An investigation into the use of multivariable control theory in multi-channel structural testing

This item was submitted to Loughborough University's Institutional Repository by the/an author.

Additional Information:

- A Doctoral Thesis. Submitted in partial fulfilment of the requirements for the award of Doctor of Philosophy of Loughborough University.

Metadata Record: https://dspace.lboro.ac.uk/2134/10545

Publisher: © Serafim Tsavdaras

Please cite the published version.
This item was submitted to Loughborough University as a PhD thesis by the author and is made available in the Institutional Repository (https://dspace.lboro.ac.uk/) under the following Creative Commons Licence conditions.

For the full text of this licence, please go to: http://creativecommons.org/licenses/by-nc-nd/2.5/
AN INVESTIGATION INTO THE USE OF MULTIVARIABLE CONTROL THEORY IN MULTI-CHANNEL STRUCTURAL TESTING

by

Serafim Tsavdaras

A Doctoral Thesis

Submitted in partial fulfilment of the requirements for the award of

PhD of the Loughborough University of Technology

22nd May 1990
ACKNOWLEDGEMENTS

I dedicate this thesis to my parents and my wife Marilou in recognition of their encouragement and understanding.

I would also like to express my gratitude to my supervisor Mr R.W. Pratt for his support and help throughout this project.
CONTENTS

ABSTRACT

1. INTRODUCTION
   1.1

2. STATEMENT OF THE PROBLEM
   2.1 Introduction
   2.2 Types of loads in dynamic structural testing
   2.3 Single actuator tests
      2.3.1 Control specifications for dynamic tests with sinusoidal loads
      2.3.2 Control specifications for dynamic tests with random loading
   2.4 Multi-load dynamic tests
      2.4.1 Interaction
      2.4.2 Multivariable controller design objectives
      2.4.3 Current test practices
      2.4.4 The need for multivariable control

3. THE TEST RIG
   3.1 The test frame
   3.2 The vibrators
   3.3 Choice of beam
   3.4 The vibrator protection circuit
4. MODELLING AND SIMULATION OF THE
EXPERIMENTAL RIG

4.1 Introduction

4.2 Lateral free vibration of a free
cantilever

4.3 Development of the state-space model
of the beam vibrator subsystem

4.4 Introducing damping

4.5 Algorithm for the calculation of the
state-space model

4.6 Evaluation of continuous elastic
body model

5. MULTIVARIABLE FEEDBACK DESIGN METHODS

5.1 Introduction

5.2 The characteristic locus method

5.2.1 Closed-loop stability assessment
using characteristic gains

5.2.2 The commutative controller

5.2.3 Interaction and system
eigenstructure

5.2.4 Physical interpretation of systems
eigenstructure

5.2.5 The design method

5.3 The inverse and the direct Nyquist
array method

5.3.1 Introduction

5.3.2 Diagonal dominance and graphical
criteria for stability

5.3.3 Achieving diagonal dominance

5.4 Robust stability and performance

5.4.1 Robust stability

5.4.2 Relationships between open-loop
and closed-loop principal gains
6. THE CONTROLLER DESIGN

6.1 Introduction 6.1
6.2 Nyquist array design 6.3
6.3 The characteristic locus design 6.11
6.4 Evaluation of the Nyquist array and characteristic locus designs 6.17

7. SOFTWARE DEVELOPMENT 7.1

7.1 Introduction 7.1
7.2 Description of CADMVS2 7.2
    7.2.1 C.A.D. options 7.3
    7.2.2 User interface 7.4
7.3 New MATLAB functions 7.4

8. CONCLUSIONS 8.1

9. FUTURE WORK 9.1

REFERENCES

APPENDIX 1

APPENDIX 2
ABSTRACT

The aim of this project is to investigate the potential use of multivariable control theory in multi-channel dynamic structural testing. The ideal behaviour of a control system for dynamic testing is analysed and this is used to provide the specifications for control schemes for both sinusoidally derived and random loadings. The need for an integrated multivariable control system approach is shown.

An experimental test rig is used to test the applicability of multivariable control methods to dynamic testing. The rig consists of a cantilever beam which can be excited into vibration by two electrodynamic shakers. The displacement on two locations on the beam is measured by Linear Variable Displacement Transducers (LVDTs). The rig is configured to provide the worst possible cross-coupling between different input-output pairs.

Analytical modelling of the test is accomplished by regarding the cantilever beam as continuous elastic body. The order of the models, which can be either in state-space or in transfer function matrix form, is a function of the frequency range over which accurate modelling is required. The output variable can be displacement, velocity or force.

