to completely describe the behaviour a dynamic system and then such n variables $x_1(t), x_2(2), \ldots, x_n(t)$ are a set of state variables.

State Vector : If n state variables are needed to describe the behaviour of a given system, then these n state variables can be considered to be the n component of a vector x(t). Such a vector is called a state vector. A state vector is thus a vector which determines uniquely the system state x(t) for any $t \geq t_0$, once the input u(t) for $t \geq t_0$, is specified.

Feedback Control : Feedback control is an operation which in the presence of disturbances, tends to reduce the difference between the output of a system and the reference input (or an arbitrary varied, desired state) and which does so on the basis of this difference. Here only unpredictable disturbances (i.e., those unknown beforehand) are designated for as such, since with predictable or known disturbances, it is always possible to include compensation within the system so that measurements are unnecessary .

Feedback Control Systems : A feedback control system is one which tends to maintain a prescribed relationship between the output and the reference input by comparing these and using the difference as a means of control.

Open-Loop Control Systems : Open-loop control systems are control systems in which the output has no effect upon the control action. That is an open-loop control system, the output neither measured nor feedback for comparison with the input. Figure 2.1.1 shows the input-output relationship of such a system.

Closed-Loop Control Systems : A closed-loop control system is one in which the output signal has a direct effect upon the control action. The actuating error signal, which is the difference between the input signal and feedback signal (which may be the output signal or a function of the output signal and its derivatives), is fed to the controller so as to reduce the error and bring the output of the system to a desired value. In other words the term closed loop implies the use of feedback action in order to reduce system error. Figure 2.1.2 shows the input-output relationship of the closed-loop control systems [18].

2.2.2 Eigenstructure Assignment Via Linear State-Feedback Control.

Consider the state space representation of a multi-variable system

$$\dot{x} = Ax + Bu$$
$$y = Cx \qquad (2.1)$$

here and in the following, all vectors and matrices have real valued elements and all matrices are constant. In equation (2.1) A and B are matrices of dimension nxn and nxm respectively; x is an n-dimensional vector denoting the state and u is an m-dimensional input vector. Hence the matrix C is of dimension (pxn) where y is a p dimensional output vector. From now on, we will assume that all the states of system (2.1) are available and therefore the output equation will not be used.

The free response of the uncontrolled plant, i.e., when u(t) is equal to a zero vector, is given by a linear combination of the dynamical modes of the system, where the mode shapes are determined by the eigenvectors and the time domain characteristics by the pole locations of the system [19]. It is possible that for some reason or another the response of the uncontrolled plant is unsatisfactory. The system response may be too slow for a particular purpose or it may even be unstable due to positive real parts of its poles.

However, if control loops are introduced which generate the input vector by linear feedback of the state vector of the plant then the response characteristics of the resulting closed-loop system will no longer be determined by the eigen properties of A matrix, but by those of some new closed-loop plant matrix whose eigenproperties and its values will depend upon the precise nature of the feedback loops. It transpires that, by introducing appropriate feedback loops, it is possible to design a closed- loop system whose plant matrix is such that those of its eigenvalues which correspond to the controllable modes of the uncontrolled system can be assigned new values which lead to the closed-loop response characteristics that are superior to the corresponding characteristics of the original uncontrolled plant. If all the elements of the state vector x(t), somehow can be measured then it is possible to modify the external input u(t) such as ;

$$u(t) = Kx(t) + v(t) \tag{2.2}$$

where v(t) is a new external input an m-dimensional vector and K is a (mxn) feedback matrix , such that the closed-loop system equation becomes

$$\dot{x} = (A+BK)x + Bv \tag{2.3}$$

The main concern of the modal control theory is to choose an appropriate feedback gain matrix K so that the new dynamic matrix (A+BK) has a desired set of eigenvalues. In this chapter we want to answer the following questions:

i) Under which conditions is it possible to find an appropriate K matrix, such that a desired closed-loop characteristic polynomial is obtained ?

ii) What are the possible approaches to pole assignment problem if all of the state variables are not accesible ?

The procedure used to determine the K matrix will be discussed in the next chapter. Equation (2.3) indicates that the effect of the input variable defined by equation (2.2) is to change the plant matrix A to a new matrix (A+BK).

2.2.3 Controllability of Linear Systems

It is shown that the controllability of an open-loop system is equivalent to the possibility of assigning an arbitrary set of poles to the transfer matrix of the closed-loop system, formed by means of suitable linear feedback of the state. As an application of this result, it is shown that an open-loop system can be stabilized by linear feedback if and only if the unstable modes of its system matrix are controllable [13].

When one thinks about the conditions which have to be satisfied, so that the existence of K is guaranteed and one is immediately led to the idea, that the possibility of existence depends upon the controllability of the state x with respect to the external input u. The property of pole assignability which is shown to be equivalent to controllability of (2.1) in the usual sense.

To be precise, consider the following. Let be an arbitrary set of n complex numbers λ_i,

$$\Lambda = \{ \lambda_1, \lambda_2, \ldots\ldots\ldots \lambda_n \} \qquad (2.4)$$

such that any λ_i with Im $\lambda_i \neq 0$ appears in Λ in a conjugate pair. The necessary and sufficient condition for the existence of an (mxn) real matrix K, such that the closed-loop system matrix (A+BK) has the set Λ as its eigenvalues is the controllability of the pair (A,B), i.e., the existence of K implies that the (nxmn) controllability matrix of the system (2.1)

$$Q = \{ B, \ AB, \ldots\ldots\ldots A_{n-1}B \} \tag{2.5}$$

is of full rank n. Then the main result to be proved is the following.

THEOREM (2.1): For the n-th order dynamical system given in (2.1), let Λ (2.4) be an arbitrary desired set of complex numbers λ_i, such that any λ_i with Im $\lambda_i \neq 0$ appears in as a conjugate pair. The closed-loop system (2.3) has Λ for its set of eigenvalues if and only if (A,B) is controllable.

Linear state variable feedback is an important compensation technique in the synthesis of linear dynamical systems. However one should be aware of one important factor concerning linear state variable feedback, which can in many cases prevent its direct employment for closed-loop pole assignment.

In particular, on closer inspection of figure (2.1.3), it is appearent that the feedback path from the state x(t) through the gain matrix K crosses the boundary which encloses the original system. This clearly implies the ability to directly measure the entire internal n-dimensional state vector. In general, however, only the external input u(t) and output y(t) are directly measurable so that the control scheme given in figure (2.1.3) is not directly realizable. Since all the states of the system are required to implement the control law, we can introduce a state estimator (observer) into the system, such that the states are estimated using only the external input u(t) and output y(t). Hence in the realization of the control law (2.2) the n dimensional estimated state vector $\hat{x}(t)$ will be used in place of x(t). Obviously this idea of using a state estimator to reconstruct the unavailable states at the output, requires the system to be completely observable. It has been shown [20] that complete observability of the pair (A, C) is necessary for the realization of an estimator. Certainly the convergence rate of the estimator must be fast compared to the time constant of the system, such that no significant delay is added to the system performance. The block diagram of the system with an estimator causes a slight modification on figure (2.1.4). Under these conditions we can modify the statement of theorem (2.1) as follows:

THEOREM (2.2): Consider the n-th order system in (2.1) and assume that initially not all the states are available. Let Λ (2.4) be an arbitrary desired set of n complex number λ_i, such that any λ_i with Im $\lambda_i \neq 0$ appears in Λ in conjugate pair.

The closed-loop system (2.3) has for its set of eigenvalues e.i., complete and arbitrary pole placement is realizable, if and only if (A, B) is controllable and (A, C) is observable.

However, estimating the unavailable states via a state estimator has the disadvantage of considerably increasing the system order. Let us assume that pole placement is primarily used for plant stabilization. The plant, however, may not need as many feedback as there are states for its stabilization, since the response to the normal range of input is often determined by a few dominant poles of the system. Therefore one may try to construct feedback-loops only from the available output variables. Pole placement using only output feedback is certainly an alternative approach to using an estimator to establish the necessary state-feedback law. For pole placement using only output feedback the external input vector u(t) will be modified, and then it is equal to,

$$u(t)=K_0 y(t)+v(t)$$
$$u(t)=K_0 Cx(t)+v(t) \qquad (2.6)$$

the closed-loop system becomes:

$$\dot{x}= (A+BK_0 C)x + Bv \qquad (2.7)$$

the output feedback matrix K_0 must be chosen such that

$\det(A+BK_0C)$ will be equal to the desired characteristic polynomial to the discussed. However determining K_0, such that arbitrary pole placement is achieved, is not easy. It has been proved [21,22] that it is always possible to locate exactly p (p is the rank of the output matrix C) of the closed-loop poles to arbitrary locations. If some other additional constraints are also satisfied then all of the n closed-loop poles can be arbitrary placed using only output feedback [23].

2.3 State Feedback Control Methods

The theory of multivariable control system is well advenced and several methods exist for choosing a feedback law to achieve desired design objectives in choosing feedback law for controllable multivariable systems to achieve a desired dynamics for the closed-loop system poles to particular locations.

2.3.1 Ackermann's Procedure For Pole Assignment In Single-Input System

It is assumed that the process to be controlled can be described by the model

$$\dot{x} = Ax + bu \tag{2.8}$$

where u(t) represents the control variable, x(t) represents the state vector. A and b are system and input matrices respectively. When a feedback law of the form

$$u(t) = Kx(t) \tag{2.9}$$

is applied such that,

$$\det[\lambda I-(A+bK)] = \Delta(\lambda) \tag{2.10}$$

where the roots of $\Delta(\lambda)$ are the desired poles of the closed-loop system subject to complex pairing. Then the feedback gain vector K is given by the following equation,

$$K = -(0.....0,1).Q^{-1}.\ \Delta(A) \tag{2.11}$$

Here Q is a (nxn) controllability matrix of the controllable pair (A, b) and is defined as

$$Q = (b, Ab, ...A^{n-1}b) \tag{2.12}$$

and $\Delta(A)$ is the characteristic polynomial evaluated at λ=A. The equation (2.11) is called Ackermann's formula [24].

Under the feedback law as given by (2.9) the closed-loop system equation becomes,

$$\begin{aligned}\dot{x} &= (A+bK)x(t)\\ &= F.x(t)\end{aligned} \tag{2.13}$$

where F=(A+bK). Let $\Delta(\lambda)$ be the desired closed-loop characteristic polynomial of the closed-loop system matrix (A+bK)

$$\Delta(\lambda) = \det(\lambda I-(A+bK))$$
$$= \lambda^n + a_1 \lambda^{n-1} + \ldots\ldots + a_{n-1}\lambda + a_n \qquad (2.14)$$

Since the pair (A, b) is controllable, the controllability matrix Q, as given in (2.12), is invertible. It is possible to write the basic definition of the inverse of a matrix.

$$Q^{-1}.Q = I \qquad (2.15)$$

Let h denotes the last row of Q^{-1}. Then,

$$h(b, Ab, \ldots\ldots A^{n-1}b) = (0, 0, \ldots\ldots 1) \qquad (2.16)$$

which is equivalent to the following equalities:

$$h.b = h.(Ab) = \ldots\ldots\ldots\ldots = h(A^{n-2}b)=0$$
$$h(A^{n-1}b) = 1 \qquad (2.17)$$

using (2.17) we obtain the set of equations,

$$hF = h(A+bK) = hA$$
$$hF^2 = (hF)F = (hA)(A+bK)= hA^2$$
$$hF^{n-1}=(hF^{n-2})F=(hA^{n-2})(A+bK)=hA^{n-1}$$
$$hF^n=hA^n+K$$

Furthermore, from the Cayley-Hamilton theorem, we know that every matrix satisfies its own characteristic equation, i.e.,

$$\Delta F = F^n + a_1 F^{n-1} + \ldots\ldots + a_n I = 0 \qquad (2.19)$$

Multiplying (2.19) by h and using (2.18) we get ,

$$h\ \Delta(F) = h(A^n) + h(a_1 A^{n-1}) + \ldots\ldots + h(a_n I) + K = 0 \qquad (2.20)$$

solving for K we obtain,

$$K = -h\ \Delta(A) \qquad (2.21)$$

we have to note also the fact that h, the last row of Q^{-1}, can be written as,

$$h = (0, \ldots\ldots 0, 1)Q^{-1}$$

hence

$$K = -(0, \ldots\ldots 0, 1)Q^{-1}\ \Delta(A) \qquad (2.22)$$

In the computation of the feedback gain vector K, it is only required to calculate the last row of Q^{-1}, which saves much from computation time. Furthermore, even if there are multiple open-loop or closed-loop poles, the same theorem can be again applied without any modification which is not the case in most of the other pole assignment algorithms.

Although Ackermann's original procedure can only be applied to single input and completely state controllable system, the procedure is later modified [25], so that it can also be applied to partially controllable systems.

2.3.2 Modal Control For Single And Multi-Input Systems

In the continious-time domain consider multi-input system equation

$$\dot{x} = Ax + Bu \tag{2.23}$$

where A, x, B and u were defined previously.

The free response of the uncontrolled plant is given by linear combination of the dynamical modes of the system, where the mode shapes are determined by the eigenvectors and time-domain characteristics by the eigenvalues of the appropriate plant matrix A.

However, if the control loops are introduced which generate the input vector by linear feedback of the state vector of the plant, then the response characteristics of the resulting closed-loop system will no longer be determined by the eigen properties of A, but by those of some new closed-loop plant matrix whose eigenvectors and eigen values will depend upon the precise nature of the feedback loops [19].

The equation describing the dynamics of the system is given by equation (2.23). If a new state vector z(t) is introduced into equation (2.23) by the transformation

$$x(t) = Mz(t) \qquad (2.24)$$

where M is the modal matrix of A, then the new state equation has the form

$$M\dot{z} = AMz + Bu \qquad (2.25a)$$

it follows from equation (2.25a) that

$$\dot{z} = M^{-1}AMz + M^{-1}Bu \qquad (2.25b)$$

and therefore that

$$\dot{z} = \Lambda z + B_n u \qquad (2.26)$$

in view of equation (2.23), $\Lambda = M^{-1}AM$ is a diagonal (nxn) matrix of A, its rank is n and $B_n = M^{-1}B$ is the normalized input matrix

$$\Lambda = M^{-1}AM = \begin{bmatrix} \lambda_1 & 0 & . & 0 \\ 0 & \lambda_2 & . & 0 \\ & & & \\ 0 & 0 & . & 0 \\ . & . & . & . \\ 0 & . & & \lambda_n \end{bmatrix} = \text{diag}\ \{\lambda_1, \lambda_2, \ldots . \lambda_n\} \qquad (2.27)$$

and λ_i's are the eigenvalues of matrix A. The importance of equation (2.26) as compared with equation (2.23) is that Λ is a diagonal matrix whereas A is, in general, non-diagonal.

Notice that, the transformation matrix M, defined by into (2.24), modifies the coefficient matrix of z into the diagonal matrix. Notice also that the diagonal elements of the matrix $M^{-1}AM$ in (2.25b) are identical with the eigenvalues of A. It is important to note that the eigenvalues of A under a linear transformation, we must show that the characteristic polynomials $|\lambda I-A|$ and $|\lambda I-M^{-1}AM|$ are identical [18].

Since, the determinant of a product is the product of the determinants, we obtain,

$$
\begin{aligned}
|\lambda I-M^{-1}AM| &= |\lambda M^{-1}M-M^{-1}AM| \\
&= |M^{-1}(\lambda I-A)M| \\
&= |M^{-1}||\lambda I-A||M \\
&= |M^{-1}||M||\lambda I-A|
\end{aligned}
$$

noting that the product of determinants $|M^{-1}|$ and $|M|$ is the determinant of the product $|M^{-1}.M|$. We obtain,

$$
\begin{aligned}
|\lambda I-M^{-1}AM| &= |M^{-1}M||\lambda I-A| \\
&= |\lambda I-A|
\end{aligned}
$$

Thus, it has been proved that the eigenvalues of A are invarient under a linear transformation.

Since, let's apply control law (2.9) into equation (2.26)

$$\dot{z} = \Lambda z+B_nKx$$

$$\dot{z} = (\Lambda + B_n KM)z \tag{2.28}$$

The equation (2.28) gives the open-loop poles of system. Suppose the desired closed-loop poles are specified by ($\Lambda = \Lambda$). Then

$$\dot{z} = \Lambda' z \tag{2.29}$$

in the equation (2.29), Λ contains the closed-loop poles which is a diagonal (nxn) matrix. Then by combining equation (2.28) and (2.29), we obtain

$$\Lambda + B_n KM = \Lambda'$$

$$K = B_n^{-1} (\Lambda' - \Lambda) M^{-1} \tag{2.30}$$

In equation (2.30) B_n is not a square matrix, hence its inverse can not be calculated. In order to determine K, the so called Pseudo-Inverse method is used,

$$K = (B_n^T B_n^{-1})^T \; B_n \; (\Lambda' - \Lambda) M^{-1} \tag{2.31}$$

Meanwhile, it is possible to obtain canonical form using modal analysis for single input systems which is developed by Wonham and Johnson [26]. It is not mentioned in this chapter.

In the modal analysis, M is the modal matrix of the system, usually in complex form. In this case, it is required the more computational effort to obtain inverse of the complex modal matrix M. At the same time the desired closed-loop eigenvalues must be in the same form with the open-loop eigenvalues for obtaining feedback matrix which only contains real part. This is the another restriction for modal analysis.

2.3.3 Direct Design Procedure For Single-Input Systems

The method proposed is based on the equivalence of the closed-loop characteristic polynomial of a multi-input and a corresponding single-input system. The latter is first designed using the previously established direct design method for single input system and the result is then transferred back to the multi-input case.

The method has a number of attractive features. It is computationally very fast and is well suited to the computer-aided design of control system. It provides the designer with complete freedom over the relative tightness of the feedback to each input and hence also allows the design with feedback to only some inputs, i.e., incomplete input feedback. A further important feature is that it permits design with incomplete state feedback, when some of the states are not accessible.

Since the method involves the use of the existing procedure for single-input systems, this is first summarised [27].

Consider a controllable single-input system described by equation (2.8). The transfer function representation of equation (2.8) is

$$X(s) = G(s)U(s) \tag{2.32}$$

where $G(s)=(sI-A)^{-1}b=\{g_i(s)\}/F(s)$, $i=1,\ldots,n$, $G(s)$ is the nx1 open-loop transfer-function matrix where $F(s)$ the characteristic polynomial of the open-loop system, $F(s)=|sI-A|$.

If the feedback law (2.9) is applied, then the transfer function representation of the closed-loop system becomes

$$X(s) = G_c(s)V(s) \tag{2.33}$$

where $G_c(s)=(sI-A-bK)^{-1}b=\{g_i(s)\}/H(s)$, $i=1,\ldots,n$, is the nx1 closed-loop transfer function matrix from $V(s)$ to $X(s)$ and $H(s)$ is the characteristic polynomial of the closed-loop system, $H(s)=|sI-A-bK|$.

It has been shown that

$$\sum_{i=1}^{n} K_i g_i(s) = H(s) - F(s) \tag{2.34}$$

i.e., the scalar product of the feedback and the numerator transfer function vectors is equal to the difference of the characteristic polynomials of the closed-loop and open-loop system. This direct relationship between the feedback vector and the closed-loop poles establishes a direct design method whereby the feedback required to shift the open-loop poles to desired closed-loop positions can readily be calculated.

The feedback vector K is simply calculated by equating coefficients of like powers of s in equation (2.34).

2.3.4 Direct Design Procedure For Multivariable Feedback Systems

Consider a controllable multi-input system described by equation (2.23). The design problem is to find the mxn state-feedback matrix K such that the closed-loop system described by equation (2.23) and the feedback law u=Kx has a prescribed behaviour characterised by n given closed-loop system poles $\lambda_1, \lambda_2, \ldots\ldots \lambda_n$.

Now, the closed-loop system poles are the roots of the characteristic equation

$$H(s) = |sI-A-BK| \qquad (2.35)$$

If we set K=qp, where q is an m-column vector and p is a n-row vector, equation (2.35) can be rewritten as

$$|sI-A-Bqp| = 0$$

or

$$|sI-A-bp| = 0 \qquad (2.36)$$

where b=Bq is an (nx1) matrix.

Comparison of equation (2.35) and (2.36) yields the followings:

The closed-loop poles of a multi-input system which has a plant matrix A, a control matrix B and a feedback matrix K are coincident with those of an equivalent single-input system which has the same plant matrix A, a control matrix b and the feedback vector p, where b=Bq and K=qp.

Making use of this equivalence, the design problem can be solved in the following steps:

i- Choose an m-dimensional vector q. In general, q is arbitrary except for special cases.

ii- Find the n-dimensional feedback vector p required to position the poles of the equivalent single input system (A, Bq) at the desired location $\lambda_1, \lambda_2, \ldots \lambda_n$, using the single-input direct design precedure based on equation (2.34).

iii- For the multi-input system (A, B) the required state feedback matrix K=qp.

2.4 Phase-Variable Canonical Form For Eigenvalue Assignment

2.4.1 Introduction

The development of the phase-variable canonical form for single-input linear controllable systems has been an active area of research [14], [15]. Partly this is because the phase-variable form has proved to be an extremely convenient starting point for certain control design problems and partly it is because canonical forms are mathematically intriguing in their own right.

Unlike the single-variable case, the corresponding canonical forms for multivariable systems are not unique. This lack of uniqueness not only tends to make their derivations more difficult but also forced the design engineer faced with a practical application to determine the best form from the several possibilities.

Now in this chapter, we are going to introduce a transformation which is examined in [21] to [23], [26] and [28] to [30], so that the transformed state equations will be in phase-variable canonical form. The use this form in pole assigment problem will be discussed and illustrated in detail. The derivation given here, however, is more general and notationally simpler since the computations are expressed in terms of matrix algebra whenever possible.

2.4.2 Time - Variable Controllability Matrix In Canonical Form For Single - Input Systems.

Consider the problem of transforming to equivalent canonical (phase-variable) form of the system

$$\dot{x} = Ax + bu \qquad (2.38)$$

where x is a n dimensional state vector, u is a scalar input function, A and b time variable matrices of appropriate order.

The phase-variable form is one of the several usuful canonical system representations and it is defined as

$$\dot{x} = \hat{A}x + \hat{b}u \tag{2.39}$$

where x is a n dimensional state vector.

$$\hat{A} = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ & & & & \\ 0 & 0 & 0 & \cdots & 1 \\ -a_n & -a_{n-1} & & \cdots & -a_1 \end{bmatrix} \qquad \hat{b} = \begin{bmatrix} 0 \\ 0 \\ \cdot \\ \cdot \\ 0 \\ 1 \end{bmatrix} \tag{2.40}$$

The coefficient a_i (in general time-variable) completely characterizes (2.39) and will be represented by the n vector whose i-th element is a_i.

The system described in (2.38) is said to be equivalent to a system of the form (2.39) if and only if, a non-singular continuously differentiable matrix T exists such that z = Tx. In the fixed case the necessary and sufficient condition for such an equivalence to exist is that the system of equation (2.38) be completely controllable [30].

Before we obtained the system equivalence problem, several properties of the controllability matrix of a time-variable system will be reviewed. The controllability matrix of the system in (2.38) is defined as

$$Q = [\ P_0, P_1, \ldots\ldots P_{n-1}] \qquad (2.41)$$

where

$$P_{K+1} = -AP_K + d/dt.P_K \quad , \quad P_0 = b$$

the controllability matrix $\hat{Q}$ of the system of (2.40) is defined similarly and the system of (2.38) is uniformly controllable if Q has rank n everywhere is proved at theorem (2.1).

Matrices $\hat{Q}$ and $\hat{P}_n$ will now be examined more closely, for it will be shown that they serve to determine the transformation form (2.38) to (2.40) when it exists. It can be verified by direct construction that

$$\hat{Q} = \begin{bmatrix} 0 & 0 & & 0 & (-1)^{n-1} \\ 0 & 0 & & (-1)^{n-2} & q_{n-1,n-1} \\ . & . & & . & . \\ . & . & & . & . \\ . & . & & . & . \\ 0 & -1 & \ldots\ldots & q_{n-2,2} & q_{n-1,2} \\ 1 & q_{11} & \ldots\ldots & q_{n-2,1} & q_{n-1,1} \end{bmatrix} \qquad \hat{P}_n = \begin{bmatrix} q_{n,n} \\ q_{n,n-1} \\ . \\ . \\ . \\ q_{n-1,2} \\ q_{n-1,2} \end{bmatrix} \qquad (2.42a)$$

where

$$\begin{aligned} q_{iK} &= -q_{i-1,K-1} + q_{i-1,K} & & 1<K<i\le n \\ &= (-1)^i a_{n-i+1} - \sum_{j=0}^{i-2} a_{n-j} q_{i-1,j+1} + \dot{q}_{i-1,1} & & K=1<i\le n \\ &= (-1)^i a_n & & 1\le K=i\le n \qquad (2.42b) \end{aligned}$$

from the form of $\hat{Q}$ it is clear that any system of the form (2.40) is uniformly controllable. A more informative relation or q_{iK} can be easily derived from (2.42b) as

$$q_{iK} = (-1)^i a_{n-i+K} + (-1)^K \sum_{j=0}^{i-K+1} a_{n-j} q_{i-K,\,j+1} + \sum_{j=0}^{K} (-1)^{j+1} q_{i-j,\,K-j+1} \qquad 1 \le K < i \le n \qquad (2.43)$$

and

$$q_{ii} = (-1)^i a_n \qquad 1 \le i \le n$$

it follows by a simple introduction argument that

$$q_{iK} = (-1)^i a_{n-i+K} + \begin{bmatrix} \text{terms involving only the} \\ \text{coefficients } a_n, \ldots a_{n-j+K+1} \end{bmatrix} \qquad 1 \le i \le K \le n \qquad (2.44)$$

For notational convenience, the bracketed expression in (2.44) will be represented by the symbol θ_{i-K}. That is, any function that can be expressed solely in terms of the coefficients $a_n, \ldots a_{n-r+1}$ will be replaced by the symbol θ_r wherever no other information about the function is needed. With this notation equation (2.44) becomes

$$q_{iK} = (-1)^i a_{n-i+K} + \theta_{i-K} = \theta_{i-K+1} \qquad 1 \le i \le K \le n$$

$$\theta_0 = 0 \qquad (2.45)$$

THEOREM (2.3): The system in (2.38) is equivalent to a system of the form in (2.40) if and only if (2.38) is uniformly controllable.