The analytical modelling led to programs for the simulation of the beam for any given input time series and for a varying number of vibrators and output measurement transducers. The locations of both shakers and output sensors can be varied at will by a procedure which is transparent to the user.

The multivariable control methods used for controller design are the characteristic locus and the Nyquist array method with recent extensions. Emphasis was given during the design process to producing robust performance and
stability. Both the characteristic locus and the Nyquist array designs were assessed using singular value decomposition analysis and were found to perform satisfactorily. The merits of both designs are analysed and compared.

In the initial stages of the project a considerable amount of time was devoted to developing a C.A.D. package for multivariable control. The features of this package are described. Also software development based on the MATLAB package is discussed.
INTRODUCTION

One of the main design objectives of any mechanical structure is to be able to withstand the variety of the loading forces it will be subjected to when introduced into its service environment. The complexity of the structure in combination with the varying nature of the applied forces makes it impractical, if not impossible to determine the necessary dynamic strength of the structure by analytical means alone. Laboratory dynamic structural tests are used to overcome this problem.

Dynamic structural testing is the application of a programmed sequence of loads to a complex mechanical structure in order to assess its response to real life stresses. Such tests could take the form of accelerated life tests to determine the fatigue resistance of the structure. A typical example is an aircraft wing where the test programme simulates the change in load patterns due to aerodynamic loads, the fuel tanks being emptied and refilled, the weight of the aircraft taken via the undercarriage on landing, wind gusts and a variety of stresses due to flight manoeuvres. This is achieved by applying known excitations to the structure in order to recreate known service conditions of a controlled variable such as displacement, velocity or acceleration while stress and strain distributions are monitored for subsequent analysis and evaluation. The response of the controlled variable will be partly due to the frequency composition of the excitation forces and partly due to the dynamics of the structure. Dynamic structural test however, are usually designed with little regard to the dynamics of the test object [1].

Single shaker tests are still widespread. Even when
multi-shaker tests are employed, individual servo controllers are often used alone, with loops being closed sequentially. Naturally this method does not take into account the interaction between different shaker inputs and outputs due to the structure's dynamics. Even when optimal inputs to the shakers, in multi-shaker tests, are calculated in order to maximise a displacement severity measure, the suggested test will run open-loop [2]. In the most sophisticated method of multi-channel dynamic structural testing, iterative numerical methods are used to determine the excitations by deconvolution of the measured outputs, thus producing the required response[3]. Input-output cross talk is in this case taken into account, but the calculation of the control inputs is done off-line and again the test runs open-loop.

The aim of this project is to introduce an alternative control strategy for dynamic structural testing based on frequency domain control theory of multivariable systems. In this approach the dynamics of the test equipment, shakers and transducers, together with the dynamics of the structure under test, form an integrated multi-input multi-output control system. The structural interaction and the decoupling of the various input-output pairs is central to the design process. The desired system performance is analysed and control specification are laid out for both sinusoidally based and random inputs. The control is based on feedback of directly measured control outputs.

The importance of the proposed control strategy is that for the first time a systematic approach is suggested to provide robust system performance and the required accuracy. In this way a more realistic simulation of the test object's service life may be achieved during structural testing. This is accomplished by designing controllers to reduce interaction and to provide increased accuracy and bandwidth.

Mathematical modelling of structural test systems can be
a very complex task. Additionally the frequent changes of test object in many tests of this type means that this difficult exercise could be frequently repeated. The solution most frequently adopted is to disregard the dynamics of the structure completely and try to achieve the required accuracy using single-loop servo system's alone. This inevitably degrades the performance of the system. Multivariable frequency domain methods can be used instead since no analytical modelling is required and the necessary frequency response data can be easily obtained experimentally.

In order to investigate the use of multivariable control systems in dynamic structural tests a special test rig was built in a previous project. This consists of a cantilever beam excited into vibration by two electrodynamic vibrators mounted one at the free end and the other at the mid-point of the beam. The same experimental set up has been used by other workers in the field in order to assess the effects of interaction [4,5]. However this configuration restricts the size of the test system to two inputs and two outputs. Further constraints are imposed by the fixed positions of the vibrators relative to the beam. In order to facilitate further investigation of the system, analytical modelling has been undertaken which made possible the digital simulation of the system with a varying number of vibrators and displacement transducers as well as different beam parameters [6].

Multivariable controllers are designed for the test rig using both the characteristic locus and the Nyquist array methods. Emphasis is placed on achieving robust performance and stability which are assessed by the use of singular value analysis. Although the primary data comes from experimentally obtained frequency responses, the results presented in this thesis are obtained by simulation.