The necessity of the controllability condition is easily established, since if the equation (2.38) is equivalent to the equation(2.40) where $z = Tx$,

$$\hat{Q} = TQ \qquad (2.46)$$

But $\hat{Q}$ and T have rank n everywhere, therefore Q must have rank n everywhere, which implies that the system (2.38) is uniformly controllable.

If the system (2.38) is uniformly controllable, the matrix

$$T = \hat{Q}.Q^{-1} \qquad (2.47)$$

is non singular when $\hat{Q}$ is the controllability matrix of any system of the form (2.40). Moreover, (2.45) shows that (2.47) must be the form of the transforming matrix if it exists [30]. Thus, to prove that the uniform controllability condition is sufficient, let $z=Tx$ where T is given by (2.47). In other words, the nxn matrix T is obtained from controllability matrix $\hat{Q}$ by setting t_1, the first row of T, equal to the last (n-th) row of Q^{-1} and recursively computing the remaining rows of T by succesive post multiplication of each preceding row of T by A. In particular,

$$T = \begin{bmatrix} t_1 \\ t_1 A \\ \cdot \\ \cdot \\ \cdot \\ t_1 A^{n-1} \end{bmatrix} \tag{2.48}$$

where t_1 is the n-th row of G^{-1}. it is thus readily apparent that

$$t_1 b = t_1 Ab = \ldots\ldots\ldots t_1 A^{n-2} b = 0$$

but (2.49a)

$$t_1 A^{n-1} b = 1$$

which immediately implies the relation

$$Tb = \left[0, 0 \ \ldots\ldots\ldots 1\right]^T \tag{2.49b}$$

if z is defined as Tx, it is seen that the first element of z, namely z_1, when differentiated with respect to time, yields the relation (dropping the time arguments for convenience)

$$\dot{z}_1 = (t_1 A)x + (t_1 b)u \tag{2.50a}$$

which in turn equal to $z_2 = t_2 x$. Furthermore,

$$\dot{z}_2 = (t_1 A^2)x + (t_1 Ab)u = z_3 \tag{2.50b}$$

and so forth, or in genaral

$$\dot{z}_i = z_{i+1} \quad , \quad i = 1, 2, \ldots\ldots (n-1) \tag{2.50c}$$

therefore, it follows that the equivalent single-input system representation. $(\hat{A}, \hat{b})$ or $\dot{z} = \hat{A}z(t) + \hat{b}u(t)$ where $\hat{A} = T.A.T^{-1}$ and $\hat{b} = T.b$ is in a particular structural canonical form (2.40).

Some immediate benefits are derived from the reduction of (A, b) to controllable canonical form. In particular the characteristic polynomial, det($\lambda I-A$), of the system is apperant from the last row of $\hat{A}$. Expanding the det($\lambda I-\hat{A}$) along any but the last row, we obtain the characteristic polynomial of the pair (A, b) or ($\hat{A}, \hat{b}$), i.e.,

$$\Delta(\lambda) = \det(\lambda I-A) = \det(\lambda I-\hat{A})$$
$$= \lambda^n + a_1 \lambda^{n-1} + \ldots\ldots a_{n-1}\lambda + a_n \tag{2.51}$$

furhermore, the input u only effects the last row of $\hat{b}$, due to its special structure obtained through the transformation z = Tx.

2.4.3 Extension Of Controllable Canonical Form To Multivariable System.

Consider a system governed by the set of first order differential equation :

$$\dot{x} = Ax + Bu \tag{2.52}$$

where

x(t) is a (nx1) state vector, u(t) is a (mx1) input vector, A is a (nxn) matrix and B is an (nxm) input matrix.

The notation of controllable canonical form is not confined only to scalar systems and can be extended to more general multivariable cases. In particular, consider any completely state controllable system pair (A, B), with B assumed to be of full rank $m \leq n$. This later assumption implies that all m available inputs are mutually independent, which usually is the case in practice.

The fundamental assumption imposed on the system is that of system controllability i.e., it is assumed that the (nxnm) controllability matrix

$$Q = [\ B, AB, \ldots\ldots\ldots A_{n-1}B] \tag{2.53}$$

has rank n. In addition, it is generally assumed that the m columns of b are linearly independent.

The controllability index μ_c of the system (2.52) is defined as the smallest positive integer for which the matrix.

$$Q_{\mu} = [\ B, AB, A^2B \ldots\ldots\ldots A^{\mu-1}B] \tag{2.54}$$

has rank n.Generally, for multivariable controllable systems $\mu_c \leq n$.

Canonical forms for the system (2.52) are constructed by transforming to state vector to a new coordinate system in which the system equations take a particular form. The transformation employed to affect the coordinate change is essentially always constructed from independent columns of the controllability matrix (2.53).

The first step in the development of a canonical form of the class discussed in this part, is the section of n linearly independent vector from the n.m columns of this controllability matrix (2.53). It will be required that the selection procedure be so devised that the n chosen linearly independent vectors comprise the columns of a matrix P of the form.

$$P=\left[b_1, Ab_1, \ldots A^{p1-1}b_1, b_2, Ab_2, \ldots A^{p2-1}b_2, b_m, \ldots A^{pm-1}b_m\right] \quad (2.55)$$

The esssential restriction, then, is that no vector of the form $A^k b_j$ is selected unless all lower powers of A times b_j are also selected.

2.4.4 Selection of Independent Vectors Controllable Canonical Form

As will be shown below, it is possible to make a selection of the required form, but in general, it is not unique. The real difficulty is in determining which of many possible P matrices leads to the best canonical form.

The selection of the vectors comprising the P matrix is straight forward (but still somewhat arbitrary) if it is done according to the following precedures.

Scheme I: Search the linear independent vectors.

1- Select one of the columns of B (without loss of generality be assumed that b_1 is selected).

2-Select either another column of B (say b_2) or the vector Ab_1. If the selected vector is linearly independent of b_1, retain it, otherwise omit it from the selection.

3- At any stage of the process, select the new vector to be of the form $A^j b_K$ where all lower powers of A times b_K have already been retained. If the new vectors is linearly independent of all previously selected vectors, retain it, otherwise omit it from the selection.

4- The selection process terminates when n linearly independent vector are found. Arrange the n vectors in their proper order to form the matrix P [27].

Scheme II: Search the crate by columns.

We first select b_1 and indicate this by putting an x in the $A^0 b_1$ cell. Now if Ab_1 is linearly independent of b_1, we put an x in this cell as well and continue down the first column of the crate until we either put x in all the cells or we find a vector, say $A^{l_1} b$, that is linearly independent on the earlier vectors in the column [31]. We denote this fact by putting 0 in the corresponding $(l_1, 1)$, then note (by a now familiar argument) that when this

b_1	b_2	b_3	b_4	
X	X	X	0	$A^0=I$
X	X	0		A
X	0			A^2
X				A^3
0				A^4
				A^5
				A^6

Table 2.4.1 Typical crate diagram filled in by searching by columns. We have $l_1=4$, $l_2=2$, $l_3=1$, $l_4=0$

happens all the remaining vectors in those columns will be linearly independent on the previously selected vectors. We indicate this by leaving the corresponding cells blank. If we have not found n linearly independent elements in the first column, we go to the second column. If b_2 is linearly independent of all previously selected vectors $\{b_1, Ab_1, \ldots A^{p_1-1}b_1\}$, we put an x in the corresponding cell. Now repeat this procedure with Ab_2 and continue in this way with succesive columns if necessary until n linearly independent vectors have been found. With this scheme, the crate diagram will have the general form shown in table (2.1). The cell with 0's correspond to the vectors $\{A^{l_i}b_i,\ i=1,\ldots m\}$. The pattern depends on the order in which the inputs are arranged, since the tendency is to have a few long chains of x's and not all inputs may be called upon. A more uniform treatment of the system inputs is provided by another natural search procedure.

Scheme III: Search the crate by rows.

Now we search the rows until we find a vector, say $A^{k_i}b_i$, that is linearly dependent on all the previously selected vectors. We put a 0 in the corresponding cell and note again that all vectors below it in the same column will also be linearly dependent on the already-selected vectors. Therefore we leave all the corresponding cells blank and go on, if necessary to the next linearly independent vector encountered in the row search (18). (We may remark that searching the crate by rows corresponds to searching the columns of the controllability matrix from left to right). A typical crate diagram produced by this scheme will

b_1	b_2	b_3	b_4	
X	X	X	0	$A^0=I$
X	0	X		A
X		X		A^2
0		0		A^3
				A^4
				A^5
				A^6

Table 2.4.2 Typical crate diagram filled in by searching by rows. We have $k_1=3$, $k_2=1$, $k_3=3$. The set of length {3, 1, 3} will be the same even if the order of the $\{b_1\}$ is permuted.

appear as in Table (3.2). The tendency now is to have several chains of nearly equal lengths $\{K_1, \ldots\ldots, K_m\}$. It can be shown that the length we get here will remain the same, even if the columns are permuted.

Scheme IV: Search the possible indices .

Let us define the controllability matrix (2.53) as the (nxn) matrix obtained by selecting from left to write as many as n linearly independent columns of the controllability matrix (2.53). Since the system (2.52) is assumed to be controllable if Q (2.53) has full rank n, we can construct the nonsingular (nxn) matrix P (2.55) by simply reordering the n(=n) columns of Q (2.53), beginning with a power ordering of the first P_1 columns of Q (2.53) which involves b_1 is first column of B, and then employing those p_2 columns of Q (2.53) which involve b_2, next and so forth [32]. Now we can define the m integers P_i as the controllability indices of the system and denote by μ, $Max(p_i)$ for $(i=1, \ldots\ldots, m)$, which we further define as the controllability of the system i.e., $max(d_i)=\mu$. It should now be noted that all m columns of B are present in P since we assumed that B was full rank m. We now set

$$K_i = \sum_{j=1}^{i} P_j \quad , \quad i=1, \ldots\ldots m \qquad (2.56)$$

which implies that

$$k_1 = p_1$$

$$k_2 = p_1+p_2 \qquad (2.57)$$

$$k_3 = p_1+p_2+..........+p_m=n$$

It is shown in Appendix I that this process does not terminate before R independent vectors have been selected. It may happen that, as a result of the selection scheme I , not all columns of the original B matrix occuring the P matrix. In this case, the corresponding input components play no special role in he associated canonical forms and will appear in an arbitrary fashion in the final result. The other input components enter the canonical forms and will input components enter the canonical system in a special way.

Although there is a certain amount of freedom in the selection process, there are two specific plans for selection that have special interest. In the first plan, one starts with the vectors b_1 and then proceeds to Ab_1, A^2b_1, ...$A^{n-1}b_1$ is obtained. In this case the system is controllable from the first input alone, or until a dependency arises. If more independent vectors are required, one then selects b_2, Ab_2, ... until a dependency arises. The prodecure continues in this manner through the b_K's until n linearly independent vectors are obtained. The tendency is to develop a few long chains in this case. The P matrix (2.55) obtained in this fashion has the property that

$A^{PK}b_K$ is linearly dependent on vectors of the form $A^1 b_i$ with $i \le K$ in the scheme I.

The crate is a table with m columns in table (3.1) and table (2.40), representing the columns of input matrix B and n rows corresponding to the power of A matrix; the (i, j)-th cell of the crate then represents the column of vector $A^{i-1}b_j$, and choosing n linearly independent columns of P matrix (2.55).

Scheme IV is used to solve our problem which is very convenient in multi-input system for computer computation. It will be discussed in detail.

2.4.5 Canonical Forms Of The System Matrix

In a part (2.4.4) it has been concerned about development of transformation to put the system under consideration into controllable canonical form. This particular canonical form was then used to develop a simple design prodecure by many authors [26]-[32]. A construction prodecure for the required transformation, for the general case will be presented in this part.

A change of the coordinates from state vector x to z defined by z = Tx transforms the system (2.52) which becomes.

$$\dot{z} = T.A.T^{-1} z + TBu \qquad (2.58)$$

Appropriate choices of T lead to canonical forms of the system (2.52).

Two basic canonical forms are developed from the matrix P (2.55) constructed in this part. Of course there are possible variations within each of two basic forms since there are possible variations in the choice of P. Each choice of P, however, leads to two basically different canonical forms.

Scheme I

The first canonical form is produced by setting $T = P^{-1}$ simple matrix bookkeeping verifies that the system is then transformed to the form

$$\dot{z} = \hat{A}z + \hat{B}u \tag{2.59}$$

$$\hat{A} = \begin{bmatrix}
0 & 0 \ldots x & & & x & & x \\
1 & 0 \ldots x & & & x & & x \\
\vdots & \vdots & & & \vdots & & \vdots \\
0 & \ldots 1 \; x & & & \vdots & & \vdots \\
 & x & 0 & 0 \ldots & x & & \vdots \\
 & x & 1 & 0 \ldots & x & & \\
 & \vdots & \vdots & \vdots & \vdots & & \\
 & & & 0 \ldots & 1 \; x & & \\
 & & & & x & & \vdots \\
 & & & & x & & \vdots \\
 & & & & \vdots & & x \\
 & & & & & & x \\
 & & & & & & 0 \ldots x \\
 & & & & & & 1 \ldots x \\
 & & & & & & \vdots \\
 & & & & & & 0 \ldots 1 \; x
\end{bmatrix}
\qquad
\hat{B} = \begin{bmatrix}
1 & 0 \ldots & 0 \\
0 & 0 & \vdots \\
\vdots & \vdots & \vdots \\
\vdots & 1 & \\
 & 0 & \\
 & \vdots & \\
 & & 1 \\
 & & 0 \\
 & & \vdots \\
 & & 0
\end{bmatrix} \tag{2.60}$$

The system may be considered as composed of fundamental companion matrices located in blocks along the diagonal. The x's in the matrix represent possiple nonzero elements and except for the indicated 1's, these occur only in the columns corresponding to the right-hand edge of a fundamental companion matrix.

Different choices of P lead to different sizes and number of companion matrices as well as different values for the nonzero elements. If P were chosen according to the first special plan of the last section, the x's in a given column of A would be zero below the companion matrix corresponding to the column. Each of the companion matrices can be considered to present a subsystem coupled to other systems. For the special choice of P mentioned above, the coupling between two subsystem is in one direction only.

Scheme II

The second kind of canonical form is more useful than the first but is somewhat more diffucult to derive.

Again start with a P matrix of the form (2.55). Write P^{-1} in terms of its vectors.

$$P^{-1} = \begin{bmatrix} e_{11} \\ e_{12} \\ \cdot \\ \cdot \\ \cdot \\ e_{1p1} \\ e_{21} \\ \cdot \\ \cdot \\ e_{mpm} \end{bmatrix} \tag{2.61}$$

Actually, only the m's of these rows play a direct role in the canonical form. These are the last rows of each of the m groups of rows, i.e., the (row) vectors $e_{i\mu i}$, $i=1,2,\ldots m$. For simlicity of notation these vectors are now labelled as

$$e_i = e_{i\mu i} \tag{2.62}$$

The vectors $e_1, e_2, \ldots e_m$ are used to construct the transformation matrix.

$$T = \begin{bmatrix} e_1 \\ e_1 A \\ \cdot \\ \cdot \\ \cdot \\ e_1 A^{p1-1} \\ e_2 \\ e_2 A \\ \cdot \\ \cdot \\ e_m A^{pm-1} \end{bmatrix} \tag{2.63}$$

It is shown in Appendix II that T defined by (2.43) is nonsingular.

It is again a simple matter of bookkeeping to verify that the transformation T defined by (2.43) reduces the system (2.52) to the form (2.43) where now,

$$\hat{A}=T.A.T^{-1}=\begin{bmatrix} \hat{A}_{11} & \hat{A}_{12} & \cdots\cdots & \hat{A}_{1m} \\ \vdots & & & \vdots \\ \hat{A}_{m1} & \hat{A}_{m2} & \cdots\cdots & \hat{A}_{mm} \end{bmatrix} \tag{2.64}$$

$$\hat{A}=\left[\begin{array}{cccccccccccccccccc}
0 & 1 & 0 & & 0 & & & & & & & & & & & & & \\
0 & 0 & 1 & & 0 & & & & & & & & & & & & & \\
 & & & & 1 & & & & & & & & & & & & & \\
x & x & .. & & x & x & x & x & & & & & & & & & & \\
 & & & & & 0 & 1 & 0 & & 0 & & & & & & & & \\
 & & & & & 0 & 0 & 1 & & 0 & & & & & & & & \\
 & & & & & & & & & 1 & & & & & & & & \\
x & x & x & \ldots & & x & x & & \ldots & x & x & & . & . & . & & & \\
x & x & & & & & \ldots & \ldots & & & & x & x & x & x & x & \ldots & \\
 & & & & & & & & & & & & & & 0 & 1 & \ldots & 0 \\
 & & & & & & & & & & & & & & \vdots & & & \\
 & & & & & & & & & & & & & & . & & & 1 \\
x & x & & & & & & \ldots & \ldots & & & x & x & x & x & x & .. & x
\end{array}\right] \tag{2.65a}$$

$$\hat{B}=\begin{bmatrix} 0 & 0 & \ldots\ldots & 0 \\ 0 & 0 & \ldots\ldots & 0 \\ \cdot & & & \\ \cdot & & & \\ 1 & x & \ldots\ldots & x \\ 0 & 0 & \ldots\ldots & 0 \\ 0 & 0 & \ldots\ldots & 0 \\ \cdot & & & \\ \cdot & & & \\ 0 & 1 & x \ldots & x \\ & & & \cdot \\ & & & \cdot \\ & & & 0 \\ & & & \cdot \\ 0 & \ldots & 0 & 1 \end{bmatrix} \tag{2.65b}$$

The m diagonal blocks $\hat{A}_{ii}$ of $\hat{A}$ are each an upper right identity companion matrix of dimension p_i while the off diagonal blocks, $\hat{A}_{ij}$ for $i=j$ are each identically zero except possibily for their respective final rows. We therefore note that all information regarding the equivalent state matrix $\hat{A}$ can be derived from knowledge of the m ordered controllability indices p_i and m ordered k rows of $\hat{A}$. The same can also be said of $\hat{B}$, since we note that only these same ordered k rows of $\hat{B}$ are nonzero.

2.4.6 Extension Of Controllable Campanion Form To Partially State Controllable Systems.

We can consider certain implication and extensions of the proceding results when the multivariable system is only state controllable. In particular, we still assume that rank $(B)=m<n$ but we consider the case when rank $(Q)=\bar{n}<n$. Note that it is still possible to define the $(n x \bar{n})$ matrix Q consisting of the first $\bar{n}$ linearly independent columns of Q, as well as the $(n x \bar{n})$ matrix P as given by (2.55) but with $K_i=\sum_{j=1}^{m} P_j = \bar{n}$ instead of n. The $\bar{n}$ linearly independent columns of P clearly form a basis of some subspace W of E^n. If we define W_1 as the orthogonal complement of i.e., the subspace of E^n consisting of all vectors in the sense of a zero inner product, it follows that the any vector v in E^n can be expressed as a linear combination of the same vector w in W and some vector w_1 in W_1. In particular, $v=\alpha w+\beta w_1$ for all v in E^n which implies that E^n can be defined as the direct sum of W and W_1. It is thus clear that the dimension q of W_1 is $n-\bar{n}$, since E^n is of dimension n. We let $\beta_1, \beta_2, \ldots, \beta_q$ be any basis of W_1 and consider the extended state representation

$$\dot{x} = Ax + B_e u_e \tag{2.66}$$

where B_e is the $nx(m+q)$ matrix obtained by appending to B the q basis vectors of W_1, i.e, $B_e=[B, \beta_1, \ldots, \beta_q]$ while u_e is an $((m+q)x1)$ input vector obtained by appending to u, q additional input elements i.e. $u_e=[u_1, \ldots, u_m, u_{m+1}, \ldots, u_{m+q}]^T$. The extended system (3.39), thus defined is clearly a controllable one and is therefore possible to employ the algorithm presented earlier, to obtain a n dimensional equivalence transformation which reduce the extended system to controllable companion form. We denote the appropriate transformation matrix T_e and utilize it to reduce the original system to be equivalent representation $\dot{z}=\hat{A}z+\hat{B}u$, where $\hat{A}=T_e.A.T_e^{-1}$ and $\hat{B}=T_e.B$. Due to the specific choice of T_e, it follows that the equivalent pair $(\hat{A},\hat{B})$ partially resembles the multivariable companion form. In particular

$$\hat{A}=\begin{bmatrix} \hat{A}_c & \vdots & \hat{A}_{c\bar{c}} \\ \cdots & \vdots & \cdots \\ 0 & \vdots & \bar{A}_{c} \end{bmatrix} \qquad \hat{B}=\begin{bmatrix} \hat{B}_c \\ \cdot \\ \cdot \\ 0 \end{bmatrix} \qquad (2.67)$$

where the pair $(\hat{A}_c,\hat{B}_c)$ is the $\bar{n}$ dimensional controllable companion form, i.e., the pair $(\hat{A}_c,\hat{B}_c)$ assumes the structure indicated by (3.38) with $k_i=\sum_{j=1}^{m} P_j = \bar{n}$. Furthermore the lower left $(qx\bar{n})$ block of $\hat{A}$ as well as the final q row of $\hat{B}$ are idenitically zero. On closer inspection it becomes apparent that the controllable and the completely uncontrollable portion of the system have been separated. More specifically, the $\bar{n}$ dimensional subsystem defined by the first $\bar{n}$ rows of the pair $(\hat{A},\hat{B})$ namely $\dot{z}_c=\hat{A}_c z_c+\hat{A}_{c\bar{c}} z_{\bar{c}}+\hat{B}_c u$ is clearly controllable, since $\hat{A}_{c\bar{c}} z_{\bar{c}}$ can be treated as a known disturbance. Furthermore, the q-dimensional subsystem defined by the remaining rows of $(\hat{A},\hat{B})$, namely $\dot{z}_{\bar{c}}=A_{\bar{c}} z_{\bar{c}}$ is completely uncontrollable. We further note that in view of (2.51) and (2.67) the characteristic polynomial $\det(\lambda I-A)$ of A (and hence of $\hat{A}$) can be written as the product of the characteristic polynomials of the controllable and completely uncontrollable portion of the system, i.e.,

$$\det(\lambda I-A) = \det(\lambda I-\hat{A}) = \det(\lambda I-\hat{A}_c)\det(\lambda I-\hat{A}_{\bar{c}}) \qquad (2.68)$$

2.4.7 Pole Assignment Via The Controllable Companion Form

We will now consider the general employment of linear state feedback for arbitrary assignment of the closed-loop of the multivariable system as given in (2.52). In particular if the linear state variable feedback control law

$$u(t) = Kx(t) + v(t) \qquad (2.69)$$

is employed to alter the pole configuration of the open-loop system, we can readily obtain a state space representation for dynamical behaviour of the compensation system by simply substituting (2.69) for u into (2.68):

$$\dot{x} = (A+BK)x + Bv \qquad (2.70)$$

In general it is not all clear what effect the control law (2.69) has on the system (2.68), since consider any arbitrary unstructured open-loop system pair (A, B). However, if the open-loop system is in controllable companion form, the effect of the feedback law in (2.69) on pole locations can be easly clarified. Let us give a main result of this section as a theorem.

THEOREM (2.4): Consider the system (2.52) and the linear state variable feedback law (2.69). All $\bar{n}$ controllable poles of the closed-loop system (2.70) can be completely and arbitrary assigned via linear state variable feedback while the n-$\bar{n}$ uncontrollable poles of the system are uneffected by (2.69).

PROOF : Assume that we have already transformed the given system into controllable companion form (2.67). The pair $(\hat{A}_c, \hat{B}_c)$ is an $\bar{n}$-dimensional controllable companion form, while $\bar{A}_c$ represents the completely uncontrollable portion of the state matrix. As we have previously noted all (m) k_i rows of $\hat{A}_c+\hat{B}_c\hat{K}_c$ can be completely and arbitrarily altered via $\hat{K}$. ($\hat{K}$ is the required feedback gain matrix in the transformed coordinate system and $\hat{K}_c$ is the portion of $\hat{K}$ corresponding to the $\bar{n}$-dimensional controllable system $(\hat{A}_c, \hat{B}_c)$. We can choose the first $\bar{n}$ columns $\hat{K}_c$ of $\hat{K}$, such that

$$\hat{A}_c+\hat{B}_c\hat{K}_c = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ & & \vdots & & \\ 0 & 0 & 0 & \cdots & 1 \\ -a_n & -a_{n-1} & \cdots & \cdots & -a_1 \end{bmatrix} \tag{2.71}$$

is an n-dimensional companion matrix, where the scalers

$a_1, \ldots, a_{\bar{n}}$ represent the coefficients of the desired characteristic polynomial, i.e., the coefficients of the polynomial $\det(\lambda I-\hat{A}_c-\hat{B}_c\hat{K}_c)$. Since the remaining $n-\bar{n}$ columns of $\hat{K}$ affect only $\hat{A}_{c\bar{c}}$, the final $n-\bar{n}$ rows of $\hat{A}$ are completely uneffected by $\hat{K}$, which implies that the $n-\bar{n}$ eigenvalues $\hat{A}_{\bar{c}}$, or equivalently the uncontrollable poles of the system, remain unaltered by linear state feedback. This follows formally from the fact that all the n poles of the closed-loop system are equivalent to the zeros of :

$$\det(\lambda I-A-BK) = \det(\lambda I-\hat{A}-\hat{B}\hat{K})$$

$$= \det(\lambda I-\hat{A}_c-\hat{B}_c\hat{K}_c)\det(\lambda I-\hat{A}_{\bar{c}}) \qquad (2.72)$$

In order to explicitly determine a $\hat{K}$ which yields the controllable part of the closed loop system matrix as represented by (2.71), we let A_m^* denote the m ordered k_i rows of $\hat{A}_c+\hat{B}_c\hat{K}_c$ as given by (2.71) and define $\hat{A}_{cm}$ and $\hat{B}_{cm}$ as the same ordered k_i rows of $\hat{A}_c$ and $\hat{B}_c$, respectively. It therefore follows that

$$A_m^* = \hat{A}_{cm} + \hat{B}_{cm}\hat{K}_c \qquad (2.73)$$

or that the control law (2.69), with the first n columns of $\hat{K}$ given by

$$\hat{K}_c = B_{cm}^{-1}(A_m^* - \hat{A}_{cm}) \qquad (2.74)$$

yields the desired $\bar{n}$-dimensional closed-loop system submatrix (2.71).

The final $n-\bar{n}$ column of $\hat{K}$ play no part in closed-loop pole assignment, since they affect only $\hat{A}_{c\bar{c}}$ which in turn, has no effect on the eigenvalues of the closed-loop system matrix. We can therefore say the final $n-\bar{n}$ columns of $\hat{K}$ equal to zero in order to complete our assignment of all (mn) entries of an appropriate. The state feedback gain matrix K, associated with the original system is given by

$$u = \hat{K}z+v = \hat{K}Tx + v= Kx + v \qquad (2.75)$$

where

$$K = \hat{K}T \qquad (2.76)$$

III. THEORETICAL ANALYSIS OF OIL FILM BEARINGS

3.1 Introduction

The technical and industrial growth in the 19th century led to the widespread development in turbomachinery. The use of oil-film journal bearings to support the rotors led many people to investigate the characteristics of oil films, and their effect on the dynamics of rotor/bearing systems.

In 1886, Reynolds [33] published his classical work to establish the well known Reynolds equation. This defines the hydro-dynamic pressure distribution in an oil-film.

In 1925, Stodola [2] modeled a shaft supported on a journal bearing as a mass-spring system and he investigated the effect of the oil-film stiffness on the critical speed. This model was used to show the discrepancy between the observed critical speed and that predicted by assuming the bearing as a point support. However, because the damping ability of the bearing was ignored, he was unable to predict the amplitude of vibration at the critical speed. Later investigations in this field have shown that it is convenient to represent the shaft and the oil-film as a mass-damper-spring system are represented by four stiffness coefficients.

Theoretical works to determine these coefficients have produced formula which are valid under certain physical conditions and assumptions [4].

In 1963, Cooper [3] performed a series of experiments on a rigid shaft supported in oil-film bearing. The smooth running of the rotor was limited by the on set of the so-called 'oil-whip' phenomenon. The shaft was then supported using rolling element bearings whose outer races were prevented from rotating. He observed that the oil-whip phenomenon disappeared. This configuration where the journal does not rotate is termed a squeeze-film bearing.

These devices are commonly used in conjuction with a rolling element bearing. A ring is firmly attached to the outer race of the roller bearing and the annulus between the outer diameter of the damper ring and the bearing housing is filled with oil in Figure 1.1.1. Although squeeze-film bearings have a relatively short history they are now being extensively used in applications where it is necessary to limit rotor vibration and their effect on the supporting structure (e.g. turbine engines).

The dynamics of the squeeze- film bearing are dependent upon the bearing parameter $\frac{\eta R(l/c)^3}{mW}$ and the supply pressure. These factors effect the extend of the oil-film in the annulus.

Squeeze-film bearings can be designed to incorporate end seals and in many applications retainer springs are used to support the static load. In many particular configuration, the Reynolds equation is modified accordingly and solved to obtain the oil-film coefficients [34].

Most of the literature concerning squeeze-film bearings is devoted to the identification of oil-film coefficients and to their design, when they are used to support rigid or flexible rotors [35].

In this thesis, a squeeze-film damper without end seals and supported by retainer springs is investigated for state feedback control.

3.2 Dynamic Equations and Transfer Function Models

A squeeze-film bearing can be regarded as a special case of a journal bearing. The dynamic equations governing the dynamics of squeeze-film bearing are given in the following section.

3.2.1 Squeeze-film Bearing Equations

Assuming that the journal does not rotate, the four stiffness terms which are a function of the journal angular velocity, disappear. The static load capacity may then be provided by an external spring. Neglecting the cross-stiffness effects, the dynamic equations of motion for a squeeze-film bearing become

$$\begin{aligned} m\ddot{x} + c_{xx}\dot{x} + K_s x + c_{xy}\dot{y} &= f_x \\ m\ddot{y} + c_{yx}\dot{x} + K_s y + c_{yy}\dot{y} &= f_y \end{aligned} \qquad (3.1)$$

Where x and y represent the displacements in x and y direction respectively. It is assumed that the stiffness k_s is equal in both x and y directions.

The damping coefficients in local coordinates and the coordinate transformations are derived in [36].

In practice f_x and f_y are the components of the mass unbalance force along the x and y axis. However, the dynamics of the squeeze-film bearing can be simulated by applying external perturbations without the rotation of rotor.

3.2.2 Squeeze-film Bearing Oil-film Coefficients

The journal bearing is characterised by eight linear oil-film coefficients which were derived by Holmes [37]. The dynamics of the squeeze-film is characterised by considering the journal bearing when the journal rotation is suppressed. Then the four stiffness terms in oil-film disappear.

Assuming that the oil-film exists over an arc of 180° in Figure 3.2.1, the linearized damping terms in the local r-s coordinate system may be written as

$$c_{rr} = \frac{\eta l^3 R}{2c^3} \pi \frac{(1+2\epsilon_0^2)}{(1-\epsilon_0^2)^{5/2}}$$

$$c_{rs} = c_{sr} = \frac{\eta l^3 R}{2c^3} \pi \frac{\epsilon_0}{(1-\epsilon_0^2)^2} \qquad (3.2)$$

$$c_{ss} = \frac{\eta l^3 R}{2c^3} \pi \frac{1}{(1-\epsilon_0^2)^{3/2}}$$

The coefficients in equation (3.2) are expressed in dimensional form and they may be non-dimentionalised by defining non-dimensional terms such as

$$C_{rr} = \frac{c_{rr}}{\pi \eta R (l/c)^3} \quad , \text{ etc.} \tag{3.3}$$

Then the non-dimensional coefficients may be expressed as

$$C_{rr} = \frac{(1+2\epsilon_0^2)}{2(1-\epsilon_0^2)^{5/2}}$$

$$C_{rs} = C_{sr} = \frac{2\epsilon_0}{\pi(1-\epsilon_0^2)^2} \tag{3.4}$$

$$C_{ss} = \frac{1}{2(1-\epsilon_0^2)^{3/2}}$$

However it is normal practice to operate the squeeze-film bearing with a full 360^0 film in the annulus. Under these circumstances, the cross-damping terms c_{rs} and c_{sr} vanish, while the two direct terms double in value, such that

$$c_{rr}\Big|_{360} = \frac{\eta l^3 R}{c^3}\,\pi\,\frac{(1+2\epsilon_0^2)}{(1-\epsilon_0^2)^{5/2}}$$

$$c_{ss}\Big|_{360} = \frac{\eta l^3 R}{c^3}\,\pi\,\frac{1}{(1-\epsilon_0^2)^{3/2}} \tag{3.5}$$

Then the non-dimensional form of the coefficients in equation (3.5) may be written as

$$C_{rr} = \frac{(1+2\epsilon_0^2)}{(1-\epsilon_0^2)^{5/2}}$$

$$C_{ss} = \frac{1}{(1-\epsilon_0^2)^{3/2}} \tag{3.6}$$

3.2.3 Coordinate Transformations

The oil-film forces are genarally derived in the local axes coordinate system which is related to the attitude angle Φ_0, Figure 3.2.1 When formulating the equations of motion of a rotor supported in oil-film bearings, it is convenient to write the equations in the fixed axes (x, y) coordinate system.

From Figure 3.2.1 the displacements along r-s axes may be written as

$$r = y \sin \Phi_0 + x \cos \Phi_0$$

$$s = y \cos \Phi_0 - x \sin \Phi_0$$

and the velocities are

$$\dot{r} = \dot{y} \sin \Phi_0 + \dot{x} \cos \Phi_0$$
$$\dot{s} = \dot{y} \cos \Phi_0 - \dot{x} \sin \Phi_0 \qquad (3.8)$$

The oil-film forces along the x-y axes are resolved along the x and y axes as

$$f_x = f_r \cos \Phi_0 - f_s \sin \Phi_0$$
$$f_y = f_r \sin \Phi_0 + f_s \cos \Phi_0 \qquad (3.9)$$

The squeeze-film bearing forces along the local coordinate r-s can be expressed in terms of the linearized damping coefficients as

$$f_r = c_{rr}\dot{r} + c_{rs}\dot{s}$$
$$f_s = c_{sr}\dot{r} + c_{ss}\dot{s} \qquad (3.10)$$

and for the x-y axes they may be written as

$$f_x = c_{xx}\dot{x} + c_{xy}\dot{y}$$
$$f_y = c_{yx}\dot{x} + c_{yy}\dot{y} \qquad (3.11)$$

By algebraic manipulation of equations (3.8) to (3.11) the oil film coefficients in stationary x-y axes may be obtained in terms of the original coefficients in r-s coordinate system.

$$
\begin{aligned}
c_{xx} &= c_{rr}\cos^2\Phi_0 + c_{ss}\sin^2\Phi_0 - (c_{rs} + c_{sr})\cos\Phi_0\sin\Phi_0 \\
c_{xy} &= c_{rs}\cos^2\Phi_0 - c_{sr}\sin^2\Phi_0 + (c_{rr} - c_{ss})\cos\Phi_0\sin\Phi_0 \\
c_{yx} &= c_{sr}\cos^2\Phi_0 - c_{rs}\sin^2\Phi_0 + (c_{rr} - c_{ss})\cos\Phi_0\sin\Phi_0 \\
c_{yy} &= c_{ss}\cos^2\Phi_0 + c_{rr}\sin^2\Phi_0 + (c_{rs} + c_{sr})\cos\Phi_0\sin\Phi_0
\end{aligned}
\quad (3.12)
$$

As described before , for 360^0 film in the annulus, c_{rs} and c_{sr} become zero and c_{rr} and c_{ss} double in value. However, from equation (3.12) it is seen that the cross damping terms in x-y axes are non-zero. To explain the circumstance under which c_{xy} and c_{yx} tend to zero, a second constraint is considered. When the journal is centralised in the bearing, or when the displacement is along the vertical or horizontal axes, then coefficients c_{xy} and c_{yx} tend to zero [38]. This situation occurs when the attitude angle Φ_0 is set to 0^o or 90^o. In this case the cross damping terms in x-y axes disappear and horizontal and vertical motions of the journal centre are coupled.

3.2.4 Rotor - Bearing Transfer Function Models

For convenience equation (3.1) can be non-dimensionalised and the transfer functions can be derived as follows [39]:

Rewriting equation (3.1) in non-dimensional form

$$\frac{\ddot{x}}{cw^2} + L\left[C_{xx}\frac{\dot{x}}{cw} + C_{xy}\frac{\dot{y}}{cw}\right] + \bar{k}_s\frac{x}{c} = u_x$$

$$\frac{\ddot{y}}{cw^2} + L\left[C_{yy}\frac{\dot{y}}{cw} + C_{yx}\frac{\dot{x}}{cw}\right] + \bar{k}_s\frac{y}{c} = u_y \tag{3.13}$$

where $u_x = f_x/mcW^2$

$u_y = f_y/mcW^2$

$\bar{K}_s = K_s/mW^2$

and $L = \pi\eta R(l/c)^3/mW$

By introducing a suitable set of state variables, for example

$$x_1 = x/c\ ,\ x_2 = \dot{x}/cW\ ,\ x_3 = y/c\ ,\ x_4 = \dot{y}/cW$$

then equation (3.13) can be written in the state space as

$$\dot{x} = Ax + Bu \tag{3.14}$$

where

$$A = WL \begin{bmatrix} 0 & 1/L & 0 & 0 \\ -k_s/L & -C_{xx} & 0 & -C_{xy} \\ 0 & 0 & 0 & 1/L \\ 0 & -C_{yx} & -k_s/L & -C_{yy} \end{bmatrix}$$

is the system dynamic matrix,

$x = \left[x_1 , x_2 , x_3 , x_4 \right]^T$ is the state vector

$$B = \begin{bmatrix} 0 & W & 0 & 0 \\ 0 & 0 & 0 & W \end{bmatrix}^T$$

is the input transducer matrix, and $u = | u_x , u_y |$ is the input vector.

The corresponding output equation may be written as

$$y = Cx \qquad (3.15)$$

where $y = \left[y_1 , y_2 , y_3 , y_4 \right]^T$ is the output vector.

The output transducer matrix C, may be taken as the identity matrix since all elements of the state vector x are directly available for measurement.

Taking Laplace transformations of equations (3.14) and (3.15), and assuming zero initial conditions, the transfer function matrix can be obtained as

$$G(s) = \frac{Y(s)}{U(s)} = |sI-A|^{-1} B \tag{3.17}$$

where

$$G(s) = \frac{1}{\Delta(s)} \begin{bmatrix} \frac{s^2}{w^2} + \frac{s}{w} LC_{yy} + \bar{k}_s & -\frac{s}{w} LC_{xy} \\ \frac{s}{w} \left| \frac{s^2}{w^2} + \frac{s}{w} LC_{yy} + \bar{k}_s \right| & -\frac{s^2}{w^2} LC_{xy} \\ -\frac{s}{w} LC_{yx} & \frac{s^2}{w^2} + \frac{s}{w} LC_{xx} + \bar{k}_s \\ -\frac{s^2}{w^2} LC_{yx} & \frac{s}{w} \left| \frac{s^2}{w^2} + \frac{s}{w} LC_{xx} + \bar{k}_s \right| \end{bmatrix}$$

The characteristic equation is

$$\Delta(s) = \frac{s^4}{w^4} + \frac{s^3}{w^3} + (C_{xx} + C_{yy}) + \frac{s^2}{w^2} \left| L^2 (C_{xx} C_{yy} - C_{xy} C_{yx}) + 2\bar{k}_s \right|$$

$$+ \frac{s}{w} L(C_{xx} + C_{yy})\bar{k}_s + \bar{k}_s = 0 \tag{3.18}$$

Denoting an element of G(s) as $g_{ij}(s)$, the response of the four states when a force is applied sequentially to each of the two input channel is,

$$Y_i(s) = \frac{g_{ij}(s)}{\Delta(s)} U_j(s) = T_{ij}(s)\, U_j(s) \tag{3.19}$$

where T_{ij} is the transfer function between the j^{th} input, $j=1,2$ and i^{th} output, $i=1,2,3,4$.

The transfer functions relating the vertical and horizontal displacements to horizontal forcing are,

$$T_{12}(s) = \frac{Y_1(s)}{U(s)} = \frac{-(s/W)\,LC_{xy}}{\Delta(s)} \qquad (3.20)$$

$$T_{32}(s) = \frac{Y_3(s)}{U(s)} = \frac{(s^2/W^2)+(s/W)LC_{xx}+K_s}{\Delta(s)} \qquad (3.21)$$

When the journal is centralised in the bearing, or when the displacement is along the vertical or horizontal axis, then coefficients C_{xy} and C_{yx} tend to zero [38]. Under these conditions the coupled model reduces to second order uncoupled model as follows

$$T_{32} = \frac{1}{s^2 + sLC_{ss} + K_s} \qquad \text{for } \Phi_0 = 0^0 \qquad (3.22)$$

$$T_{32} = \frac{1}{s^2 + sLC_{rr} + K_s} \qquad \text{for } \Phi_0 = 90^0 \qquad (3.23)$$

IV. STATE-FEEDBACK CONTROL OF SQUEEZE-FILM BEARING-ROTOR SYSTEM AND COMPUTER PROGRAMMING

4.1 State-feedback Modelling of Squeeze-film Bearing-Rotor System

It is well known that the state variable feedback can be used to control system modes of vibration. The object in eigenvalue assignment in rotor-bearing system would be to stabilize an unstable system or to obtain a better operating system which would be physically difficult to design.

In this work, the equations used are in dimensional form. The dynamic equations of motion for a rigid rotor supported at the ends by squeeze-film bearing are given by:

$$M\ddot{x} + C_{xx}\dot{x} + K_s x + C_{xy}\dot{y} = U_x$$
$$M\ddot{y} + C_{yx}\dot{x} + K_s y + C_{yy}\dot{y} = U_y \qquad (4.1)$$

By introducing a suitable set of state variables, for example

$$x_1 = x\ ,\ x_2 = \dot{x}\ ,\ x_3 = y\ ,\ x_4 = \dot{y}$$

then the equation (4.1) can be written in the state space as

$$\dot{x} = Ax + Bu \qquad (4.2)$$

where

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -K_s/M & -C_{xx}/M & 0 & -C_{xy}/M \\ 0 & 0 & 0 & 1 \\ 0 & -C_{yx}/M & -K_s/M & -C_{yy}/M \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 1/M & 0 & 0 \\ 0 & 0 & 0 & 1/M \end{bmatrix}$$

4.1.1 Selection of Models For Squeeze-film Rotor-Bearing System

For the simulations four types of models are selected which have different configurations of eigenvalues. Datas of these models are given in Table 4.1.1. These models are tabulated in table 4.1.2. The eigenvalues of the uncontrolled models are shown on Figure 4.1.1 and given in table 4.1.3.

As it is seen in Figure 4.1.1 all the eigenvalues of each model are located at the left hand side of the s-plane. Hence the original (open-loop) models are stable. Obviously, their behaviour in the time domain are dependent upon the eigenvalues.

In Model I, the first two eigenvalues are complex conjugates and the other two are distinct real. It is possible to find out the eigenvalues corresponding to the x and y-directions respectively. Considering the system equations it is observed that the displacements in x and y-directions are weakly coupled by the cross-damping coefficients C_{xy} and C_{yx} which are equal and small.

Therefore, it is possible to treat the system as uncoupled. In this case, the system can be reduced to two uncoupled second order models in x and y-directions. For a second order model, general characteristic equation is of the form $s^2 + a_1 s + a_2 = 0$ in which a_1 represents the addition and a_2 is the multiplication of eigenvalues respectively.

For Model I, we can find the coefficients of the characteristic equations in x and y-direction as $a_{x1}=20000$, $a_{x2}=85.532$, $a_{y1}=20000$ and $a_{y2}=429.840$. First, let us consider two poles from Figure 4.1.2, namely, $\lambda_{1,2} = -39 \pm i135$ and try to find out to which direction these poles will correspond. It is clear that, the multiplication of these poles is 19746 and the addition is 78. These numbers are very close to a_{x1} and a_{x2} and therefore these poles are the eigenvalues of the displacement in x-direction.

It is also possible to reach to the same conclusion for the y-direction. Like the multiplication of the other two poles is 20228 and the addition is 441. Therefore, there is enough proof that these poles lie in the y-direction.

In order to make a time domain analysis we can make use of the dominant root concept. The time constant, $\tau=1/\zeta w_n$, is the reciprocal of the distance from the root to the imaginary axis. All roots lying on a given vertical line in the s-plane have the same time constant and the greater the distance of the line from the imaginery axis, the smaller is the time constant. This leads us to the concept of the dominant root. For a given characteristic equation, this is the root that lies nearest to the imaginary axis. therefore, if the system is stable, the dominant root is the root with the largest time constant. The usefulness of the dominant root is that it allows us to approximate the speed of response for the system.

The free response curves in x and y directions of the uncontrolled Models I, II, III and IV are given in Figure 4.1.2 to 4.1.9 respectively. In Model I, oscillation occurs only in x-direction and not in y-direction, because in x-direction the eigenvalues have imaginary parts whereas in y-direction, they are distinct real.

If 2% settling time criteria is used, then the settling time (t_s) is approximately four times the time constant of the system. In Model I, the settling times of the x and y-directions are calculated as 0.102 sec. and 0.076 sec. respectively. When these are compared with Figures 4.1.2 and 4.1.3, it is seen that the results are reasonable. Similar results can be obtained for the other models.

In order to apply the state-feedback control to the rotor-bearing system, a set of desired eigenvalues should be chosen. The choice should be such that the motion in x and y directions are overdamped. However, for the sake of application two set of desired eigenvalues given below are chosen.

1 st set of desired eigenvalues :

$\lambda_1 = -1 + i0.5$ $\lambda_2 = -1 - i0.5$ $\lambda_3 = -1.25$ $\lambda_4 = -1.5$

2 nd set of desired eigenvalues :

$\lambda_1 = -2$ $\lambda_2 = -3$ $\lambda_3 = -4$ $\lambda_4 = -5$

The response of the closed-loop system for the first set of desired eigenvalues is given in Figures 4.1.10 -4.1.11 and for the second set of desired eigenvalues in Figure 4.1.12-4.1.13 for x and y-directions respectively. From these figures, it is seen that the assignment of a complex conjugate pair of eigenvalues causes oscillatory motion in x-direction in Figure 4.1.10, but a smooth response in y-direction in Figure 4.1.11. For the second set of distinct desired eigenvalues, the expected displacements are smoother than those of the original uncontrolled system. Therefore these results verify the correctness of the phase-variable canonical method for eigenvalue assignment and also the correctness of the computational work. Hence the performance of any given system can be improved using state-feedback control. State-feedback matrices for first and second desired eigenvalues are obtained as

$$K_1 = \begin{bmatrix} 500000.00 & 5.86 & 25.00 & 3.12 \\ -339.84 & -589.05 & 499634.00 & 89.26 \end{bmatrix}$$

$$K_2 = \begin{bmatrix} 500000.00 & 5.86 & 25.00 & 3.12 \\ -3000.00 & -3846.87 & 498225.00 & -320.05 \end{bmatrix}$$

4.2 Computer Programming

In this thesis, the phase-variable canonical method is programmed for state-feedback control application. The program named THESIS1 is written in FORTRAN language according to the mathematical models presented in section 4.1. The computer program is listed in Appendix III. The computer program THESIS1 is modular and user friendly. In main program there are 18 subroutines, some of them being called IMSL library. The flowchart of the programming logic is given in figure 4.2.1. The program after asking for the inputs interactively, creates the dynamic matrix of the system using oil-film coefficients which are given in Chapter III and then finds its eigenvalues. Followingly, it imtegrates the system equations using Runge-Kutta method and then draws x-y displacement responses of the system. Then, it computes the controllability matrix of the system and subsequently it continues for calculations of obtaining phase-variable canonical-form. At the end of these calculations, THESIS1 computes the state-feedback gain matrix which are to satisfy for the desired eigenvalues. Finally, THESIS1 computes the closed-loop dynamic matrix and it solves the new state-space equations for the controlled responsesare drawn on the screen.

4.2.1 Subroutines

The main subroutines which are used are given below:

Subroutine INTER : This subroutine asks interactively for parameters which are necessary to design rotor bearing coefficients.

Subroutine ABMAT : This subroutine computes the state-space representation matrices A and B after calculating the oil-film damping coefficients.

Subroutine TRANS : This subroutine renamed the A and B matrices for following steps.

Subroutine EIGEN : The eigenvalues of dynamic matrix A are computed using subroutine EIGRF in IMSL library.

Subroutine RUN1 : This subroutine solves the open-loop system equations using Runge-Kutta method using subroutine DVERK in IMSL library. The subroutine FCN1 is called in subroutine RUN1 is used for writing system equations which are to be solved. The subroutine GRAPH in RUN1 draws graphics using the points obtained with the Runge-Kutta method. In addition, there are two more subroutine in GRAPH called as AXIS and G which are used for plotting purpose.

Subroutine COM : This subroutine computes the time variable controllability matrix in canonical form using the method explained in part 2.4.3.

Subroutine INV : This subroutine computes the inverse matrix of a given input matrix using subroutine LINV1F in IMSL library.

Subroutine TTRA : The transpose of matrices are computed using TTRA subroutine.

Subroutine TMAT : This subroutine computes the T transformation matrix in $x = Tz$ using the inverse controllability matrix.

Subroutine ATRA : This subroutine computes the transformed matrices A and B in phase-variable canonical form.

Subroutine SEIG : This subroutine computes the characteristic coefficients of the system in matrix form.

Subroutine DEIG : This subroutine arranges the desired matrix respect to desired eigenvalues and computes the coefficients of desired characteristic equation in matrix form.

Subroutine GAIN : The state-feedback gain matrix is computed using the transformations which are explained in Chapter II.

Subroutine COE : This subroutine computes the closed-loop dynamic matrix after state-feedback control.

Subroutine RUN2 : This subroutine integrates the closed-loop system equations using Runge-Kutta method using subroutine DVERK in IMSL library. The subroutine FCN2 in RUN2 is used for writing equations which are to be solved. Then it is called to subroutine GRAPH for graphics similarly as in subroutine RUN1.

V. CONCLUSIONS

In this study various approaches to pole assignment problem and its application to a rigid rotor supported on squeeze-film bearings have been discussed. However, there is not a unique state-feedback control method due to the ambuiguity of the problem in multi-input -multi-output systems. Furthermore most of the available algorithms proceed by first transforming the system equations into a canonical form in the interest of computational tractability. The most attention deserving part of this study is the generalization of Ackermann's [24] procedure to multivariable systems. A trick is used to generalize Ackermann's procedure to multivariable systems of interest into an equivalent single input system which was given direct design procedure in chapter II. Direct design procedure is extremely convenient to use with multivariable systems, since it requires no explicit transformation of the system equations into a canonical form and it considerably reduces the number of computations required in determining the feedback gain matrix K. As explained in detail in the second chapter, this transformation into an equivalent single input system is establised by choosing the feedback matrix K, namely by setting K = qp, where q is arbitrary chosen, with the only restriction of preserving the system's controllability characteristics. However, additional flexibility can be introduced into the control system design if q can be chosen appropriately. Therefore, an interesting point which still deserves special attention is the way in which q must be chosen.