The results were published as obtained in the various stages of the project to both international control...
conferences [7,8,9] as well as being made known to users of dynamical test equipment [10]. This provided useful feedback which favourably affected the progress of the project.
CHAPTER 2

STATEMENT OF THE PROBLEM

2.1 INTRODUCTION

The purpose of dynamic structural tests is to determine the dynamic strength of the tested structure as well as its fatigue resistance.

Fatigue is the process of cumulative damage in a mechanical structure which is caused by repeated fluctuating loads. Fatigue damage is a localised phenomenon that occurs only in regions that deform plastically. Accumulated damage in these regions initiates and subsequently propagates one or more cracks. The process can and in many cases does cause the fracture of the component.

The purpose of fatigue tests is to provide the necessary data that will make possible the estimation of the fatigue life of the component or structure under test. I.e. the estimation of the time during which the probability that a fatigue crack will propagate beyond a certain level is sufficiently low. From the test data "cycles" or load reversals are counted, and subsequently are used to extrapolate the fatigue life using linear or non-linear rules.

The problems of designing a dynamic structural test can be summarised in the following two requirements:

1. Decide the type and locations of the loads that should be applied to the structure during the test and
2. Design such control schemes that will apply the previously determined load to the structure.
The solution to requirement 1, will depend on how accurately service conditions should be reproduced, what method is going to be used to evaluate the test results and in the case of fatigue tests estimate the total fatigue life, and the laboratory equipment available to reproduce these loads. In general, laboratory loads will be chosen to approximate field collected data to a varying degree of accuracy. Until recently all dynamic testing was done using sinusoidal loads with varying or constant amplitude and frequency. Various methods have been developed over the years for calculating fatigue life using the data generated by such tests.

Although the use of sinusoidal loads is still widespread today, very rarely they can approximate accurately actual service loads. This is especially true for land, sea, air, ground and space vehicles. It is now generally agreed that the sources of loads for such vehicles, be it road shocks, air turbulence or propellant combustion, can be properly represented by a continuous stationary random process. For such applications wherein minimum-weight, non-rigid body mechanics and extreme safety and reliability are paramount, the power-spectral-density (PSD) method of load analysis is being widely used. In tests that employ this method, the PSDs of the excitations which are applied to the structure in the laboratory, must match the PSDs of field measured loads.

A further improvement in the accuracy of the laboratory simulation of service conditions can be achieved by using multiple loads simultaneously. This however imposes additional and often complicated control problems as it will be shown later in this chapter, after the single actuator control problem is discussed for both sinusoidal and random loadings.
2.2 TYPES OF LOADS IN DYNAMIC STRUCTURAL TESTS.

The different types of loads that are used in dynamic tests fall in one of two categories: sinusoidally derived loads or random loads described by stochastic processes. Both categories are discussed and their merits analysed in what follows in this section.

The simplest form of sinusoidal loading is the one whose both amplitude and frequency are constant. Such tests were first used, before the widespread use of servo-hydraulic actuators in dynamic tests, since they are relatively simple to generate. This is why the bulk of the available data is still constant-amplitude, constant-frequency sine load generated. Also the simple form of this type of loading makes the prediction of fatigue life relatively easy, even if the accuracy of this prediction is often questioned. Techniques for fatigue life prediction however exist today based on tests with more representative of real life, and inevitably more complex, loadings. This, combined with the availability and use of more and more sophisticated control schemes to generate loadings will eventually reduce the dependence on data generated by such tests. Constant-amplitude single-frequency test are still used today, but mainly for historical reasons.

It is known for many years now that even if the frequency distribution of service loads is ignored in the fatigue test, the amplitude sequencing of the loading, that is high-low or low-high, will affect the fatigue life of the test object. Therefore the next more advanced state in sine wave loadings was to come, by varying the amplitude of the signal alone. The amplitude of this type of loading can be varied in a pre-programmed way in any single block of loads to achieve various sequencing effects. In this type of test which is called a program or block test, a sequence of load cycles with
different amplitudes is repeated periodically. In any one period, the cycles with the same amplitude are grouped together. The amplitude can be arranged in each period to vary from low to high, or high to low, or low to high and then low again, or even in a pseudo-random manner as is shown in figure 2.1. Usually the amplitude is varied in such a way so that the test load spectrum matches the load spectrum of a representative record of service loads. The term "load spectrum" in the fatigue literature is not associated with the frequency composition of the applied load as the word spectrum would imply; it is generally taken to mean the plot of the number of occurrences or exceedences of any given load level per unit time or distance. This curve is arrived at by summing the number of events in a load-time trace by any of the common counting methods: peak counting, level crossing counting, or rainflow.