Another point which is still open for further research is the modification of Ackermann's original procedure such that it will also cover pole assignment through only output feedback. Use of the output controllability matrix to derive a formula similar to Ackermann's original one will be a logical step to start this further research.

Finally, it is concluded that the phase-variable canonical method is successfully applied for eigenvalue assignment in rotor bearing systems. The performance of these systems can be improved in certain cases or they can be stabilized if they are unstable. For example, it is well known that journal bearings become unstable when the operational speed is twice the first critical speed [2]. Such theoretically a system has been stabilized using state-feedback control approach [40].

However, the proof presented in this study make use of the canonical system equations and is general enough to include all possible systems.

APPENDIX I

Lemma 1

Unless n vectors have already been selected (and retained in the process (2.15)-(2.18) in chapter 2, there is a vector of form $A^j b_K$, where all lower powers of A times b_K have been retained, which is linearly independent of all previously selected vectors.

Proof: Suppose that the selected vectors are:

$$b_1, Ab_1, \ldots A^{q1} b_1, b_2, Ab_2, \ldots A^{q2} b_2, b_m, \ldots A^{qm} b_m \qquad \text{(A1a)}$$

and that each of the vectors

$$A^{q_1+1} b_1, A^{q_2+1} b_2, \ldots\ldots A^{q_m+1} b_m$$

is linearly independent on the selected vectors, so that the process terminates. It then follows by the induction argument sketched below that all other vectors in the controllability matrix (2.16). This in turn implies that either the controllability matrix is less than n or there are n independent vectors in the selection (A1a).

A sketch of induction proof is as follows: The vector $A^{q_1+2}.b_1$ is $A.A^{q_1+1}$, since $A^{q_1+1}.b_1$ is a linear combination of A times the selected vectors.

$A^{q_1+1}.b_1$ is the same linear combination of A times selected vectors. However, by hypothesis, A times any selected vector is also a linear combination of selected vectors, thus $A^{q_1+2}.b_1$ is a linear combination of selected vectors. Proceeding in this fashion one proves that all remaining vectors in the controllability matrix are depend on the selected vectors.

APPENDIX II

It is shown that the matrix T defined by (2.26) is nonsingular. To this and it is sufficient to show that the rows of T are linearly independent or equivalently that any null linear combination of the rows must be the linear combination consisting of zeros.

Suppose there are constants a_{ij} such that

$$\sum_{i=1}^{m} \sum_{j=1}^{p_i} a_{ij} e_j A^{j-1} = 0 \qquad \text{(A2a)}$$

Taking the inner product of both sides of the this equation with b_K produces

$$a_{Kp_K} = 0 \qquad \text{(A2b)}$$

since by definition of the a_i's each term in the inner linear product is zero except the one involving $e_K A^{p_K - 1} . b_K$ which is unity.

In view of (A2a), (A2b) can be written equivalently as

$$\sum_{i=1}^{m} \sum_{j=1}^{p_{i-1}} a_{ij} e_j A^{j-1} = 0 \qquad \text{(A2c)}$$

Taking the inner product of both sides of this with A_{bK} produces

$$a_{K,\,pK-1} = 0 \qquad \text{(A2d)}$$

continuing in tis manner, by induction, it is proved that each $a_{ij}=0$ which completes the proof.

APPENDIX III

```
C      ********************************************************************
C          THIS PROGRAM APPLIES THE STSTE-FEEDBACK CONTROL
C        TO A RIGID ROTOR BEARING SUPPORTED ON SQUEEZE-FILM
C      ********************************************************************
       PROGRAM CONT(INPUT,OUTPUT,OUT1,OUT2,OUT3,OUT4,TAPE5=INPUT,
      $ TAPE6=OUTPUT,TAPE8=OUT1,TAPE9=OUT2,TAPE10=OUT3,TAPE11=OUT4)
       REAL AINV(10,10),WKAREA(10),D(10,10)
       REAL LU(10,10),D1,D2,EQUIL(10),WA
       REAL A(10,10),A1(10,10),A2(10,10),A3(10,10)
       REAL B(10,10),B1(10,10),B2(10,10),B3(10,10)
       REAL AP1(10,10),AP2(10,10),AP3(10,10)
       REAL C1(10,10),C2(10,10),C3(10,10),C4(10,10)
       REAL COO(10,10),COEX(10,10),CON(10,10),COINV(10,10)
       REAL TR(10,10),TTO(10,10),TON(10,10)
       REAL TX(10,10),TI(10,10),TB(10,10),TC(10,10)
       REAL AT(10,10),BT(10,10),AMT(10,10)
       REAL SA(10,10),SB(10,10),BTINV(10,10)
       REAL NAD(10,10),AD(10,10)
       REAL KK1(10,10),TA(10,10)
       REAL CC(10,10),EI(10,10),EIG(10,10)
       REAL WK(24),RZ(32),RW(8)
       REAL C(24),W1(4,100),T,TOL,TEND
       REAL X(1000),Y1(1000)
       REAL X1(1000),X2(1000),X3(1000),X4(1000)
       REAL LE,MA,KS
       INTEGER IND,NW,K
       INTEGER N,IA,IDGT,IER
       INTEGER IPVT(4)
       COMPLEX W(8),Z(8,8),ZN
       EQUIVALENCE (W(1),RW(1)), (Z(1,1),RZ(1))
       COMMON/QWER/CXX,CXY,CYX,CYY,MA,KS
       COMMON/COEF1/CA11,CA12,CA13,CA14
       COMMON/COEF2/CA21,CA22,CA23,CA24
       COMMON/COEF3/CA31,CA32,CA33,CA34
       COMMON/COEF4/CA41,CA42,CA43,CA44
       CHARACTER*1 CH
       REWIND 8
       REWIND 9
       REWIND 10
       REWIND 11
C      SPECIFIC VALUES GIVEN FOR WHICH THE SOLUTION IS SEARCHED
C      DX1=-1.
C      DX2=.25
C      DX3=-1.25
C      DX4=-1.5
C      LE=.009
C      R=.09
C      CL=.0004
C      VI=.03
C      EP=0.6
C      FI=45
C      MA=25.
C      KS=5E+05
C      L=4
```

```
C     M=2
C     NOUT=6
C     ID1=ID2=2
C
      CALL INTER (LE,R,CL,FI,EP,KS,MA,VI,L,M,ID1,ID2,
     $DX1,DX2,DX3,DX4,NOUT,IGRA,IPLT,CHK)
      CALL ABMAT(A,B,LE,R,CL,VI,EP,FI,L,M)
      CALL TRANS (A,B,A1,B1,A2,B2,A3,B3,L,M)
      CALL EIGEN (A3,L,M,NOUT)
      PRINT*,'*ENTER 1 TO CONTINUE'
      READ(5,*)CONTI
      CALL RUN1(L,CL,FI,IGRA,IPLT,NOUT)
      CALL POWA (A1,A2,AP1,AP2,AP3,L,M)
      CALL COM (B1,AP1,AP2,AP3,COEX,ID1,ID2,L,M,NOUT)
      CALL INV (L,COEX,COINV,NOUT)
      CALL TTRA (COINV,TR,ID1,ID2,L,M,NOUT)
      CALL TMAT (TR,AP1,AP2,AP3,TX,ID1,ID2,TI,TB,TC,L,M,NOUT)
      CALL INV (L,TX,TINV,NOUT)
      CALL ATRA(TI,A1,TINV,AT,B1,BT,L,M,NOUT)
      CALL SEIG (AT,BT,SA,SB,ID1,L,M,NOUT)
      CALL DEIG(NAD,ID1,L,M,DX1,DX2,DX3,DX4,NOUT)
      CALL INV (M,SB,BTINV,NOUT)
      CALL GAIN (NAD,SA,BTINV,C1,TC,KK1,L,M,NOUT)
      CALL COE (A2,B2,KK1,EI,L,M,NOUT)
      CALL EIGEN (EI,L,M,NOUT)
      PRINT*,'*WHICH ARE SAME AS DESIRED ONES'
      PRINT*,'*IF YES     ENTER 1'
      PRINT*,'*IF NO      ENTER 0'
      READ(5,*)CONTI
      IF (CONTI.EQ.0.0) GO TO 50
      CALL RUN2(L,CL,FI,IGRA,IPLT,NOUT)
C
   50 STOP
      END

C     ****************************************************************
C     THIS SUBROUTINE ASKS THE VALUE OF VARIABLES INTERACTIVELY
C     ****************************************************************
      SUBROUTINE INTER (LE,R,CL,FI,EP,KS,MA,VI,L,M,ID1,ID2,
     $DX1,DX2,DX3,DX4,NOUT,IGRA,IPLT,CHK)
      REAL LE,MA,KS
      CHARACTER*1 CH
      IF(CHK.EQ.1)GO TO 50
    1 PRINT*,' 1-ENTER THE LAND LENGTH OF BEARING              ',
     $'                  (MT)'
      READ(5,*)LE
      IF(CHK.EQ.1)GO TO 50
    2 PRINT*,' 2-ENTER THE RADIUS OF JOURNAL BEARING           ',
     $'                  (MT)'
      READ(5,*)R
      IF(CHK.EQ.1)GO TO 50
    3 PRINT*,' 3-ENTER THE CLEARANCE BETWEEN HOUSING AND JOU',
     $'RNAL BEARING  (MT)'
      READ(5,*)CL
      IF(CHK.EQ.1)GO TO 50
    4 PRINT*,' 4-ENTER THE ATTITUDE ANGLE IN COORDINATE SYSTEM'
      READ(5,*)FI
      IF(CHK.EQ.1)GO TO 50
    5 PRINT*,' 5-ENTER THE STATIC ECCENTRICITY RATIO'
      READ(5,*)EP
      IF(CHK.EQ.1)GO TO 50
```

```
  6 PRINT*,' 6-ENTER THE RETAINER SPRING STIFFNESS COEFFIC',
   $'IENT          (N/M)'
    READ(5,*)KS
    IF(CHK.EQ.1)GO TO 50
  7 PRINT*,' 7-ENTER THE MASS PER LAND OF THE BEARING',
   $'    (KG)'
    READ(5,*)MA
    IF(CHK.EQ.1)GO TO 50
  8 PRINT*,' 8-ENTER THE VISCOSITY OF OIL',
   $'          (NM/S**2)'
    READ(5,*)VI
    IF(CHK.EQ.1)GO TO 50
  9 PRINT*,' 9-ENTER THE REAL PART OF DESIRED DOMINANT POLES'
    READ(5,*)DX1
    IF(CHK.EQ.1)GO TO 50
 10 PRINT*,'10-ENTER THE IMAGINER PART OF DOMINANT POLES',
   $' IN (+) SIGN'
    READ(5,*)DX2
    IF(CHK.EQ.1)GO TO 50
 11 PRINT*,'11-ENTER THE THIRD REAL DESIRED POLE'
    READ(5,*)DX3
    IF(CHK.EQ.1)GO TO 50
 12 PRINT*,'12-ENTER THE FOURTH REAL DESIRED POLE'
    READ(5,*)DX4
    IF(CHK.EQ.1)GO TO 50
 13 PRINT*,'13-NUMERICAL OUTPUTS REQUIRED ?'
    PRINT*,'IF YES       ENTER 6'
    PRINT*,'IF NO        ENTER 8'
    READ(5,*)NOUT
    IF(CHK.EQ.1)GO TO 50
 14 PRINT*,'14-GRAPHICAL OUTPUT REQUIRED ?'
    PRINT*,'IF YES       ENTER 1'
    PRINT*,'IF NO        ENTER 0'
    READ(5,*)IGRA
    IF(CHK.EQ.1)GO TO 50
 15 PRINT*,'15-PLOTTER OUTPUT REQUIRED ?'
    PRINT*,'IF YES       ENTER 1'
    PRINT*,'IF NO        ENTER 0'
    READ(5,*)IPLT
    IF(CHK.EQ.1)GO TO 50
C       SPECIFIC VALUES FOR SYSTEM
C       L=DEGREE OF SYSTEM
C       M=DEGREE OF INPUT MATRIX
C       ID1=DEGREE OF FIRST DIAGONAL MATRIX IN JOURDAN FORM
C       ID2=DEGREE OF SECONT DIAGONAL MATRIX IN JOURDAN FORM
    L=4
    M=2
    ID1=2
    ID2=2
C
 50 PRINT*,' 1-LAND LENGTH OF BEARING                        :',LE
    PRINT*,' 2-RADIUS OF JOURNAL BEARING                     :',R
    PRINT*,' 3-CLEARANCE                                     :',CL
    PRINT*,' 4-ATTITUDE ANGLE IN COORDINATE SYSTEM           :',FI
    PRINT*,' 5-STATIC ECCENTRICITY                           :',EP
    PRINT*,' 6-RETAINER SPRING STIFFNESS COEFFICIENT         :',KS
    PRINT*,' 7-MASS PER LAND OF BEARING                      :',MA
```

```
      PRINT*,' 8-OIL VISCOSITY                                 :',VI
      PRINT*,' 9-REAL PART OF DESIRED DOMINANT POLES           :',DX1
      PRINT*,'10-IMAGINER PART OF D.DOMINANT POLES             :',DX2
      PRINT*,'11-REAL THIRD DESIRED POLE                       :',DX3
      PRINT*,'12-REAL FOURTH DESIRED POLE                      :',DX4
      PRINT*,'13-NUMERICAL OUTPUT                              :',NOUT
      PRINT*,'14-GRAPHICAL OUTPUT                              :',IGRA
      PRINT*,'15-PLOTTER OUTPUT                                :',IPLT
C
      PRINT*,'*DO YOU WANT TO CHANGE ANY VARIABLE ? Y/N'
      READ(5,30)CH
   30 FORMAT(A1)
      IF(CH.EQ.'N')GO TO 60
   80 WRITE(6,'("*WRITE THE NUMBER OF VARIABLE THAT YOU WANT ",
     $"TO CHANGE")')
      READ(5,*)NU
      IF(NU.GT.15) GO TO 85
      CHK=1
      GO TO (1,2,3,4,5,6,7,8,9,10,11,12,13,14,15)NU
   85 WRITE(6,'("*THERE IS NO VARIABLE THAT MATCHES YOUR CHOISE",///,
     $"*PLEASE TRY AGAIN")')
      GO TO 80
   60 CONTINUE
      RETURN
      END

C     ***********************************************************
C     THIS SUBROUTINE COMPUTES THE STATE-SPACE REPRESENTATION
C     MATRIX
C     ***********************************************************
      SUBROUTINE ABMAT(A,B,LE,R,CL,VI,EP,FI,L,M)
      REAL A(10,10),B(10,10)
      REAL LE,MA,KS
      COMMON/QWER/CXX,CXY,CYX,CYY,MA,KS
      PI=3.1415927
      CRR=VI*(LE**3)*R/(2*(CL**3))*PI*(1+2*(EP**2))/
     $((1-(EP**2))**2.5)
      CRS=VI*(LE**3)*R/(2*(CL**3))*4*EP/((1-(EP**2))**2)
      CSR=CRS
      CSS=VI*(LE**3)*R/(2*(CL**3))*PI*1/((1-(EP**2))**1.5)
C
      CXX=(CRR*(COS(FI)**2))+(CSS*(SIN(FI)**2))-
     $((CSR+CRS)*COS(FI)*SIN(FI))
      CXY=CRS*(COS(FI)**2)-CSR*(SIN(FI)**2)+
     $(CRR-CSS)*COS(FI)*SIN(FI)
      CYX=CXY
      CYY=CSS*(COS(FI)**2)+CRR*(SIN(FI)**2)+
     $(CRS+CSR)*COS(FI)*SIN(FI)
      WRITE(NOUT,75)
      WRITE(NOUT,78)
      WRITE(NOUT,75)
      WRITE(NOUT,79)
      WRITE(NOUT,80)
      WRITE(NOUT,15)CRR,CXX,CSR,CXY,CRS,CYX,CSS,CYY
      WRITE(NOUT,80)
```

```
      A(1,1)=A(1,3)=A(1,4)=A(2,3)=0.0
      A(3,1)=A(3,2)=A(3,3)=A(4,1)=0.0
      A(1,2)=A(3,4)=1.0
      A(2,1)=-KS/MA
      A(2,2)=-CXX/MA
      A(2,4)=-CXY/MA
      A(4,2)=-CYX/MA
      A(4,3)=-KS/MA
      A(4,4)=-CYY/MA
      B(2,1)=B(4,2)=1/MA
      B(1,1)=B(1,2)=B(2,2)=B(3,1)=B(3,2)=B(4,1)=0.0
      WRITE(6,75)
      WRITE(6.76)
      WRITE(6 75)
      DO 1 I=1,4
    1 WRITE(6.10)(A(I,J),J=1,4)
      WRITE(6,75)
      WRITE(6.77)
      WRITE(6,75)
      DO 2 I=1,4
    2 WRITE(6.10)(B(I,J),J=1,2)
   10 FORMAT(4(1X,E15.8))
   15 FORMAT(7X,'CRR',4X,F15.8,8X,'CXX',4X,F15.8,/,
     $7X,'CRS',4X,F15.8,8X,'CXY',4X,F15.8,/,
     $7X,'CSR',4X,F15.8,8X,'CYX',4X,F15.8,/,
     $7X,'CSS',4X,F15.8,8X,'CYY',4X,F15.8)
   75 FORMAT(8X,'-----------------------------------------------------------------------------------'
   76 FORMAT(8X,'***                          DYNAMIC MATRIX                                   ***'
   77 FORMAT(8X,'***                           INPUT MATRIX                                    ***'
   78 FORMAT(8X,'***                     DAMPING COEFFICIENTS                                  ***'
   79 FORMAT(8X,'**POLAR COORDINATES**',6X,'**CARTASIAN COORDINATES**'
   80 FORMAT(8X,'-------------------------',6X,'-----------------------------------------'
      RETURN
      END

C     ******************************************************
C     THIS SUBROTINE RENAMED THE STATE-SPACE MATRICES
C     ******************************************************
      SUBROUTINE TRANS (A,B,A1,B1,A2,B2,A3,B3,L,M)
      REAL A(10,10),A1(10,10),A2(10,10),A3(10,10)
      REAL B(10,10),B1(10,10),B2(10,10),B3(10,10)
      REWIND 8
      DO 1 I=1,L
      DO 1 J=1,L
      A1(I,J)=A(I,J)
      A2(I,J)=A(I,J)
    1 A3(I,J)=A(I,J)
      DO 2 I=1,L
      DO 2 J=1,M
      B1(I,J)=B(I,J)
      B2(I,J)=B(I,J)
    2 B3(I,J)=B(I,J)
      DO 51 I=1,L
      WRITE(8,10)(A1(I,J),J=1,L)
      WRITE(8,10)(A2(I,J),J=1,L)
   51 WRITE(8,10)(A3(I,J),J=1,L)
```

```
      DO 52 I=1,L
      WRITE(8,10)(B1(I,J),J=1,M)
      WRITE(8,10)(B2(I,J),J=1,M)
   52 WRITE(8,10)(B3(I,J),J=1,M)
   10 FORMAT(10(1X,E15.8))
      RETURN
      END
C     ***********************************************
C     THIS SUBROUTINE CALCULATES POWERS OF A-MATRIX
C     ***********************************************
      SUBROUTINE POWA (A1,A2,AP1,AP2,AP3,L,M)
      REAL A1(10,10),A2(10,10)
      REAL AP1(10,10),AP2(10,10),AP3(10,10)
      REWIND 8
      DO 1 I=1,L
      DO 1 J=1,L
    1 AP1(I,J)=A1(I,J)
      DO 51 I=1,L
   51 WRITE(8,10)(AP1(I,J),J=1,L)
      DO 2 I=1,L
      DO 2 J=1,L
      AP2(I,J)=0.0
      DO 2 K=1,L
    2 AP2(I,J)=AP2(I,J)+A1(I,K)*A2(K,J)
      DO 52 I=1,L
   52 WRITE(8,10)(AP2(I,J),J=1,L)
      DO 3 I=1,L
      DO 3 J=1,L
      AP3(I,J)=0.0
      DO 3 K=1,L
    3 AP3(I,J)=AP3(I,J)+A1(I,K)*AP2(K,J)
      DO 53 I=1,L
   53 WRITE(8,10)(AP3(I,J),J=1,L)
   10 FORMAT(10(1X,E15.8))
      RETURN
      END
C     ***********************************************************
C     THIS SUBROUTINE CALCULATES CONTROLLABILITY MATRIX OF SYSTEM
C     ***********************************************************
      SUBROUTINE COM (B1,AP1,AP2,AP3,COEX,ID1,ID2,L,M,NOUT)
      REAL B1(10,10),AP1(10,10),AP2(10,10),AP3(10,10)
      REAL C1(10,10),C2(10,10),C3(10,10),C4(10,10)
      REAL CCO(10,10),COEX(10,10),CON(10,10)
      REWIND 8
      DO 1 I=1,L
      DO 1 J=1,M
    1 C1(I,J)=B1(I,J)
      DO 51 I=1,L
   51 WRITE(8,10)(C1(I,J),J=1,M)
      DO 2 I=1,L
      DO 2 J=1,M
      C2(I,J)=0.0
      DO 2 K=1,L
```

```
  2 C2(I,J)=C2(I,J)+AP1(I,K)*B1(K,J)
    DO 52 I=1,L
 52 WRITE(8,10)(C2(I,J),J=1,M)
    DO 3 I=1,L
    DO 3 J=1,M
    C3(I,J)=0.0
    DO 3 K=1,L
  3 C3(I,J)=C3(I,J)+AP2(I,K)*B1(K,J)
    DO 53 I=1,L
 53 WRITE(8,10)(C3(I,J),J=1,M)
    DO 4 I=1,L
    DO 4 J=1,M
    C4(I,J)=0.0
    DO 4 K=1,L
  4 C4(I,J)=C4(I,J)+AP3(I,K)*B1(K,J)
    DO 54 I=1,L
 54 WRITE(8,10)(C4(I,J),J=1,M)
    DO 5 I=1,L
    DO 5 J=1,M
    CCO(I,J)=C1(I,J)
    CCO(I,J+2)=C2(I,J)
    CCO(I,J+4)=C3(I,J)
  5 CCO(I,J+6)=C4(I,J)
    DO 55 I=1,L
 55 WRITE(8,10)(CCO(I,J),J=1,L*M)
C
    KK=-1
    K=-L
    DO 6 ITER=1,M
    KK=KK+1
    K=K+L
    DO 7 I=1,L
    DO 7 J=1,L
    LL=2*J-(2-1)+KK
  7 CON(I,J+K)=CCO(I,LL)
  6 CONTINUE
    WRITE(NOUT,75)
    WRITE(NOUT,76)
    WRITE(NOUT,75)
    DO 8 I=1,L
    WRITE(8,10)(CON(I,J),J=1,L*M)
  8 WRITE(NOUT,10)(CON(I,J),J=1,L*M)
C
C   USE ID1,ID2
C
    DO 9 J=1,ID1
    DO 9 I=1,L
  9 COEX(I,J)=CON(I,J)
    DO 11 J=1,ID2
    DO 11 I=1,L
 11 COEX(I,J+ID1)=CON(I,J+4)
    WRITE(NOUT,75)
    WRITE(NOUT,77)
    WRITE(NOUT,75)
    DO 12 I=1,L
    WRITE(8,10)(COEX(I,J),J=1,L)
```

```
   12 WRITE(NOUT,10)(COEX(I,J),J=1,L)
   10 FORMAT(10(1X,E15.8))
   75 FORMAT(12X,'---------------------------------------------------------')
   76 FORMAT(12X,'***CONTROLLABILITY MATRIX (B,BA,..(B1,B2) ***')
   77 FORMAT(12X,'***    SELECTED CONTROLLABILITY MATRIX    ***')
      RETURN
      END
C     *********************************************************************
C     THIS SUBROUTINE COMPUTES THE INVERSE MATRIX USING BY IMSL
C     SUBROUTINE LINV1F
C     *********************************************************************
      SUBROUTINE INV (NN,A,AINV,NOUT)
      INTEGER NN,IA,IDGT,IER
      REAL A(10,10),AINV(10,10),WKAREA(10),D(10,10)
      IA=10
      IDGT=3
      WRITE(NOUT,75)
      WRITE(NOUT,76)
      WRITE(NOUT,75)
      DO 1 I=1,NN
    1 WRITE(NOUT,10)(A(I,J),J=1,NN)
C
      CALL LINV1F(A,NN,IA,AINV,IDGT,WKAREA,IER)
      WRITE(NOUT,75)
      WRITE(NOUT,77)
      WRITE(NOUT,75)
      DO 2 I=1,NN
      WRITE(NOUT,10)(AINV(I,J),J=1,NN)
    2 WRITE(8,10)(AINV(I,J),J=1,NN)
   10 FORMAT(4(1X,E15.8))
   75 FORMAT(12X,'---------------------------------------------------------')
   76 FORMAT(12X,'***            INPUT  MATRIX                 ***')
   77 FORMAT(12X,'***          INVERSE MATRIX                  ***')
      RETURN
      END
C     *********************************************************************
C     THIS SUBROUTINE MATCHES INVERSE OF CONTROLLABILITY MATRIX WITH
C     T-TRANSPOSE MATRIX
C     *********************************************************************
      SUBROUTINE TTRA (A,TR,ID1,ID2,L,M,NOUT)
      REAL A(10,10),TR(10,10)
      REWIND 8
      I=ID1
      DO 1 J=1,L
    1 TR(1,J)=A(I,J)
      I=ID1+ID2
      DO 2 J=1,L
    2 TR(2,J)=A(I,J)
      WRITE(NOUT,75)
      WRITE(NOUT,76)
      WRITE(NOUT,75)
      DO 51 I=1,M
      WRITE(8,10)(TR(I,J),J=1,L)
   51 WRITE(NOUT,10)(TR(I,J),J=1,L)
```

```
   10 FORMAT(4(1X,E15.8))
   75 FORMAT(12X,'------------------------------------------------------------')
   76 FORMAT(12X,'***              TRANSPOSE MATRIX                      ***')
      RETURN
      END

C     **********************************************************
C     THIS SUBROUTINE COMPUTES THE T TRANSFORMATION MATRIX
C     **********************************************************
      SUBROUTINE TMAT (TR,AP1,AP2,AP3,TX,ID1,ID2,TI,TB,TC,L,M,NOUT)
      REAL TR(10,10),AP1(10,10),AP2(10,10),AP3(10,10)
      REAL T1(10,10),T2(10,10),T3(10,10),T4(10,10)
      REAL TTO(10,10),TON(10,10)
      REAL TX(10,10),TI(10,10),TB(10,10),TC(10,10)
      REWIND 8
      DO 1 I=1,M
      DO 1 J=1,L
    1 T1(I,J)=TR(I,J)
      DO 51 I=1,M
   51 WRITE(8,10)(T1(I,J),J=1,L)
      DO 2 I=1,M
      DO 2 J=1,L
      T2(I,J)=0.0
      DO 2 K=1,L
    2 T2(I,J)=T2(I,J)+TR(I,K)*AP1(K,J)
      DO 52 I=1,M
   52 WRITE(8,10)(T2(I,J),J=1,4)
      DO 3 I=1,M
      DO 3 J=1,L
      T3(I,J)=0.0
      DO 3 K=1,L
    3 T3(I,J)=T3(I,J)+TR(I,K)*AP2(K,J)
      DO 53 I=1,M
   53 WRITE(8,10)(T3(I,J),J=1,L)
      DO 4 I=1,M
      DO 4 J=1,L
      T4(I,J)=0.0
      DO 4 K=1,L
    4 T4(I,J)=T4(I,J)+TR(I,K)*AP3(K,J)
      DO 54 I=1,M
   54 WRITE(8,10)(T4(I,J),J=1,L)
      DO 5 I=1,M
      DO 5 J=1,L
      TTO(I,J)=T1(I,J)
      TTO(I+2,J)=T2(I,J)
      TTO(I+4,J)=T3(I,J)
    5 TTO(I+6,J)=T4(I,J)
      DO 55 I=1,L*M
   55 WRITE(8,10)(TTO(I,J),J=1,L)
C
      DO 6 K=1,L
      I=2*K-1
      DO 6 J=1,L
    6 TON(K,J)=TTO(I,J)
```

```
      DO 7 K=1,L
      I=2*K
      DO 7 J=1,L
    7 TON(K+L,J)=TTO(I,J)
      WRITE(NOUT,75)
      WRITE(NOUT,76)
      WRITE(NOUT,75)
      DO 56 I=1,L*M
      WRITE(8,10)(TON(I,J),J=1,L)
   56 WRITE(NOUT,10)(TON(I,J),J=1,L)
C
C     ID1,ID2
      DO 8 I=1,ID1
      DO 8 J=1,L
    8 TX(I,J)=TON(I,J)
      DO 9 I=1,ID2
      DO 9 J=1,L
    9 TX(I+ID1,J)=TON(I+L,J)
      WRITE(NOUT,75)
      WRITE(NOUT,77)
      WRITE(NOUT,75)
      DO 57 I=1,L
      WRITE(8,10)(TX(I,J),J=1,L)
   57 WRITE(NOUT,10)(TX(I,J),J=1,L)
C
      DO 11 I=1,L
      DO 11 J=1,L
      TI(I,J)=TX(I,J)
      TC(I,J)=TX(I,J)
   11 TB(I,J)=TX(I,J)
      DO 58 I=1,L
      WRITE(8,*)(TI(I,J),J=1,L)
      WRITE(8,*)(TC(I,J),J=1,L)
   58 WRITE(8,*)(TB(I,J),J=1,L)
   10 FORMAT(4(1X,E15.8))
   75 FORMAT(12X,'-----------------------------------------------------')
   76 FORMAT(12X,'***TRANSFORMED MATRIX (T,TA) (T1,T1A FORM)***')
   77 FORMAT(12X,'***                 T-MATRIX                 ***')
      RETURN
      END
C     ***************************************************************
C     THIS SUBROUTINE COMPUTES THE TRANSFORMED MATRICES IN
C     PHASE-VARIABLE CANONICAL FORM
C     ***************************************************************
      SUBROUTINE ATRA(TI,A1,TINV,AT,B1,BT,L,M,NOUT)
      REAL TI(10,10),A1(10,10),TINV(10,10),AT(10,10)
      REAL B1(10,10),BT(10,10),AMT(10,10)
      REWIND 8
      DO 1 I=1,L
      DO 1 J=1,L
      AMT(I,J)=0.0
      DO 1 K=1,L
    1 AMT(I,J)=AMT(I,J)+TI(I,K)*A1(K,J)
C     PRINT*,'A*T'
      DO 51 I=1,L
   51 WRITE(8,10)(AMT(I,J),J=1,4)
      DO 2 I=1,L
      DO 2 J=1,L
```

```
      AT(I,J)=0.0
      DO 2 K=1,L
    2 AT(I,J)=AT(I,J)+AMT(I,K)*TINV(K,J)
      WRITE(NOUT,75)
      WRITE(NOUT,76)
      WRITE(NOUT,75)
      DO 52 I=1,L
      WRITE(8,10)(AT(I,J),J=1,L)
   52 WRITE(NOUT,10)(AT(I,J),J=1,L)
      DO 3 I=1,L
      DO 3 J=1,M
      BT(I,J)=0.0
      DO 3 K=1,L
    3 BT(I,J)=BT(I,J)+TI(I,K)*B1(K,J)
      WRITE(NOUT,75)
      WRITE(NOUT,77)
      WRITE(NOUT,75)
      DO 53 I=1,L
      WRITE(8,10)(BT(I,J),J=1,M)
   53 WRITE(NOUT,10)(BT(I,J),J=1,M)
   10 FORMAT(4(1X,E15.8))
   75 FORMAT(12X,'------------------------------------------------------------')
   76 FORMAT(12X,'***       A-TRANSFORMED MATRIX  (A*T*AINV) ***')
   77 FORMAT(12X,'***         B-TRANSFORMED MATRIX             ***')
      RETURN
      END

C     ***************************************************************
C     THIS SUBROUTINE COMPUTES COEFFICIENTS OF CHARACTERISTIC
C     EQUATION
C     ***************************************************************
      SUBROUTINE SEIG (AT,BT,SA,SB,ID1,L,M,NOUT)
      REAL AT(10,10),BT(10,10),SA(10,10),SB(10,10)
      REWIND 8
      DO 1 J=1,L
      SA(1,J)=AT(ID1,J)
    1 SA(2,J)=AT(L,J)
      DO 2 J=1,M
      SB(1,J)=BT(ID1,J)
    2 SB(2,J)=BT(4,J)
      WRITE(NOUT,75)
      WRITE(NOUT,76)
      WRITE(NOUT,75)
      DO 51 I=1,M
      WRITE(8,10)(SA(I,J),J=1,L)
   51 WRITE(NOUT,10)(SA(I,J),J=1,L)
C     PRINT*,'INPUT B-MATRIX OF SYSTEM'
      DO 52 I=1,M
   52 WRITE(8,10)(SB(I,J),J=1,M)
   10 FORMAT(4(1X,E15.8))
   75 FORMAT(12X,'------------------------------------------------------------')
   76 FORMAT(12X,'***COEFF. OF CHARACTERISTIC MATRIX(SYSTEM)***')
      RETURN
      END
```

```
C     ****************************************************
C     THIS SUBROUTINE ARRANGES THE DESIRED MATRIX
C     ****************************************************
      SUBROUTINE DEIG (NAD,ID1,L,M,DX1,DX2,DX3,DX4,NOUT)
      REAL AD(10,10),NAD(10,10)
      REWIND 8
C     LAM1,2=DX1+IDX2   LAM3=DX3    LAM4=DX4
      P1=-2*DX1-(DX3+DX4)
      P2=(DX1**2+DX2**2)+2*DX1*(DX3+DX4)+DX3*DX4
      P3=-(DX3+DX4)*(DX1**2+DX2**2)-2*DX1*DX3*DX4
      P4=(DX1**2+DX2**2)*DX3*DX4
      DO 1 I=1,3
    1 AD(I,I+1)=1.
      AD(4,1)=-P4
      AD(4,2)=-P3
      AD(4,3)=-P2
      AD(4,4)=-P1
      WRITE(NOUT,75)
      WRITE(NOUT,76)
      WRITE(NOUT,75)
      DO 51 I=1,L
      WRITE(8,10)(AD(I,J),J=1,L)
   51 WRITE(NOUT,10)(AD(I,J),J=1,L)
      DO 2 J=1,L
      NAD(1,J)=AD(ID1,J)
    2 NAD(2,J)=AD(4,J)
      WRITE(NOUT,75)
      WRITE(NOUT,77)
      WRITE(NOUT,75)
      DO 52 I=1,M
      WRITE(8,10)(NAD(I,J),J=1,L)
   52 WRITE(NOUT,10)(NAD(I,J),J=1,L)
   10 FORMAT(4(1X,E15.8))
   75 FORMAT(12X,'------------------------------------------------------')
   76 FORMAT(12X,'***          DESIRED MATRIX                    ***')
   77 FORMAT(12X,'***    COEFFICIENT OF DESIRED MATRIX           ***')
      RETURN
      END

C     ****************************************************
C     THIS SUBROUTINES COMPUTES THE GAIN MATRIX
C     ****************************************************
      SUBROUTINE GAIN (NAD,SA,BTINV,C1,TC,KK1,L,M,NOUT)
      REAL NAD(10,10),SA(10,10),TA(10,10),C1(10,10)
      REAL TC(10,10),KK1(10,10),BTINV(10,10)
      REWIND 8
      DO 1 I=1,M
      DO 1 J=1,L
    1 TA(I,J)=NAD(I,J)-SA(I,J)
      WRITE(NOUT,75)
      WRITE(NOUT,76)
      WRITE(NOUT,75)
      DO 51 I=1,M
      WRITE(8,10)(TA(I,J),J=1,L)
   51 WRITE(NOUT,10)(TA(I,J),J=1,L)
```

```
      DO 2 I=1,M
      DO 2 J=1,L
      C1(I,J)=0
      DO 2 K=1,M
    2 C1(I,J)=C1(I,J)+BTINV(I,K)*TA(K,J)
      WRITE(NOUT,75)
      WRITE(NOUT,77)
      WRITE(NOUT,75)
      DO 54 I=1,M
      WRITE(8,10) (C1(I,J),J=1,L)
   54 WRITE(NOUT,10) (C1(I,J),J=1,L)
C     CALL GAIN
      DO 3 I=1,M
      DO 3 J=1,L
      KK1(I,J)=0
      DO 3 K=1,L
    3 KK1(I,J)=KK1(I,J)+C1(I,K)*TC(K,J)
      WRITE(NOUT,75)
      WRITE(NOUT,78)
      WRITE(NOUT,75)
      DO 55 I=1,M
      WRITE(8,10) (KK1(I,J),J=1,L)
   55 WRITE(NOUT,10) (KK1(I,J),J=1,L)
   10 FORMAT(4(1X,E15.8))
   75 FORMAT(12X,'-------------------------------------------------------------------')
   76 FORMAT(12X,'***      DIFFERENCE OF EIGEN VALUES              ***')
   77 FORMAT(12X,'***             GAIN MATRIX                      ***')
   78 FORMAT(12X,'***       TRANSFORMED GAIN MATRIX                ***')
      RETURN
      END

C     **********************************************************
C     THIS SUBROUTINE COMPUTES THE CLOSED-LOOP MATRIX
C     **********************************************************
      SUBROUTINE  COE (A2,B2,KK1,EI,L,M,NOUT)
      REAL A2(10,10),B2(10,10),CC(10,10),KK1(10,10)
      REAL EI(10,10)
      COMMON/COEF1/CA11,CA12,CA13,CA14
      COMMON/COEF2/CA21,CA22,CA23,CA24
      COMMON/COEF3/CA31,CA32,CA33,CA34
      COMMON/COEF4/CA41,CA42,CA43,CA44
      REWIND 8
      DO 1 I=1,L
      DO 1 J=1,L
      CC(I,J)=0.
      DO 1 K=1,M
    1 CC(I,J)=CC(I,J)+B2(I,K)*KK1(K,J)
      DO 5 I=1,L
    5 WRITE(8,10)(CC(I,J),J=1,L)
      EI(I,J)=0.
      DO 2 I=1,L
      DO 2 J=1,L
    2 EI(I,J)=A2(I,J)+CC(I,J)
      CA11=EI(1,1)
      CA12=EI(1,2)
```

```
      CA13=EI(1,3)
      CA14=EI(1,4)
      CA21=EI(2,1)
      CA22=EI(2,2)
      CA23=EI(2,3)
      CA24=EI(2,4)
      CA31=EI(3,1)
      CA32=EI(3,2)
      CA33=EI(3,3)
      CA34=EI(3,4)
      CA41=EI(4,1)
      CA42=EI(4,2)
      CA43=EI(4,3)
      CA44=EI(4,4)
      WRITE(NOUT,75)
      WRITE(NOUT,76)
      WRITE(NOUT,75)
      DO 3 I=1,L
      WRITE(8,10)(EI(I,J),J=1,L)
    3 WRITE(NOUT,10)(EI(I,J),J=1,L)
   10 FORMAT(4(1X,E15.8))
   75 FORMAT(12X,'-------------------------------------------------------------')
   76 FORMAT(12X,'***CLOSED LOOP MATRIX AFTER FEEDBACK(A+BK)***')
      RETURN
      END
C     ****************************************
C     THIS SUBROUTINES COMPUTES THE EIGENVALUES
C     USING IMSL SUBROUTINE EIGRF
C     ****************************************
      SUBROUTINE EIGEN(EIG,L,M,NOUT)
      REAL EIG(10,10),WK(24),RZ(32),RW(8)
      COMPLEX W(8),Z(8,8),ZN
      EQUIVALENCE (W(1),RW(1)), (Z(1,1),RZ(1))
      REWIND 8
      NN=L
      IA=10
      WRITE(NOUT,75)
      WRITE(NOUT,76)
      WRITE(NOUT,75)
      DO 6 I=1,NN
      WRITE(NOUT,101)(EIG(I,J),J=1,NN)
    6 CONTINUE
  101 FORMAT(4(1X,E15.8))
C     COMPUTE THE EIGENVALUES/VECTORS OF A
   10 CONTINUE
      IJOB=2
      CALL EIGRF(EIG,NN,IA,IJOB,W,Z,IA,WK,IER)
C     WRITE THE EIGENVALUES/VECTORS OF A
      WRITE(6,78)
      WRITE(6,77)
      WRITE(6,78)
      WRITE(6,150)
  150 FORMAT(14H EIGENVALUE NO,10X,2HRE,16X,2HIM)
```

```
      DO 20 I=1,NN
      I2=2*I
      WRITE(6,200)I,RW(I2-1),RW(I2)
   20 CONTINUE
  200 FORMAT(10X,I2,2F18.6)
   75 FORMAT(12X,'---------------------------------------------------------')
   76 FORMAT(12X,'***              INPUT MATRIX                       ***')
   77 FORMAT(12X,'***             EIGEN SOLUTION                      ***')
   78 FORMAT(12X,'---------------------------------------------------------')
      PRINT*,'  '
      PRINT*,'*CHECK THE EIGEN VALUES OF SYSTEM'
      RETURN
      END
C     ***************************************************************
C     THIS SUBROUTINE SOLVES THE OPEN-LOOP SYSTEM EQUATION
C     USING IMSL SUBROUTINE DVERK
C     ***************************************************************
      SUBROUTINE RUN1(L,CL,FI,IGRA,IPLT,NOUT)
      PARAMETER (KK=1000)
      EXTERNAL FCN1
      REAL C(24),W1(4,100),T,TOL,TEND
      REAL X(KK),Y1(KK)
      REAL X1(KK),X2(KK),X3(KK),X4(KK)
      REAL LE,MA,KS
      INTEGER L,IND,NW,IER,K
      COMMON/QWER/CXX,CXY,CYX,CYY,MA,KS
      DT=0.01
      KMAX=30
      T=0
      NW=L
      TOL=0.00001
      IND=1
C
C     INITIAL VALUES FOR DIFFERENTIAL EQUATION
      X(1)=CL*SIN(FI)
      X(2)=0.0
      X(3)=CL*COS(FI)
      X(4)=0.0
      WRITE(NOUT,4)
      WRITE(NOUT,5)
      WRITE(NOUT,4)
      WRITE(NOUT,2)
      Y1(1)=0.0
      X1(1)=CL*SIN(FI)
      X2(1)=0.0
      X3(1)=CL*COS(FI)
      X4(1)=0.0
      WRITE(NOUT,3)Y1(1),X1(1),X2(1),X3(1),X4(1)
      WRITE(8,20)Y1(1),X1(1)
      WRITE(9,20)Y1(1),X2(1)
      WRITE(10,20)Y1(1),X3(1)
      WRITE(11,20)Y1(1),X4(1)
```

```
      DO 1 K=1,KMAX
      TEND=FLOAT(K)*DT
      CALL DVERK(L,FCN1,T,X,TEND,TOL,IND,C,NW,W1,IER)
      WRITE(NOUT,3)TEND,(X(I),I=1,L)
      Y1(K+1)=TEND
      X1(K+1)=X(1)
      X2(K+1)=X(2)
      X3(K+1)=X(3)
      X4(K+1)=X(4)
      WRITE(8,20)Y1(K+1),X1(K+1)
      WRITE(9,20)Y1(K+1),X2(K+1)
      WRITE(10,20)Y1(K+1),X3(K+1)
      WRITE(11,20)Y1(K+1),X4(K+1)
    1 CONTINUE
    2 FORMAT(3X,'T',11X,'X(1)',13X,'X(2)',13X,'X(3)',13X,'X(4)')
    3 FORMAT(F6.3,4(4X,F13.11))
    4 FORMAT(12X,'-----------------------------------------------------------')
    5 FORMAT(12X,'***       OPEN LOOP STATES OF SYSTEM        ***')
   20 FORMAT(2F15.10)
      REWIND 8
      REWIND 9
      REWIND 10
      REWIND 11
      IF(IGRA.EQ.0) GO TO 25
      CALL GRAPH (Y1,X1,KK,KMAX,IPLT)
      CALL GRAPH (Y1,X2,KK,KMAX,IPLT)
      CALL GRAPH (Y1,X3,KK,KMAX,IPLT)
      CALL GRAPH (Y1,X4,KK,KMAX,IPLT)
   25 RETURN
      END
C     *********************************************************
C     THIS SUBROUTINE IS USED FOR WRITING SYSTEM EQUATION
C     *********************************************************
      SUBROUTINE FCN1(L,T,X,XPRIME)
      INTEGER NN
      REAL X(L),XPRIME(L),T
      REAL MA,KS
      COMMON/QWER/CXX,CXY,CYX,CYY,MA,KS
      XPRIME(1)=X(2)
      XPRIME(2)=(-KS*X(1)-CXX*X(2)-CXY*X(4))/MA
      XPRIME(3)=X(4)
      XPRIME(4)=(-CXY*X(2)-KS*X(3)-CYY*X(4))/MA
      RETURN
      END
C     *************************************************************
C     THIS SUBROUTINE SOLVES THE CLOSED-LOOP SYSTEM EQUATION
C     USING IMSL SUBROUTINE DVERK
C     *************************************************************
      SUBROUTINE RUN2 (L,CL,FI,IGRA,IPLT,NOUT)
```

```
      PARAMETER (KK=1000)
      EXTERNAL FCN2
      COMMON/COEF1/CA11,CA12,CA13,CA14
      COMMON/COEF2/CA21,CA22,CA23,CA24
      COMMON/COEF3/CA31,CA32,CA33,CA34
      COMMON/COEF4/CA41,CA42,CA43,CA44
      REAL X(KK),Y1(KK)
      REAL X1(KK),X2(KK),X3(KK),X4(KK)
      REAL C(24),W1(4,100),T,TOL,TEND
      INTEGER L,IND,NW,IER,K
      T=0
      NW=L
      KMAX=5000
C     INITIAL VALUES FOR DIFFERENTIAL EQUATION
      X(3)=CL*SIN(FI)
      X(2)=0.0
      X(1)=CL*COS(FI)
      X(4)=0.0
      TOL=0.00001
      DT=0.1
      IND=1
      WRITE(NOUT,4)
      WRITE(NOUT,5)
      WRITE(NOUT,4)
      WRITE(NOUT,2)
      Y1(1)=0.0
      X1(1)=CL*SIN(FI)
      X2(1)=0.0
      X3(1)=CL*COS(FI)
      X4(1)=0.0
      WRITE(NOUT,3)Y1(1),X1(1),X2(1),X3(1),X4(1)
      WRITE(8,20)Y1(1),X1(1)
      WRITE(9,20)Y1(1),X2(1)
      WRITE(10,20)Y1(1),X3(1)
      WRITE(11,20)Y1(1),X4(1)
      DO 1 K=1,KMAX
      TEND=FLOAT(K)*DT
      CALL DVERK(L,FCN2,T,X,TEND,TOL,IND,C,NW,W1,IER)
      WRITE(NOUT,3)TEND,(X(I),I=1,L)
      Y1(K+1)=TEND
      X1(K+1)=X(1)
      X2(K+1)=X(2)
      X3(K+1)=X(3)
      X4(K+1)=X(4)
      WRITE(8,20)Y1(K+1),X1(K+1)
      WRITE(9,20)Y1(K+1),X2(K+1)
      WRITE(10,20)Y1(K+1),X3(K+1)
      WRITE(11,20)Y1(K+1),X4(K+1)
    1 CONTINUE
    2 FORMAT(3X,'T',11X,'X(1)',13X,'X(2)',13X,'X(3)',13X,'X(4)')
    3 FORMAT(F6.3,4(4X,F13.11))
    4 FORMAT(12X,'--------------------------------------------------')
    5 FORMAT(12X,'** CLOSED LOOP STATES AFTER STATE FEEDBACK **')
   20 FORMAT(2F15.10)
```

```
      REWIND 8
      REWIND 9
      REWIND 10
      REWIND 11
      IF(IGRA.EQ.0) GO TO 25
      CALL GRAPH (Y1,X1,KK,KMAX,IPLT)
      CALL GRAPH (Y1,X2,KK,KMAX,IPLT)
      CALL GRAPH (Y1,X3,KK,KMAX,IPLT)
      CALL GRAPH (Y1,X4,KK,KMAX,IPLT)
   25 RETURN
      END
C     ******************************************************
C     THIS SUBROUTINE IS USED FOR WRITING SYSTEM EQUATION
C     ******************************************************
      SUBROUTINE FCN2 (L,T,X,XPRIME)
      COMMON/COEF1/CA11,CA12,CA13,CA14
      COMMON/COEF2/CA21,CA22,CA23,CA24
      COMMON/COEF3/CA31,CA32,CA33,CA34
      COMMON/COEF4/CA41,CA42,CA43,CA44
      INTEGER L
      REAL X(L),XPRIME(L),T
      XPRIME(1)=CA11*X(1)+CA12*X(2)+CA13*X(3)+CA14*X(4)
      XPRIME(2)=CA21*X(1)+CA22*X(2)+CA23*X(3)+CA24*X(4)
      XPRIME(3)=CA31*X(1)+CA32*X(2)+CA33*X(3)+CA34*X(4)
      XPRIME(4)=CA41*X(1)+CA42*X(2)+CA43*X(3)+CA44*X(4)
      RETURN
      END

C     ********************************
C     THIS SUBROUTINE DRAWS GRAPHICS
C     ********************************
      SUBROUTINE GRAPH(X,Y,KK,KMAX,IPLT)
      DIMENSION X(KK),Y(KK)
      REWIND 8
      REWIND 9
      REWIND 10
      REWIND 11
      IPLT=0.
      SIZEX=0.08
      SIZEY=0.04
      XMAX=YMAX=0.0
      DO 1000 J=1,KMAX
      IF(XMAX.LT.ABS(X(J)))XMAX=ABS(X(J))
      IF(YMAX.LT.ABS(Y(J)))YMAX=ABS(Y(J))
 1000 CONTINUE
      IF(XMAX.EQ.0)XMAX=1.0
      IF(YMAX.EQ.0)YMAX=1.0
      DT=SIZEX/XMAX*10.0
      DY=SIZEY/YMAX*10.0
      CALL INITIG(.TRUE.,.TRUE.,4HNDF1)
      CALL AXIS(SIZEX,SIZEY)
```

```
      CALL G(DT,DY,X,Y,KK,KMAX)
      CALL PROMPT(15,'STATE VARIATION')
      CALL POSCUR(D,4)
      CALL PROMPT(13,'ENTER ANY KEY')
      CALL PROMPT(13,'  TO CONTINUE')
      CALL AUTKEY(1,IJR,1,NC,IC)
      IF(IPLT.EQ.1)THEN
      CALL UNION
      CALL AXIS(SIZEX,SIZEY)
      CALL G(DT,DY,X,Y,KK,KMAX)
      CALL UNIOFF
      END IF
      CALL CLRSVP
      CALL QUITIG(.TRUE.)
      RETURN
      END
C     ******************************
C     THIS SUBROUTINE DRAWS AXISES
C     ******************************
      SUBROUTINE AXIS(SIZEX,SIZEY)
C     DRAWING AXIS (X,Y)
      CALL MOVEA(.01,.0)
      CALL DRAWA(.01,.95)
      CALL MOVEA(.01,.5)
      CALL DRAWA(.9,.5)
C     DRAWING ARROWS ON X AXIS
      CALL MOVEA(.85,.5)
      CALL DRAWA(.83,.51)
      CALL DRAWA(.83,.49)
      CALL DRAWA(.85,.5)
C     DRAWING ARROWS ON Y AXIS
      CALL MOVEA(.01,.95)
      CALL DRAWA(.0,.93)
      CALL DRAWA(.02,.93)
      CALL DRAWA(.01,.95)
C     WRITING VARIABLE ON X AXIS
      CALL MOVEA (.85,.5)
      CALL TEXT(10,10H  TIME     )
C     WRITING VARIABLE ON Y AXIS
      CALL MOVEA (.01,.97)
      CALL TEXT(10,10H STATE     )
C     SCALING ON X AXIS
      DO 1 XX=.01,.80,.07
      CALL MOVEA(XX,.51)
      CALL DRAWA(XX,.49)
    1 CONTINUE
C     SCALING ON +Y AXIS
      CALL MOVEA(.01,.5)
      DO 2 YY=.5,.90,.07
      CALL MOVEA(.0,YY)
      CALL DRAWA(.02,YY)
```

```
    2   CONTINUE
C       SCALING ON -Y AXIS
        CALL MOVEA(.01,.5)
        DO 3 YY=.5,.02,-.07
        CALL MOVEA(.0,YY)
        CALL DRAWA(.02,YY)
    3   CONTINUE
C
        CALL SMSTYL(1)
C       RETURN THE AXIS ORIGION
        CALL MOVEA(.01,.5)
        RETURN
        END

C       ********************************
C       THIS SUBROUTINE DRAWS THE LINES
C       ********************************
        SUBROUTINE G(DT,DY,X,Y,KK,KMAX)
        DIMENSION X(KK),Y(KK)
        REWIND 8
        REWIND 9
        REWIND 10
        REWIND 11
        DDY=DY*Y(1)
        DDT=DT*X(1)
        CALL DRAWR(DDT,DDY)
        DO 7 J=2,KMAX+1
        DDT=DT*(X(J)-X(J-1))
        DDY=DY*(Y(J)-Y(J-1))
        CALL DRAWR(DDT,DDY)
    7   CONTINUE
        RETURN
        END
```