Approximating the service load spectrum, by the load spectrum of a block test generates more realistic fatigue tests. The frequency composition of the service loads are however ignored and therefore severe limitations are imposed to the accuracy of the simulation of real life conditions in the laboratory by this method. To overcome this while still using sinusoidal loads, a frequency range is swept by the load signal so as to excite all the significant modes of a multi-degree of freedom test object. Again there are two types of tests used. One in which the amplitude of the load is kept constant while the frequency is varied and a second type in which both the amplitude and the frequency of the exciting signal are varied. This last type of sinusoidal load tests is usually arranged to provide a program or block frequency sweep test. Using this test, service loads can be simulated in the laboratory with an increasing degree of accuracy as the frequency increment is decreased, by approximating their frequency spectrum.

The fact, however, that is very rare for actual service
loads to be of a sinusoidal nature will ultimately put limits to how realistic sinusoidal tests of any type can be. In the cases where conservative life predictions are not acceptable for technical or financial reasons and more accurate tests are called for, accurate simulation of service loads seems to rule out the use of sinusoidal tests.

Service loads are very often random in nature and as such they are described by probabilistic or stochastic processes. Such a process is completely described in the frequency domain by its power spectral density function (PSD). A dynamic test therefore that uses a load whose PSD matches the PSD of the actual service loads will accurately reproduce, in a probabilistic way, the measured service conditions in the laboratory. Additionally if the random process is stationary (its statistical properties are invariant with time) and ergodic (one record completely describes statistically the process) reproduction of one record of time history will achieve the same result.

Having described all the different types of loads used in dynamic structural tests, the problem of reproducing them accurately and safely in the laboratory will be examined next. First the control problem for the case of single actuator tests will be analysed and the more complex case of multi-load tests will follow.

2.3 SINGLE ACTUATOR TESTS.

Servo-hydraulic or electromechanical actuators are employed in most dynamic tests to apply the loads to the structure. Both types are driven by voltage signals \( u(t) \), to produce the applied loads \( y(t) \). The functions \( u(t) \) and \( y(t) \) will be from now on referred to as input and output signals respectively.
If they are expressed in terms of Laplace transforms, their relationship is described by the equation:

\[
y(s) = g(s) . u(s)
\]  

(2.1)

and in the frequency domain.

\[
y(j\omega) = g(j\omega) . u(j\omega)
\]  

(2.2)

Where \( g(s) \) is a transfer function that describes the dynamics of the structure under test as well as the dynamics of the actuator and the test equipment. Linear behaviour of \( g(s) \) has been assumed above and throughout this thesis.

The obvious solution to the problem should be to calculate the input function \( u(s) \), that will produce the desirable output \( y(s) \). However for this solution to be viable, accurate knowledge of \( g(s) \) is required. Even if \( g(s) \) is known accurately, variations in the structure dynamics affected by the test and non-linearities present in the structure will reduce the accuracy of the system as the test progresses. Also any electrical noise present in the input signal will be transmitted to the output thereby further reducing accuracy of the test.

The alternative approach is to use a self correcting scheme whereby the system is reacting to the error between a reference signal, \( r(s) \), and the actual system output. Such a closed-loop system is shown in fig. 1. In this case the input reference signal is equal to the desired output, and the transfer function \( h(s) \), between \( r(s) \) and \( y(s) \), is manipulated to produce the required output.

-2.6 -
The relationship between \( r(s) \) and \( y(s) \) is now given by

\[
y(s) = h(s) \cdot r(s) = \frac{k(s) \cdot g(s)}{1 + k(s) \cdot g(s)} \cdot r(s) \tag{2.3}
\]

where \( k(s) \) is a transfer function that describes the controller which can be dynamic or a pure gain factor, and is used to manipulate the closed-loop transfer function \( h(s) \). The form of \( k(s) \) can vary from a simple gain factor to a three term PID (proportional plus integral plus derivative) or even more specialised forms of controllers. Powerful tools have been in existence for many years to aid the control engineer in designing \( k(s) \). These tools are collectively called "classical-control" techniques, and include well known methods such as the Nyquist diagrams, Bode plot and root locus. It is well documented that classical closed-loop control schemes can provide insensitivity to system parameter variations, disturbance rejection, good stability margins and accurate performance. It is for this reason that they are almost exclusively used in single actuator tests. Before any design commences however, the performance specifications for the control system must be specified. In what follows in this section the required behaviour of \( h(s) \) will be examined, to produce either sinusoidally derived or random loads to be applied to the structure under test.

### 2.3.1 Control specifications for dynamic tests with sinusoidal loads.

The simplest form of load to be reproduced by an actuator controlled by closed-loop schemes is sinusoidal loads with constant frequency.
This is easy to see if the input-output relationship is examined in the frequency domain:

\[ y(j\omega) = h(j\omega) \cdot r(j\omega) = \frac{k(j\omega) \cdot g(j\omega)}{1 + k(j\omega) \cdot g(j\omega)} \cdot r(j\omega) \quad (2.4) \]

For constant frequency, \((\omega = \omega_0)\), \(g(j\omega)\) and \(k(j\omega)\) are constant, assuming a linear time invariant system. The transfer function \(h(j\omega)\), is in this case, effectively a scaling or gain factor, introducing perhaps some phase shift as well between the reference signal and the output, which however has no significant effect in system performance. In other words if the system input is a sine wave, the output will also be a sine wave with the same frequency. The amplitude of the output signal will be that of the input signal scaled by a factor equal to the amplitude of the transfer function \(h(j\omega_0)\), where \(\omega_0\) is the frequency of the input signal. The controller, \(k(j\omega)\), need only be chosen to give to the system adequate stability margins, noise rejection, and insensitivity to parameter variations for the specific frequency point \(\omega = \omega_0\). This is easily accomplished, since for \(\omega = \omega_0\), all calculations are done with constant values. The reference input to the system for such a load will be a signal with the same mathematical description as the desired load, appropriately scaled to offset the effect of, the now constant, transfer function matrix \(h(j\omega)\).

The problem is more complicated when sweep sinusoidal loads of constant or variable amplitude are to be produced.
The added complexity results from the fact that \( \omega \) now varies over a range of frequencies:

\[
\omega_1 \leq \omega \leq \omega_2
\]  

(2.5)

and the value of \( h(j\omega) \) will also vary over the same frequency range. Therefore in addition to the above consideration (stability margins, noise rejection and insensitivity to parameter variations), the controller \( k(j\omega) \) must be designed to provide a constant gain over the frequency range of interest. In the simplest cases \( k(s) \) can be a constant gain factor whose value will be high enough to give to the system the required gain. Theoretically infinitely high gains will produce an output which is equal to the reference input since

\[
\lim_{k \to \infty} h(j\omega) = \lim_{k \to \infty} \frac{kg(j\omega)}{1 + kg(j\omega)} = 1.
\]

(2.6)

and

\[
\lim_{k \to \infty} y(j\omega) = r(j\omega)
\]

However stability or other practical considerations can put limits to how high a gain can be used. In this cases controllers with dynamic terms are designed to shape the frequency response and provide constant overall gain over the frequency range of interest.
2.3.2 Control specifications for dynamic tests with random loading

Consider now the case where the system input is a random function \( x(t) \) with a PSD function \( \Phi_{xx}(\omega) \); the system output will also be a random function, \( y(t) \), with a PSD function denoted by \( \Phi_{yy}(\omega) \). The PSD functions of input and output signals will have the following relationship:

\[
\Phi_{yy}(\omega) = \frac{1}{\gamma_{xy}^2(\omega)} |h(j\omega)|^2 \Phi_{xx}(\omega)
\]

(2.7)

where \( \gamma_{xy}(\omega) \) is the coherence function.

The coherence function however for linear systems is equal to 1 and therefore

\[
\Phi_{yy}(\omega) = |h(j\omega)|^2 \Phi_{xx}(\omega)
\]

(2.8)

for linear systems.

When fatigue tests are done with random loading, the PSD of the load should match the PSD of the random function that describes the actual service loads. In control terms, for a closed-loop system which would produce these loads, the requirement is that the PSD of the system output should match the PSD of the reference input. Equation [8] shows that this requirement is translated into constant overall gain for the system, over the useful frequency range. Again this is accomplished by designing the controller \( k(s) \) to shape the systems frequency response, to provide constant gain, or "flat
frequency response", over the system bandwidth. The resulted constant gain should ideally be equal to 1, but if that is not possible the reference input \( r(s) \) can easily be scaled by a constant factor to produce the desired output.

It can be seen from the above discussion that although a diverse variety of loads is used in fatigue tests, the control system requirements do not differ for any particular type of loads. The same set of objectives must be achieved when designing controllers for loads as diverse as simple sinusoidal and random signals. The reason for that is that all these signals being sinusoids or random described by PSD functions are analysed predominantly in terms of their frequency characteristics. The system performance can therefore be directly related to its frequency response and performance objectives can easily be stated in terms of the "shape" of the frequency response curves. These objectives can be summarised into the following:

Design the controller \( k(s) \) to achieve:

(a) Good stability margins.
(b) Noise and rejection.
(c) Insensitivity to system parameter variations
(d) Disturbance rejection.
(e) Increased bandwidth
(f) Constant amplitude (flat frequency response) over the useful frequency

The closed-loop system amplitude in requirement (f) should ideally be equal to 1, but failing that scaling of the reference input by a constant factor can achieve the required output over the frequency range that the system amplitude remains constant. A controller designed to the above specification can be used virtually unchanged for all usual fatigue loads, and test objects with similar dynamic characteristics.
2.4 MULTI LOAD DYNAMIC TESTS.