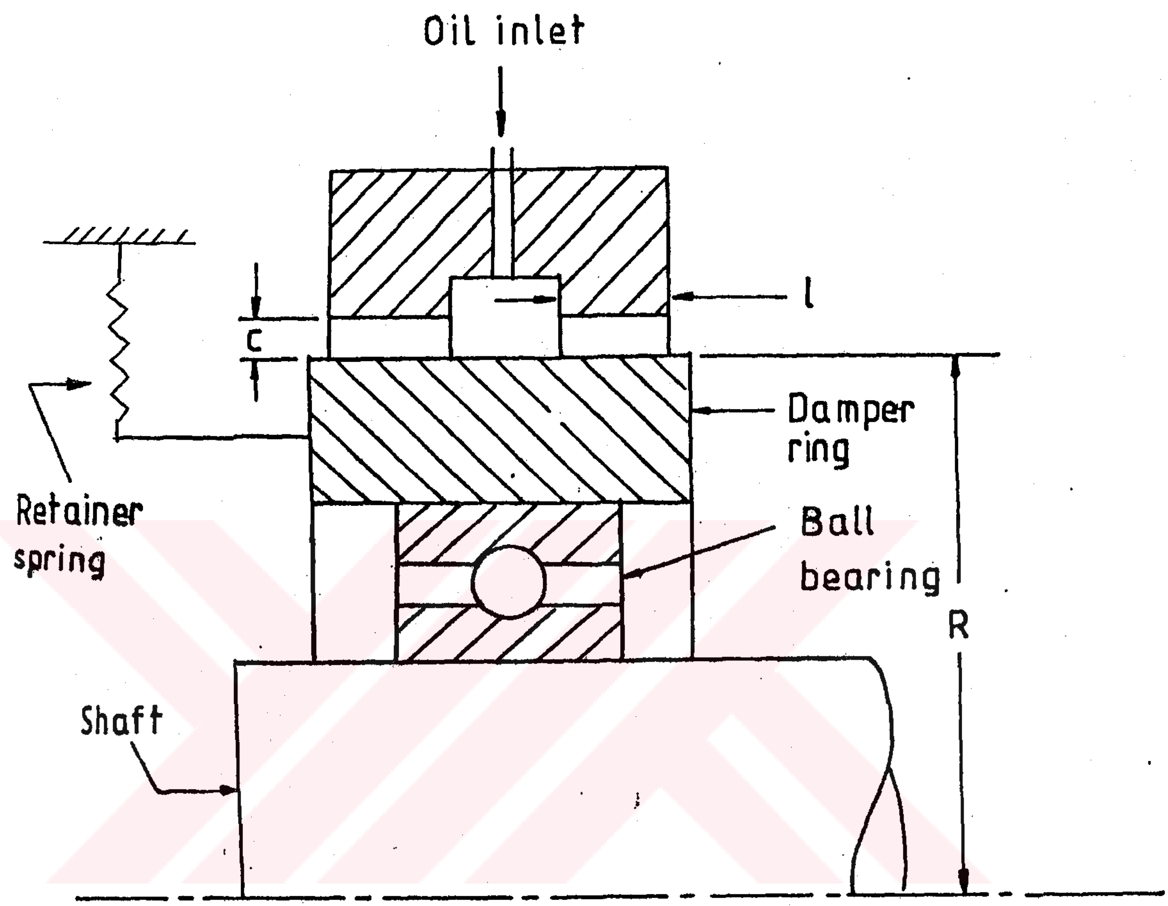


Figure 1.1.1 Schematic representation of squeeze-film bearing in practical application.

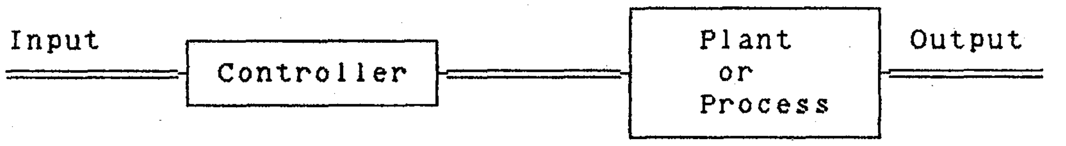


Figure 2.1.1 Open-loop control system.

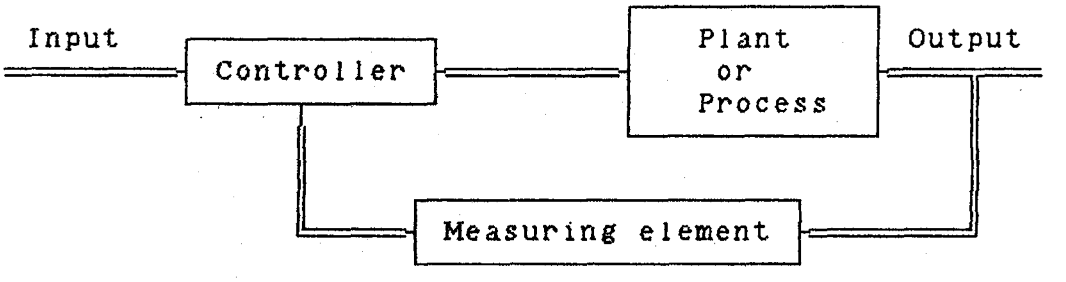


Figure 2.1.2 Closed-loop control system.

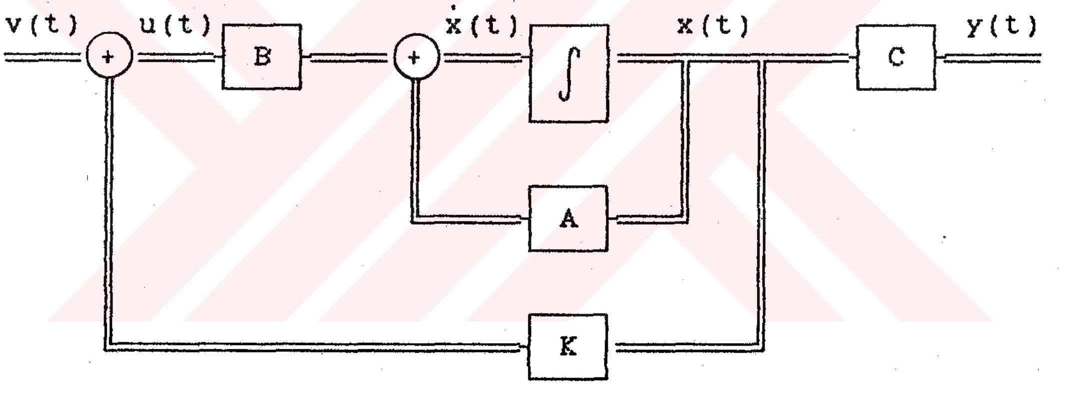


Figure 2.1.3 General structure of the linear state-variable feedback.

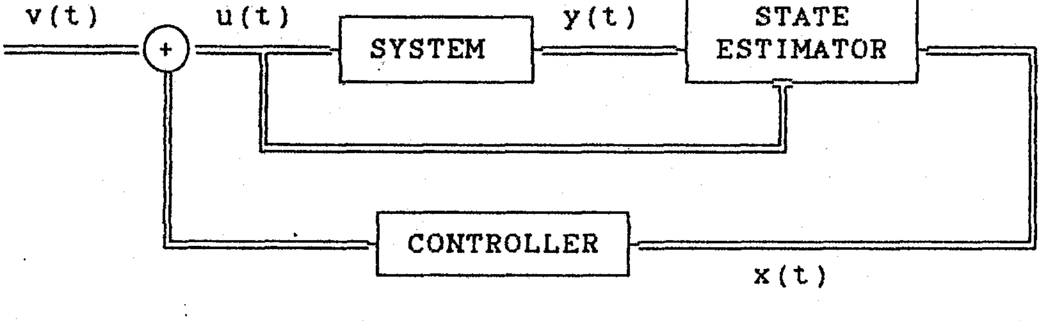
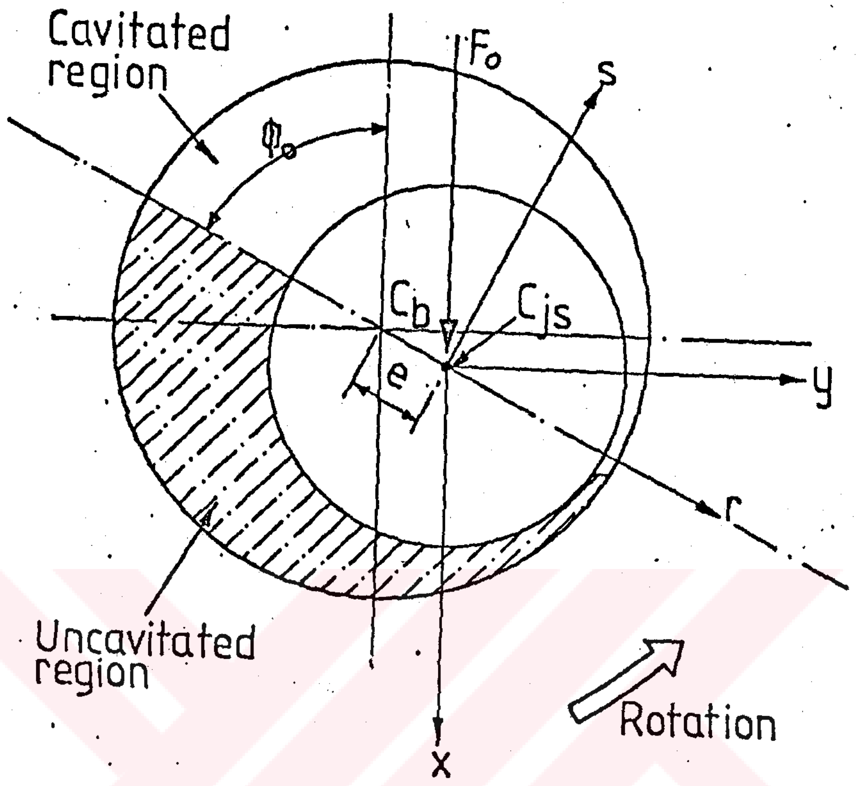


Figure 2.1.4 The block diagram of the system with an estimator.



C_b : Bearing Centre

C_{js} : Journal Centre

F_o : Static Load

Figure 3.2.1 Coordinate system.

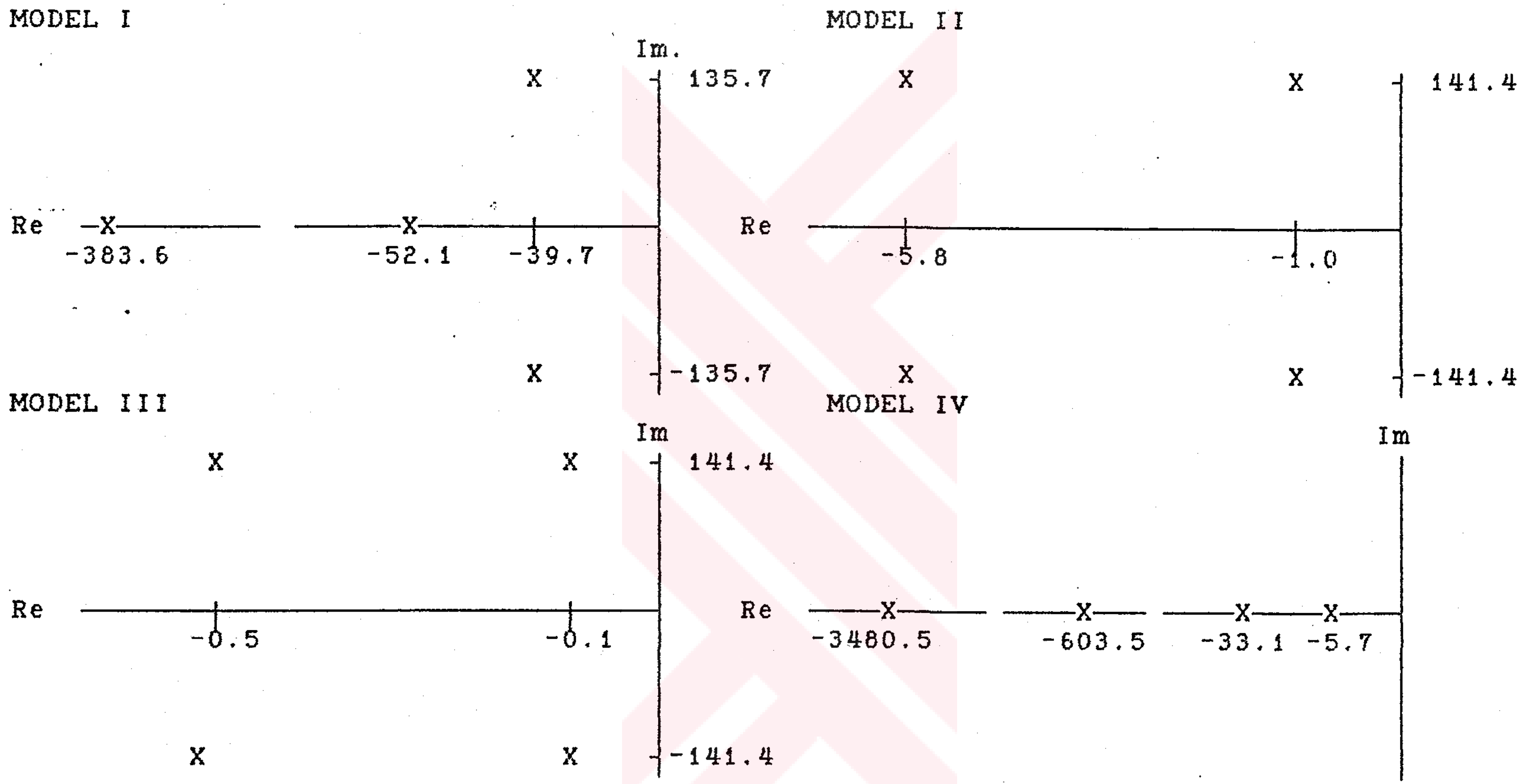


Figure 4.1.1 Eigenvalues of selected models.

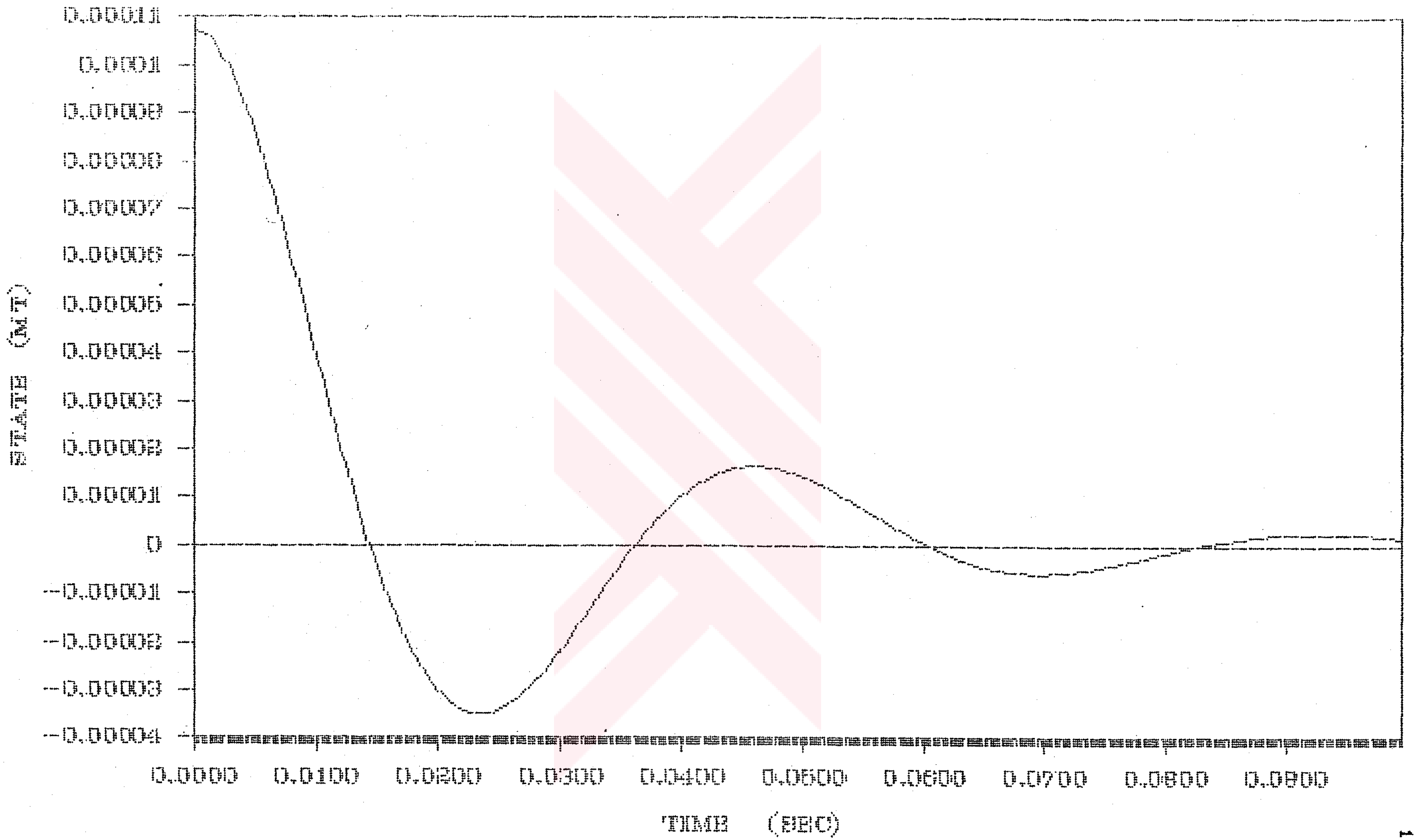


Figure 4.1.2 Free response of Model I in the x-direction

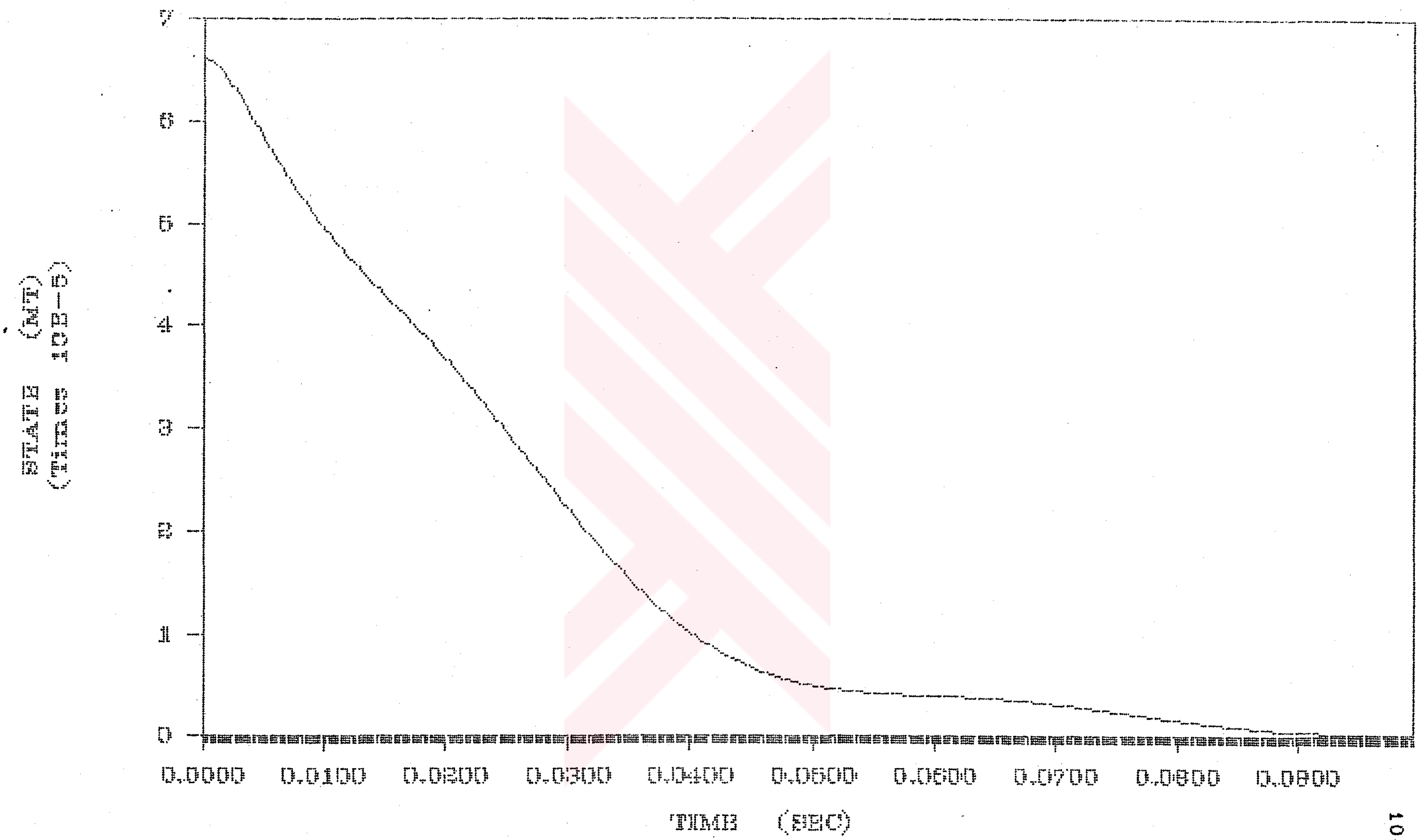


Figure 4.1.3 Free response of Model I in the y-direction

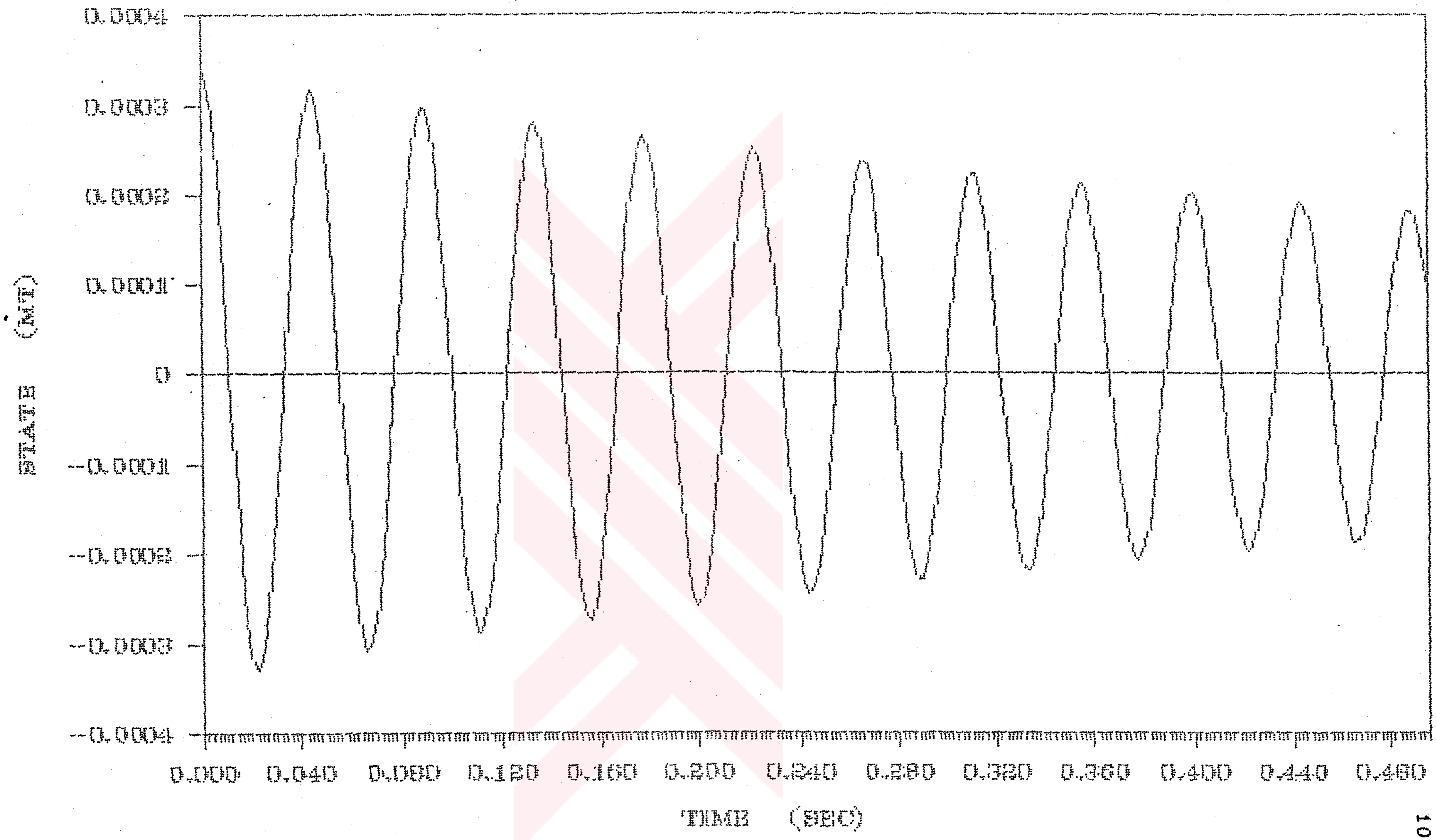


Figure 4.1.4 Free response of Model II in the x-direction

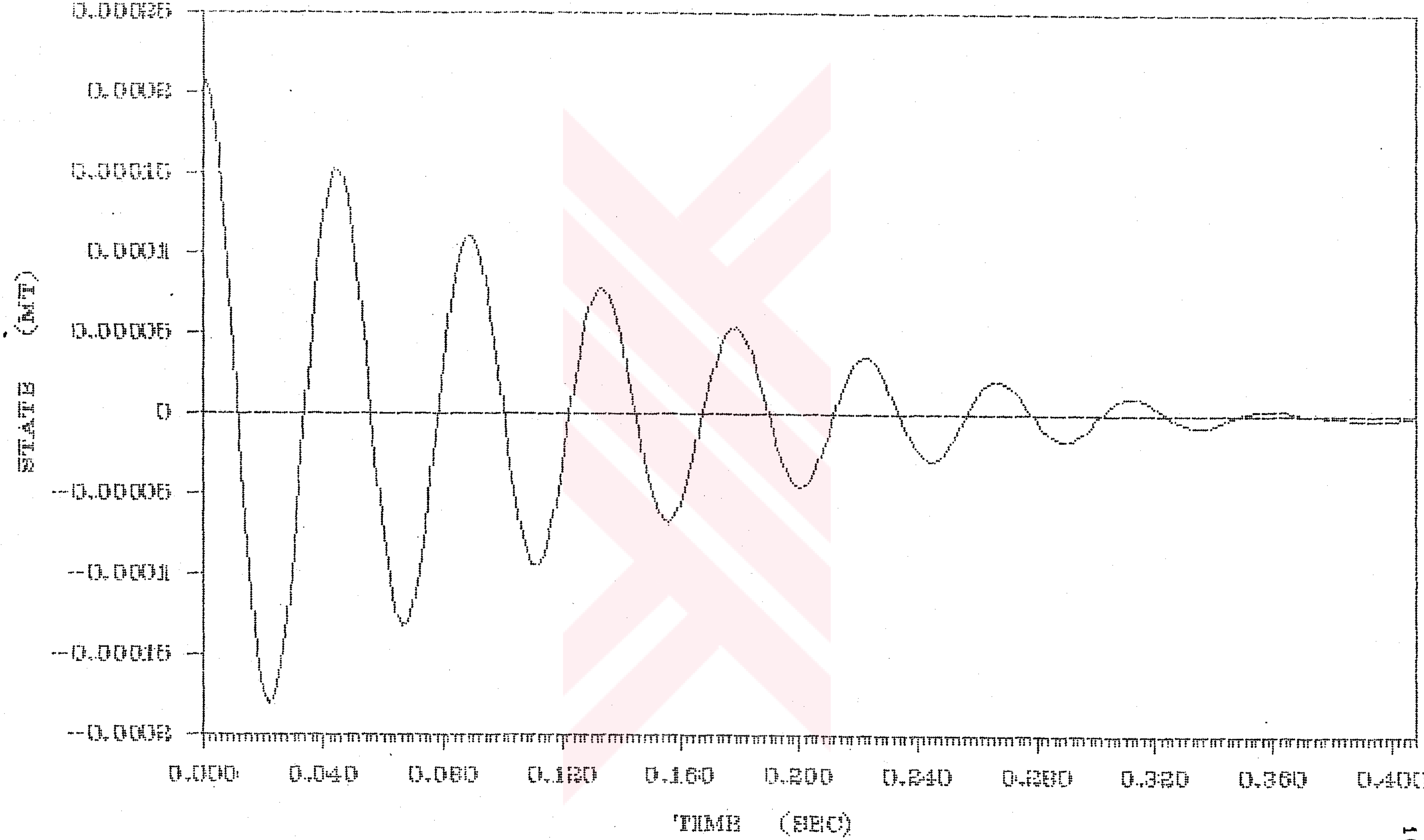


Figure 4.1.5 Free response of Model II in the y-direction

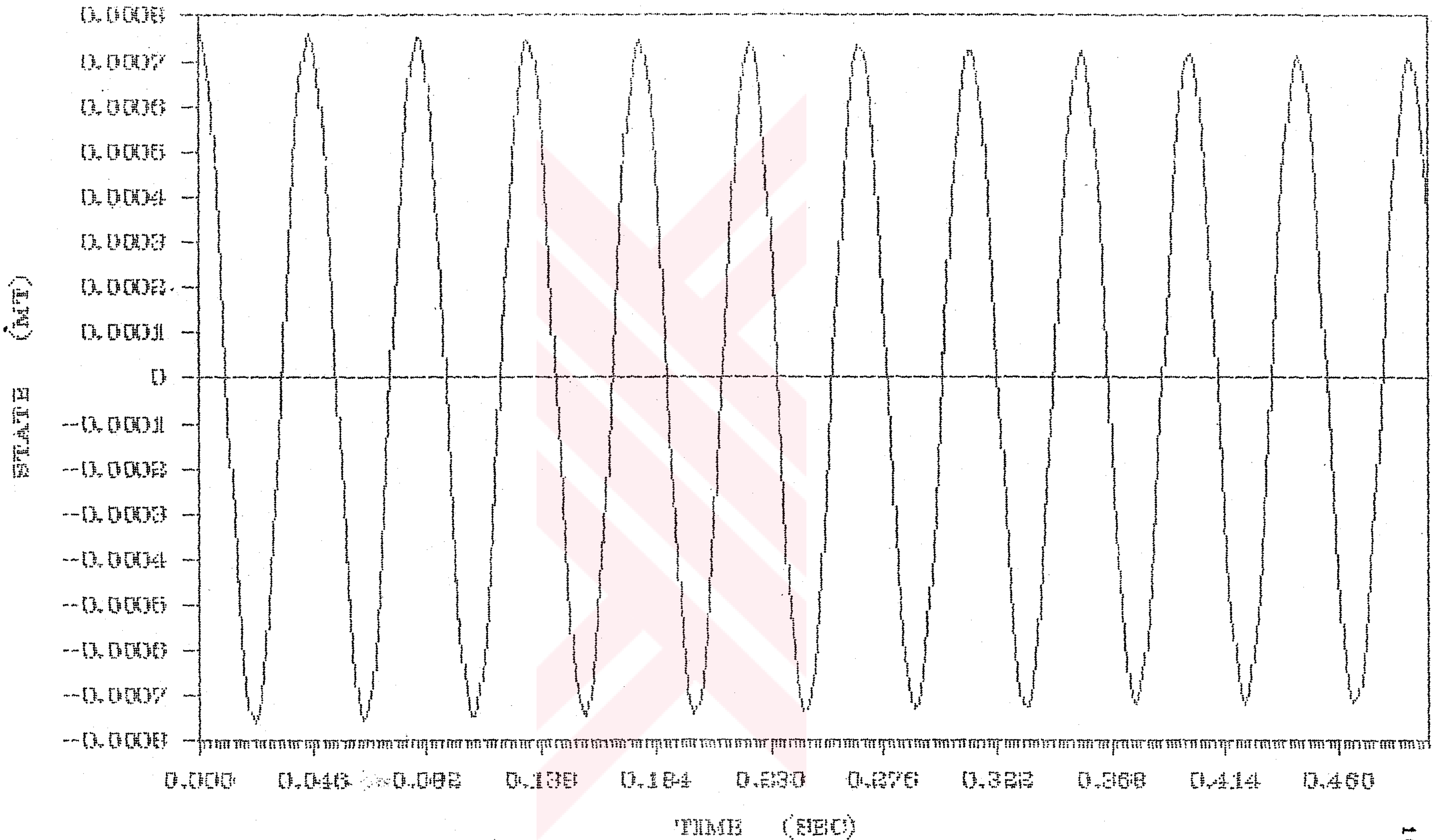


Figure 4.1.6 Free response of Model III in the x-direction.

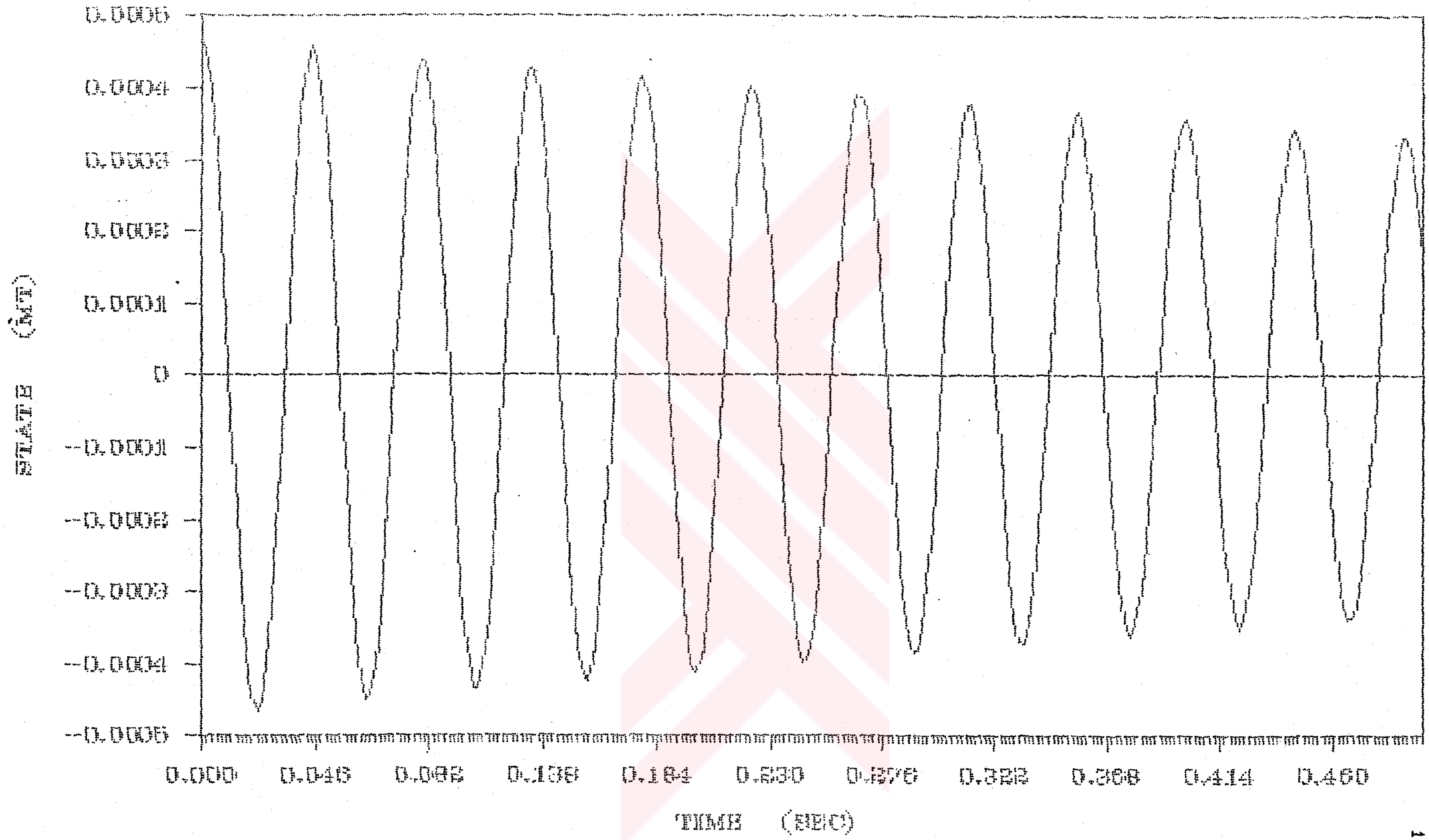


Figure 4.1.7 Free response of Model III in the y-direction

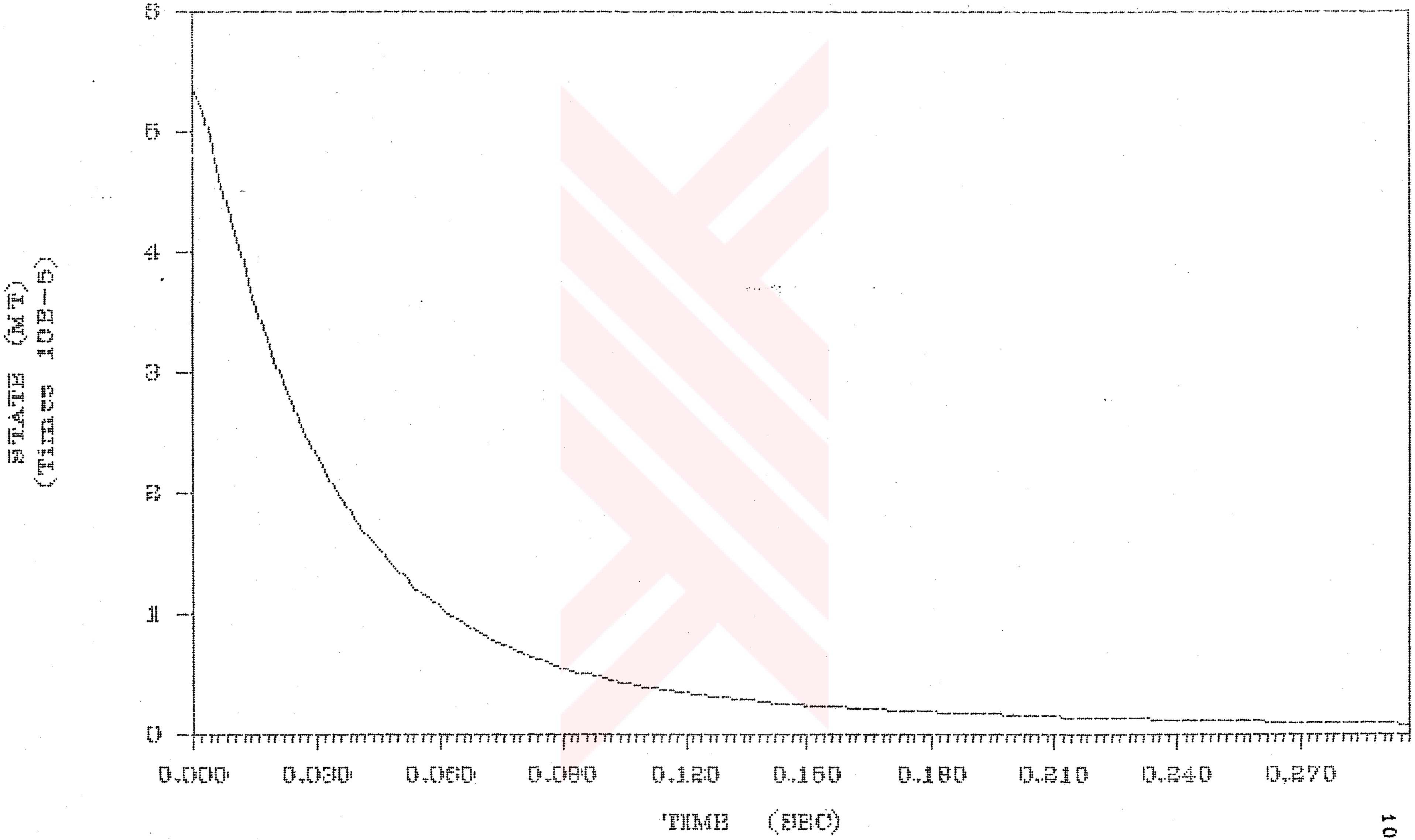


Figure 4.1.8 Free response of Model IV in the x-direction

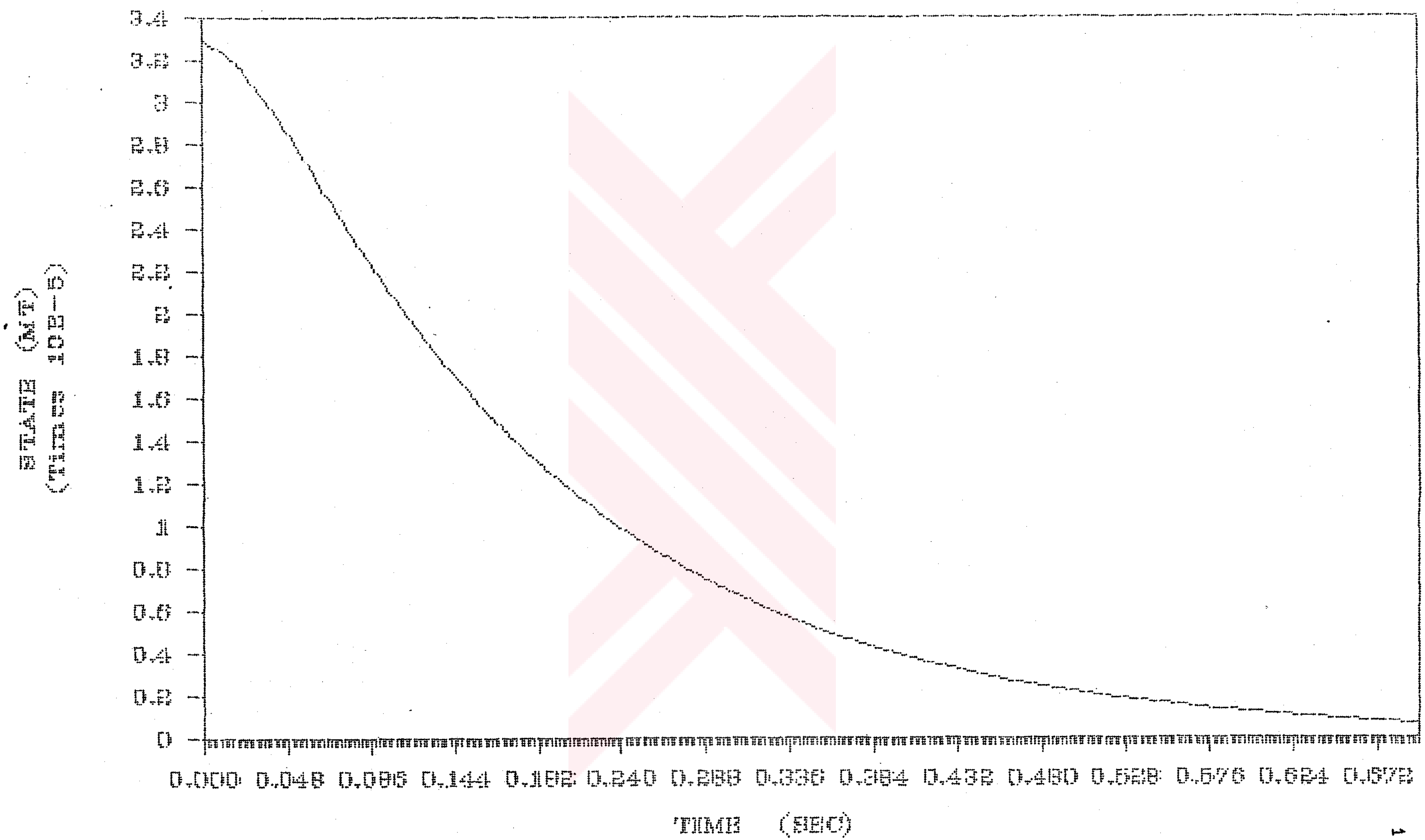


Figure 4.1.9 Free response of Model IV in the y-direction

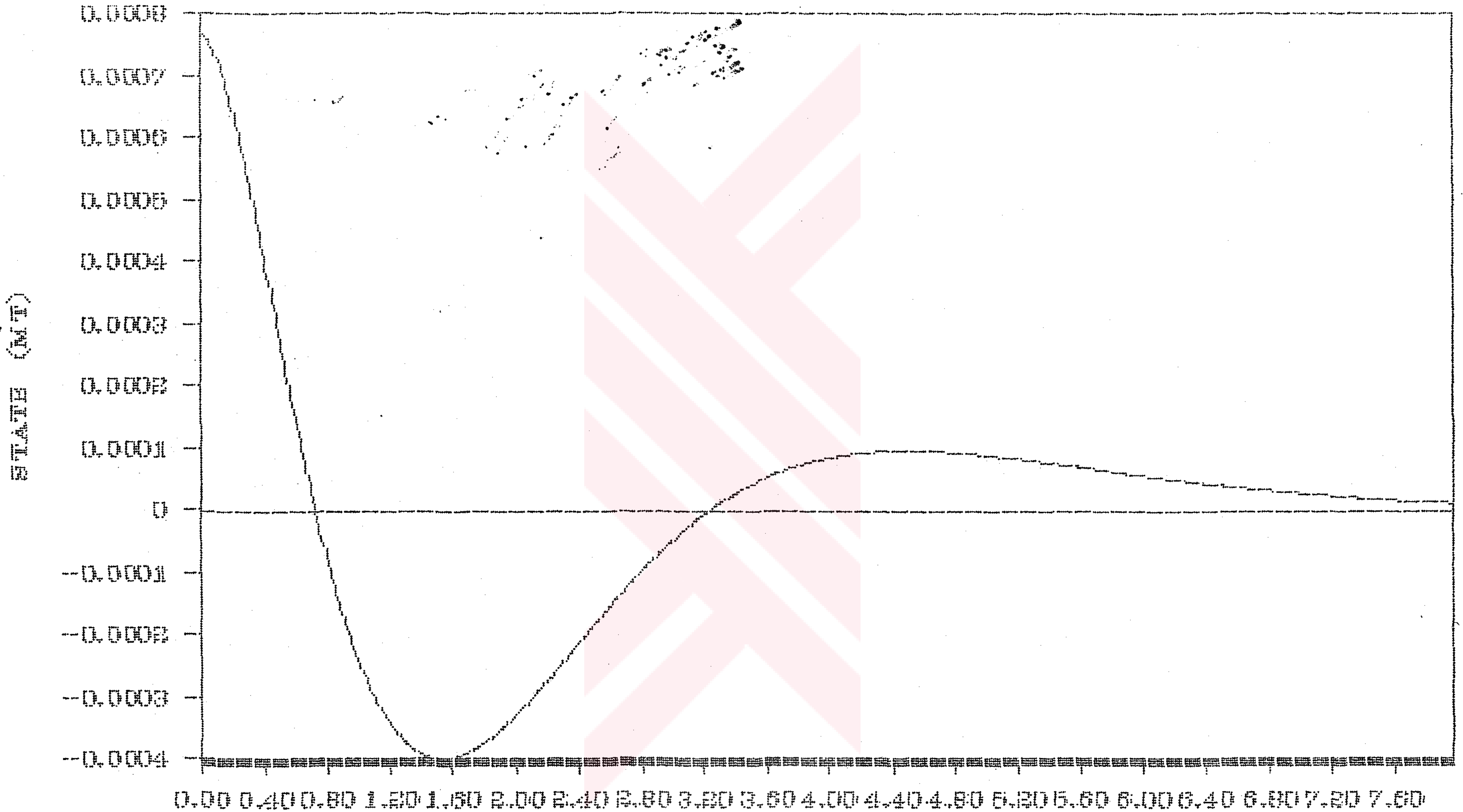


Figure 4.1.10 Closed - loop response of Model III in the x-directon respect to the desired first set of eigenvalues.