In the previous section ways of improving the accuracy of the simulation of service conditions by careful choice of the type of load applied to the structure was discussed. The discussion however was confined to single actuator tests. Although this type of test is widely used, it is clear that in many situations cannot provide realistic simulation of service condition. An example of this is the case of road vehicles subjected to fatigue tests. Road simulation cannot be possibly achieved by the use of a single actuator. At least four actuator will be needed to provide the lateral vibrations to the four wheels or hubs. In many cases additional vibrators will be used to simulate braking action and sideways movement of the vehicle. Another example where the need of multi-load structural testing arises is when testing aeronautical structures. A significant part of real life loads on such a structure will be due to air turbulence, and thus they will be spatially distributed over the surface of the structure. Even loads due to engine vibration will be applied to more than one points on the structure, for aircrafts with multiple engines. Multiple loading is however a lot more difficult to simulate accurately in the laboratory. In this section the reasons for the added complexity which is usually present in multi-load tests will be examined. Also the methods in use today will be critically analysed, and ways of dealing with the problems associated with these method will be suggested.

2.4.1 Interaction

The problem associated with the control system design for multi-load structural testing is the result of the mechanical coupling that exist between different inputs and outputs. As an example consider a structural test situation of an aircraft
wing. Field data has been collected in the form of displacement at the tip of the wing and at another point towards the middle of the wing. The immediate task is to reproduce the field data in the laboratory. Naturally two shakers will have to be used each mounted on the corresponding points on the wing where the field data was collected from. Each shaker will be driven by input functions $u_1(t)$ and $u_2(t)$ respectively. But any movement of the vibrator mounted at the middle of the wing will cause displacement not only where it is mounted, but at every point along the wing including the tip of it. The same of course is true for the other vibrator. In other words each output is not due to its corresponding input alone, but other inputs affect it as well. The input-output relationship described by equation (2.1) must be changed to reflect the coupling between different input and outputs, found in multi-load testing as follows:

$$y(s) = G(s).u(s) \tag{2.9}$$

Where $y(s)$ and $u(s)$ are now vectors representing the outputs and inputs respectively. $G(s)$ is the matrix of transfer functions, called the transfer function matrix. The coupling of the different inputs and outputs is represented in the transfer function matrix by the off-diagonal terms. In the above example of the aircraft wing the input and output vectors are,

$$u(s) = \begin{bmatrix} u_1(s) \\ u_2(s) \end{bmatrix} \quad \text{and} \quad y(s) = \begin{bmatrix} y_1(s) \\ y_2(s) \end{bmatrix}$$
while the dynamics of the structure are represented by the transfer function matrix, \[ G(s) = \begin{bmatrix} g_{11}(s) & g_{12}(s) \\ g_{12}(s) & g_{22}(s) \end{bmatrix} \]
equation (2.9) then becomes \[
\begin{bmatrix} y_1(s) \\ y_2(s) \end{bmatrix} = \begin{bmatrix} g_{11}(s) & g_{12}(s) \\ g_{21}(s) & g_{22}(s) \end{bmatrix} \begin{bmatrix} u_1(s) \\ u_2(s) \end{bmatrix}
\]
with
\[ y_1(s) = g_{11}(s) . u_1(s) + g_{12}(s) . u_2(s) \]
and
\[ y_2(s) = g_{21}(s) . u_1(s) + g_{22}(s) . u_2(s) \]
The contribution of the input \( u_2(s) \) to the output \( y_1(s) \) and the contribution of the input \( u_1(s) \) to the output \( y_2(s) \), can clearly be seen in the last two equations. For the general case of system with \( m \) inputs and \( 1 \) outputs, sometimes called "m by 1 system", these last equations become:
\[
y_i(s) = g_{i1}(s) . u_1(s) + \sum_{j=1 \\
\text{j \neq i}}^{m} g_{ij}(s) . u_j(s), \quad i = 1, \ldots, 1
\]
The second term is the right hand side of equation (2.10) represents the contribution of all the other inputs beside the ith, to the ith output. This is clearly an undesirable feature of control systems for multi-load tests. Such multi-input, multi-output control systems are called multivariable control systems, and the coupling between different inputs and outputs that exists in such systems is termed interaction. It is apparent that interaction will affect the accuracy of any control system. Accurate control will therefore be difficult in the presence of considerable interaction.