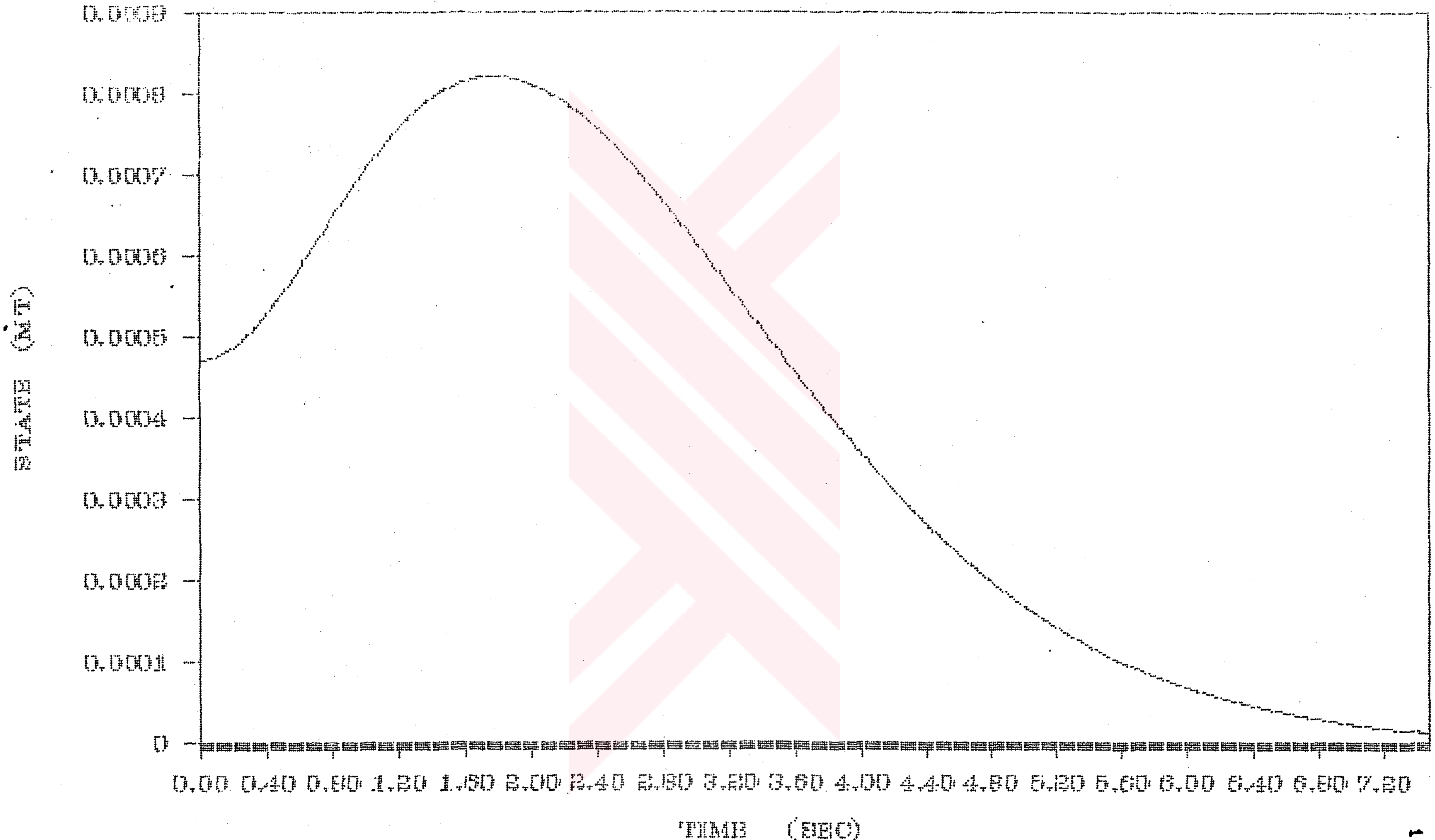


Figure 4.1.11 Closed - loop response of Model III in the y-directon respect to the desired first set of eigenvalues.

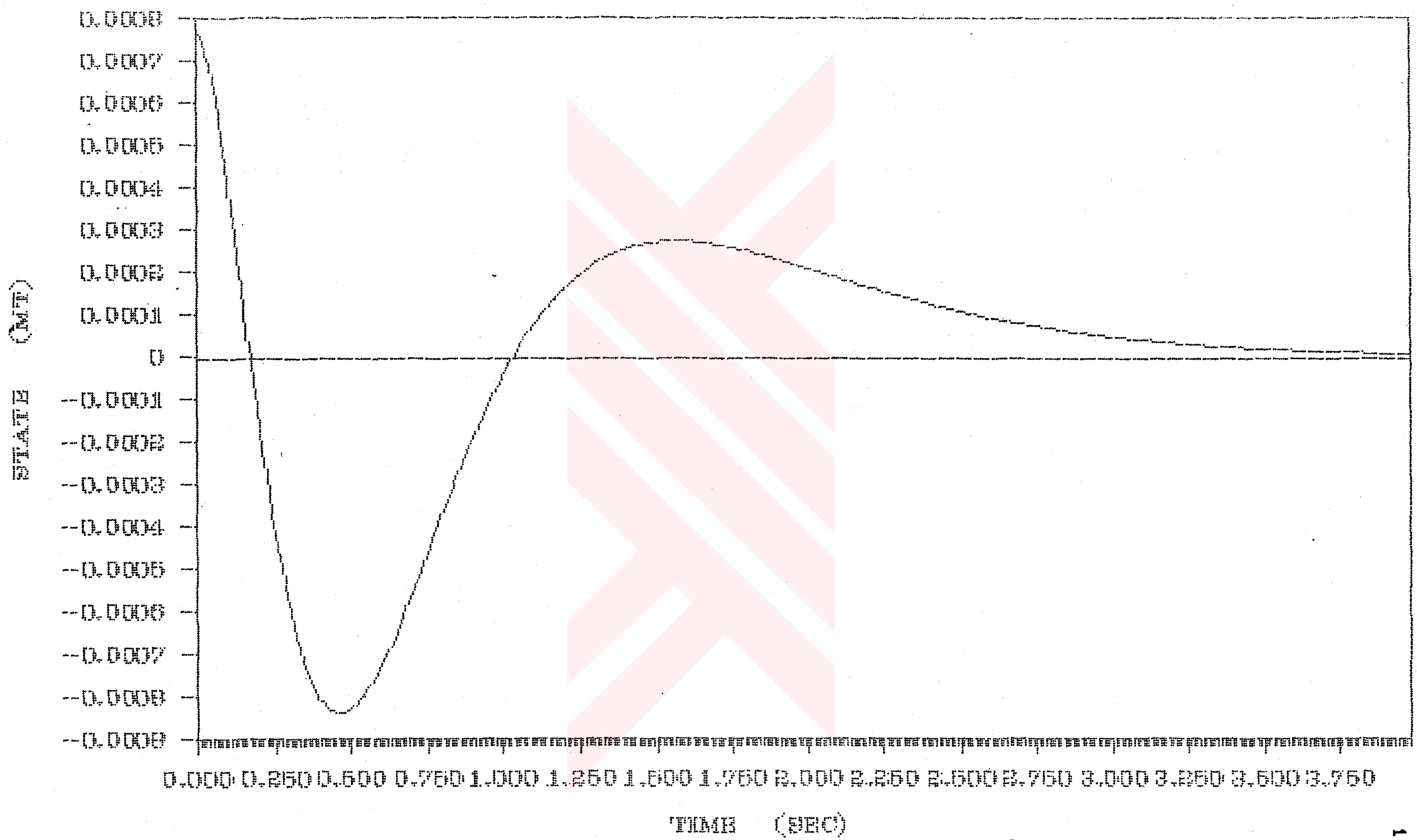


Figure 4.1.12 Closed - loop response of Model III in the x-directon respect to the desired second set of eigenvalues.

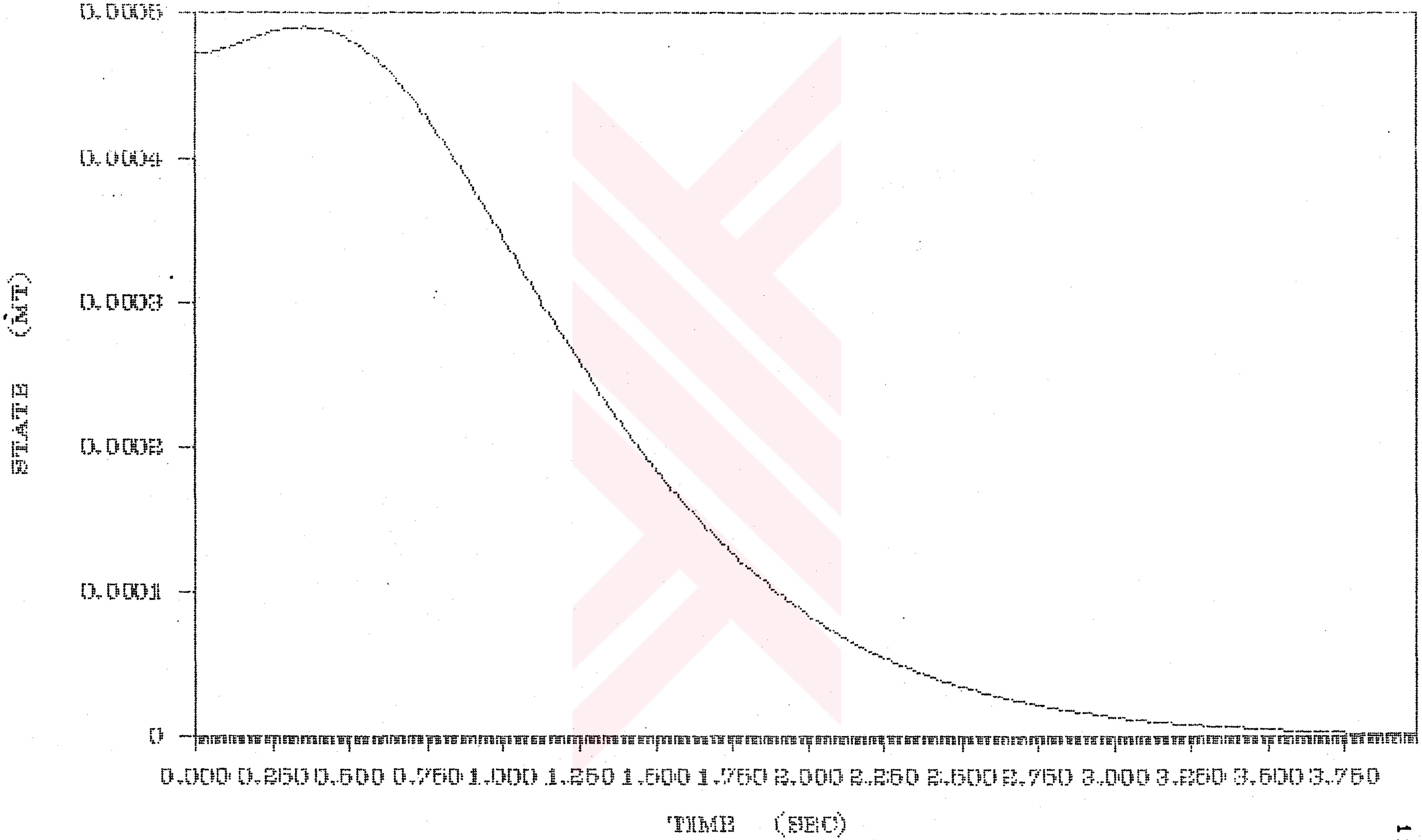
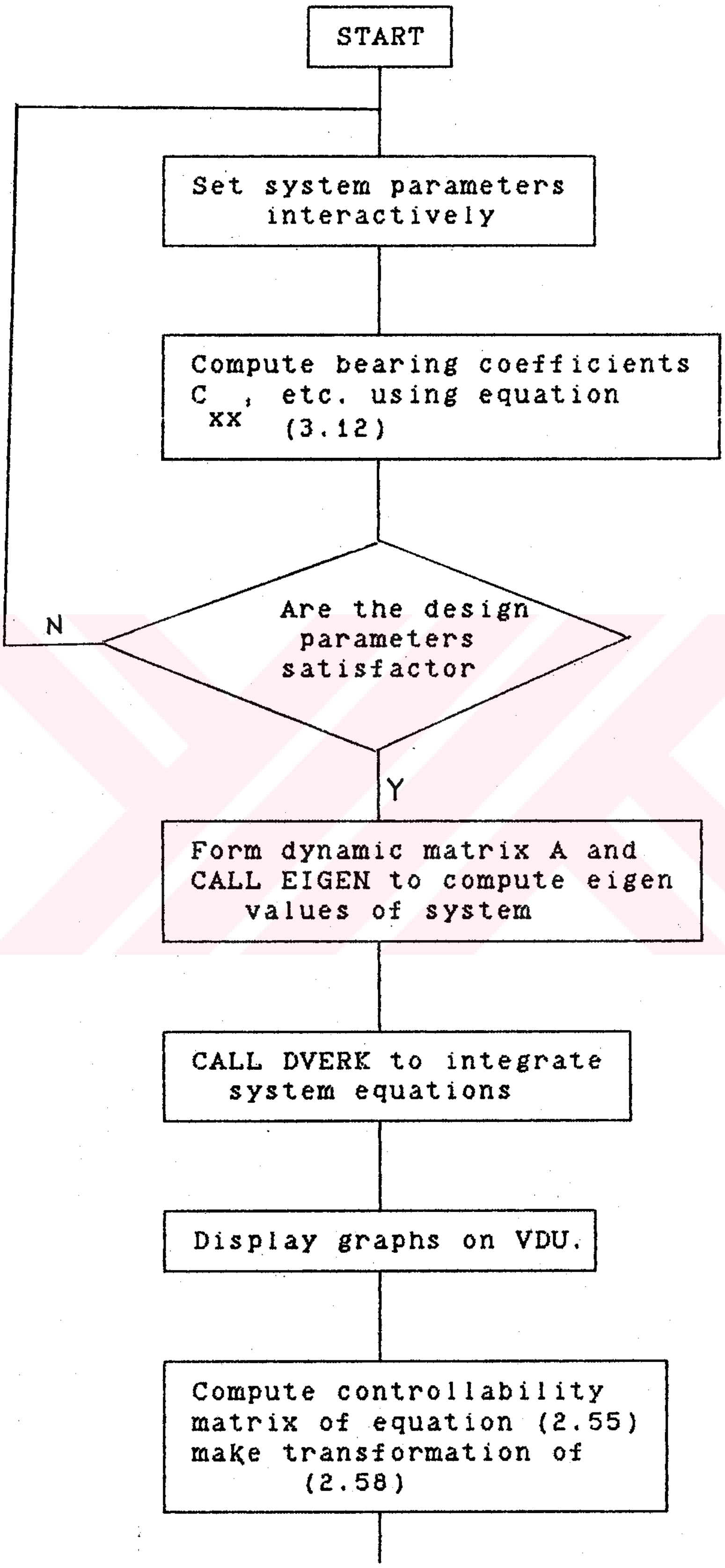


Figure 4.1.13 Closed - loop response of Model III in the y-directon respect to the desired second set of eigenvalues.


START
Set system parameters interactively
Compute bearing coefficients C xx , etc. using equation (3.12)
Are the design parameters satisfactor
N
Y
Form dynamic matrix A and CALL EIGEN to compute eigen values of system
CALL DVERK to integrate system equations
Display graphs on VDU.
Compute controllability matrix of equation (2.55) make transformation of (2.58)

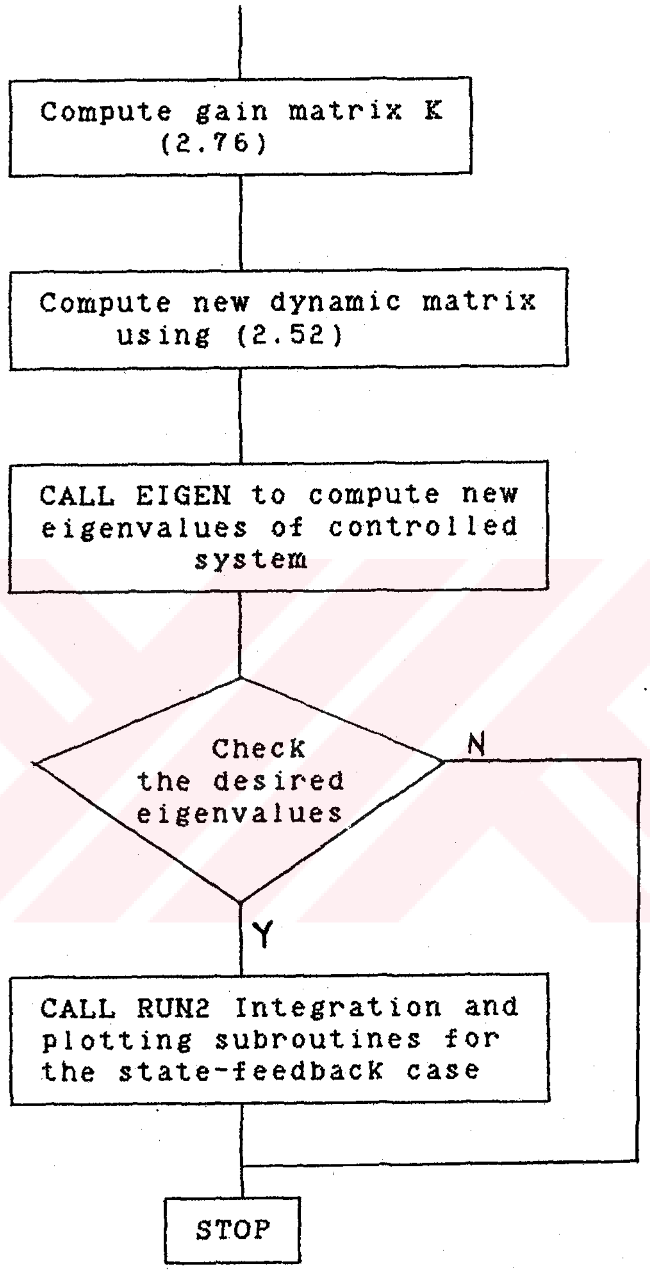


Figure 4.2.1 Flowchart of computer program.

	Model I	Model II	Model III	Model IV
Land length of bearing (m)	0.024	0.009	0.024	0.024
Radius of squeeze bearing (m)	0.0016	0.09	0.0016	0.0016
Clearance (m)	0.000126	0.0004	0.0009	0.000063
Attitude angle in co.ord. system	45	45	45	45
Static eccentricity	0.6	0.6	0.6	0.6
Retainer spring stiffness coef. (N/m)	500000	500000	500000	500000
Mass per land of bearing (kg)	25	25	25	25
Oil viscosity (Ns/m^2)	0.103	0.03	0.103	0.103

Table 4.1.1 Datas of selected models.

MODEL I			
0.00	1.00	0.00	0.00
-20000.00	-85.532	0.00	-45.622
0.00	0.00	0.00	1.00
0.00	-45.622	-20000.00	-423.840
MODEL II			
0.00	1.00	0.00	0.00
-20000.00	-2.309	0.00	-1.232
0.00	0.00	0.00	1.00
0.00	-1.232	-20000.00	-11.607
MODEL III			
0.00	1.00	0.00	0.00
-20000.00	-0.234	0.00	-0.125
0.00	0.00	0.00	1.00
0.00	-0.125	-20000.00	-1.179
MODEL IV			
0.00	1.00	0.00	0.00
-20000.00	-684.256	0.00	-364.981
0.00	0.00	0.00	1.00
0.00	-364.981	-20000.00	-3438.723

Table 4.1.2 Dynamic matrices of selected models

	Eigenvalue no	Re.	Im.
Model I	1	-39.794	135.706
	2	-39.794	-135.706
	3	-383.652	0.0
	4	-52.130	0.0
Model II	1	-1.074	141.417
	2	-1.074	-141.417
	3	-5.883	141.298
	4	-5.883	-141.298
Model III	1	-0.109	141.421
	2	-0.109	-141.421
	3	-0.597	141.420
	4	-0.597	-141.420
Model IV	1	-603.580	0.0
	2	-33.135	0.0
	3	-3480.518	0.0
	4	-5.746	0.0

Table 4.1.3 Eigenvalues of selected models.

REFERENCES

[1] JEFFCOTT, H.H., "The Lateral Vibrations of Loaded Shafts in the Neighborhood of a Whirling Speed... The Effect of Want of Balance", Phil. Mag., Series6, Vol.37, 1919, p.304.

[2] STODOLA, A., "Kritische Wellenstorung Infolge Der Nachgiebkeit Des Oelpolstern Im Lager", Schweizeriche Bauzeitung, 1925, 85, pp 265-266.

[3] COOPER, S., "Preliminary Investigation of Oil-films For The Control Of Vibration", I. Mech. Eng. Lub. and Wear Convention, 1963, paper 28, pp 305-315.

[4] SMITH, D.M., "Journal Bearing in Turbo Machinary", Chapman and Hall, 1969.

[5] MORRISON, D., "The Choice of Optimal Journal Bearing", I. Mech. Eng. Paper No. C161/176, 1976.

[6] SCHWEITZER, G., "Stabilization of Self Exited Rotor Vibrations by an Active Damper, Dynamic of Rotors", Ed. Niordson, F.I., Springer-Verlag, Berlin, pp 472-493, 1976.

[7] SCHWEITZER, G. and LANGE, R., "Characteristics a Magnetic Rotor Bearing For Active Vibration Control", I. Mech. Eng. Conf., Paper no. C239/76, 1976

[8] BLEULER, H. and SCHWEITZER, G., "Dynamics of a Magnetically Suspended Rotor With Decentralized Control", IASTED Applied Dynamics and Control Sym., Copenhagen 1983.

[9] STANWAY, R. and BURROWS, C.R., "Stabilization of a Flexible Rotor by Eigenvalue Assignment", IEE Conf. Contr. and Its Applications, 1981, pp102-105.

[10] BURROWS, C.R., and SAHINKAYA, M.N., "Vibration Control of Multi-mode Rotor-bearing Systems", Proc. R. Soc. Land. Vol.A386, 1983, pp 77-94.

[11] MORE, B.C. "On The Flexibility Offered by State Feedback in Multivariable System Beyond Closed-Loop Eigen Value Assignment", IEEE Trans. Auto. Contr. pp 689-692, October 1976.

[12] LAGENHOP, C.E., "On The Stabilization of The Linear System" Proc. Amer. Math. Soc. vol.15, No.5, pp 735-742

[13] WONHAM, W.M., "On The Pole Assingment in Multi-Input controllable Linear System", IEEE Trans. Auto. Contr. vol AC 12, pp 660-665, Dec.1967

[14] SIMON, J.D. and MITTER, S.K., "A Theory of Modal Control", Inform. Contr. Vol.13, pp 316-353, 1968

[15] BRUNOVSKY, P., "A Classification of Linear Controllable Systems", Kybernetica, Vol.13, pp 173-187, 1970

[16] SERAJI, H.. "Cyclicity of Linear Multivariable System", Int. J. Contr. Vol.21, pp 497-504, 1975

[17] PORTER, B., "Eigen Value Assingment by State Feedback in Multi-Input Linear Systems Using The Generalized Control Canonical Form", Control System Design by Pole-Zero Assignment, Fallside, pp 166-171, 1977

[18] OGATA, K., "Modern Control Engineering"

[19] PORTER, B. and CROSSLEY, R., "Modal Control Theory and Applications", Taylor and Francis Ltd., 1972

[20] ASTROM, K.J. and WITTENMARK, B., "Computer Controlled Systems Theory and Design", Prentice-Hall Inc. 1984

[21] LAYTON, J.M., " Multivariable Control Theory", London Peter Peregrinus Ltd. 1976

[22] DAVISON, E.J., "On Pole Assignment in Linear Systems With Incomplete State Feedback", IEEE Trans. Auto. Contr. Vol.15, pp 348-351, 1970

[23] MUNRO, N., "Further Results On Pole Shifting Using Output Feedback", Control System Design by Pole-Zero Assignment, Fallside, pp 123-135, 1977

[24] ACKERMANN, J., "Der Entwurf Linearer Regelungssyteme im Zustandsraum", Regelungstechnik und Prozess-Datenverarbeitung, Vol.7, pp 297-300, 1972.

[25] HICKIN, J., "Pole Assignment In Single-Input Linear Systems", IEEE. Trans. Auto. Contr., Vol. AC-25, pp 282-284 April.

[26] WONHAM, W.M and JOHNSON, C.D., "A Note On The Transformation To Canonical (Phase-Variable) Form", IEEE Trans. Auto. Contr. Vol. AC-9, pp 312-313, July 1964

[27] FALLSIDE, F. and SERAJI, H., "Direct Design Procedure For Multivariable Feedback Systems", Proc. IEE, Vol. 118, No: 6, June 1971.

[28] TUEL, W.G.Jr., "On The Transformation To (Phase Variable) Canonical Form", IEEE Trans. Auto. Contr. (correspondence) Vol. AC-11, p 607, July 1965

[29] LEUNBERGER, D.G., "Canonical Form For Linear Multivariable System", IEEE Trans. Auto. Contr. Vol. AC-12, pp 290-293, June 1967

[30] SILVERMAN, L.M., "Transformation Of Time Variable System To Canonical (Phase-Variable) Form", IEEE Trans. Auto. Contr. Vol. AC-11, pp 300-303, April 1966

[31] KAILATH, T., "Linear Systems", Prentice-Hall Inc., 1980

[32] MUNRO, N. and VARDULAKIS, A.I., "Computer Aided Design Procedure For Reduced-Order Observers" Proc. IEE, Vol.121, No.4, April 1974

[33] REYNOLDS, O., "On the Theory of Lubrication", Phil. Trans. of the Royal Soc., Series A, 1886, 177, 157.

[34] DEDE, M.M., "An Investigation Into Squeeze-film Dampers", D. Phil. Thesis, University of Sussex, September 1981.

[35] TONNESEN, J., "Experimental Parametric Study of a Squeeze-film Bearing", Jnl of Lubrication Technology, Trans. of the ASME, April 1976, pp 206-213.

[36] TURKAY, O.S., "Adaptive Vibration Control Using A Squeeze-film Bearing", D.Phil. Thesis, University of Sussex, 1982.

[37] HOLMES, R., "The Vibration of A Rigid Shaft on Short Sleeve Bearings", Jnl. of Mech. Eng. Sci., 1960, Vol.2, pp 337-341.

[38] HOLMES, R., "Vibration and Its Control in Rotating System", Proc. IUTAM Symposium on Dynamics of Rotors, Copenhagen, 1974, 156.

[39] STANWAY, R., "Parametric estimation in Linearized Models of Oil-film Bearings", D. Phil. Thesis, August 1978, University of Sussex.