The other consequence of equation (2.10) is that for the ith output to be stable, stability of the corresponding transfer function $g_{ii}(s)$ is not enough. The rest of the transfer functions in the same row in $G(s)$ must also be stable. This means, apart from the obvious requirement that all transfer function elements of $G(s)$ must be stable, that calculation of stability margins in each loop is difficult to achieve. Indeed as it will be seen later although Nyquist type criteria have been extended to cater for the case of multivariable systems, stability margins cannot be quantified in a manner analogous to the single-input, single-output case.

In the light of this difference between control systems for single and multiple actuators the controller design objectives for the multivariable case will be described next.

2.4.2 Multivariable controller design objectives

Since all the added complexity associated with multivariable systems is the result of interaction, inevitably the conclusion is reached that the control scheme must be designed to alleviate interaction. If no interaction is present in the system the off diagonal terms of the transfer
function matrix will be zero. Equation (2.10) in this instance will become

\[ y_i(s) = g_{ii}(s)u_i(s), \quad i = 1, 2, \ldots, m \] (2.11)

The multivariable problem would therefore be reduced to a collection of single-input single-output control systems. Each of these systems could then be designed using the well known classical control techniques mentioned in section 2.3. The controller that would achieve an absolutely non-interacting system would be equal to the inverse of the transfer function matrix \( G(s) \). The transfer function matrix however is a matrix of rational function. To invert the transfer function matrix of a high order system, assuming it is not singular, can be a difficult job. In any case this controller would be very complex and therefore difficult to implement and could even be unstable. It is for this reason that the controller is usually designed to reduced the interaction to such a level that its effects will not be significant and can therefore be ignored.

Once interaction has been reduced to acceptable levels, the same design objective for accuracy must be pursued. It has been shown in section 2.3 that for single-input single-output systems good accuracy is provided if the magnitude curve of the frequency response is "flat" over the frequency range of interest. This result will now be extended to the case of multivariable systems with the proviso that interaction has been effectively eliminated.

Consider a multivariable system, with no interaction, which is described by a diagonal transfer function matrix \( G(s) \). The frequency domain description of the system outputs \( y(j\omega) \), in
response to inputs \( u(j\omega) \) will be:

\[ y(j\omega) = G(j\omega)u(j\omega) \]

If \( u(j\omega) \) is a deterministic signal then

\[
\begin{bmatrix}
    y_1(j\omega) \\
    y_2(j\omega) \\
    \vdots \\
    y_m(j\omega)
\end{bmatrix} =
\begin{bmatrix}
    g_{11}(j\omega) & 0 & \ldots & 0 \\
    0 & g_{22}(j\omega) & \ldots & 0 \\
    \vdots & \vdots & \ddots & \vdots \\
    0 & 0 & \ldots & g_{mm}(j\omega)
\end{bmatrix}
\begin{bmatrix}
    u_1(j\omega) \\
    u_2(j\omega) \\
    \vdots \\
    u_m(j\omega)
\end{bmatrix}
\]

(2.12)

In the ideal case therefore that the system is non-interacting, it can be split into \( m \) single-input single-output subsystems, such that:

\[ y_i(j\omega) = g_{ii}(j\omega)u_i(j\omega) \]

If \( y_1(j\omega) \) is to be any of the sinusoidal loads that are used in fatigue tests, it is clear that the "shape" of the frequency response of the transfer function \( g_{11}(j\omega) \) that will provide good accuracy is as described in section 2.3. Namely a flat gain curve in the frequency response.

If the input to the system is stochastic, and is described by its spectral density function, the analysis is more complicated. The reason for the added complexity is that the PSD function of a vector time function, is a matrix. This matrix will have as diagonal elements the autospectral density.
functions of the elements of the vector time function. Its off diagonal elements will be the cross-spectral density functions of different elements of the vector time function. The PSD of the input vector \( u(t) \) will therefore be:

\[
\Phi_{uu}(\omega) = \begin{bmatrix}
\phi_{u_{11}}(\omega) & \phi_{u_{12}}(\omega) & \ldots & \phi_{u_{1m}}(\omega) \\
\phi_{u_{21}}(\omega) & \phi_{u_{22}}(\omega) & \ldots & \phi_{u_{2m}}(\omega) \\
& \ddots & \ddots & \ddots \\
\phi_{u_{m1}}(\omega) & \phi_{u_{m2}}(\omega) & \ldots & \phi_{u_{mm}}(\omega)
\end{bmatrix} \tag{2.14}
\]

In (2.14) \( \phi_{u_{11}}(\omega) \) is the autospectral density function of \( u_1(t) \) and \( \phi_{u_{ij}}(\omega) \) is the cross-spectral density function of \( u_i(t) \) and \( u_j(t) \).

It is shown below that even in this case the control specification for accuracy remain the same.

By definition:

\[
\Phi_{yy}(\omega) = \mathbb{E} [y(j\omega) \cdot y^*(j\omega)] \\
= \mathbb{E} [G(j\omega) \cdot u(j\omega) \cdot \{G(j\omega) \cdot u(j\omega)\}^*] \tag{2.15}
\]

Assuming that \( G(j\omega) \) is constant at each frequency point \( \omega \).

\[
\Phi_{yy}(\omega) = G(j\omega) \cdot \mathbb{E} [u(j\omega) \cdot u^*(j\omega)] \cdot G^*(j\omega) \tag{2.16}
\]

or

\[
\Phi_{yy}(\omega) = G(j\omega) \cdot \Phi_{uu}(\omega) \cdot G^*(j\omega) \tag{2.17}
\]
Assume further that $G(j\omega)$ is non-interacting (i.e. it is diagonal). If for brevity $g_{ij}$ is used instead of $g_{ij}(j\omega)$ and $\phi_{ij}$ instead of $\phi_{ij}(\omega)$, equation (2.17) becomes:

$$\Phi_{yy}(\omega) = \begin{bmatrix}
g_{11}g_{12}\phi_{11}^* & g_{11}g_{12}\phi_{12}^* & \cdots & g_{11}g_{mrn}\phi_{1m}^* \\
g_{22}g_{12}\phi_{21}^* & g_{22}g_{22}\phi_{22}^* & \cdots & g_{22}g_{mrn}\phi_{2m}^* \\
\vdots & \vdots & \ddots & \vdots \\
g_{mrn}g_{12}\phi_{m1}^* & g_{mrn}g_{22}\phi_{m2}^* & \cdots & g_{mrn}g_{mrn}\phi_{mn}^*
\end{bmatrix} \quad (2.18)$$

and for the $(i,j)_{th}$ element of $\Phi_{yy}(\omega)$ we have:

$$\phi_{ij}(\omega) = g_{ii}(j\omega) . g_{jj}(j\omega) . \phi_{ij}(\omega) \quad (2.19)$$

Expressing $g_{pq}(j\omega)$ in polar form:

$$g_{pq}(j\omega) = |g_{pq}(j\omega)| . e^{j\theta_{pq}} \quad (2.20)$$

and substituting in equation (2.19) we get:

$$\phi_{ij}(\omega) = \phi_{ij}(\omega) . |g_{ii}(j\omega)| . |g_{jj}(j\omega)| . e^{j(\theta_{ii}(\omega)-\theta_{jj}(\omega))} \quad (2.21)$$

Now if we could transmit the input signal to the output in such a way that the PSDs of the two signals are equal the problem would have been solved.
That however would mean that we require

$$\left| g_{ii}(j\omega) \right| \cdot \left| g_{jj}(j\omega) \right| \cdot e^{j(\theta_{ii}(\omega) - \theta_{jj}(\omega))} = 1 \quad (2.22)$$

for \( i \neq j \),

and just

$$\left| g_{ii}(j\omega) \right| = 1 \quad (2.23)$$

for \( i = j \), and for all \( i \).

Equation (2.22) means that we need a flat gain curve of the frequency response, with the constant gain equal to 1, over the frequency range of interest. If this is the case however, equations (2.21) and (2.22) show that the diagonal elements of \( \Phi_{uu}(\omega) \) will be reproduced in the output of the system, while at the same time the off diagonal elements of \( \Phi_{yy}(\omega) \) will have the same magnitude as the corresponding elements of \( \Phi_{uu}(\omega) \) but possibly a phase shift will exist between them. Furthermore, assuming a minimum phase and open-loop stable system, this phase shift will be very small at low frequencies. This will be true as the gain-phase relationship of the system would dictate, since the gain curves of the frequency responses are identical up to a certain frequency. If the gain roll-off of the different transfer function elements of \( G(s) \) is the same, then the phase shift \( \theta_{ii}(\omega) - \theta_{jj}(\omega) \) will be approximately zero for frequencies up to the system bandwidth.

The above analysis show that the control objectives for the single-input single-output case can be extended to the case of multi-actuator tests. In particular we should aim to achieve a system with no interaction, and flat gain curves of frequency...
response, with unity constant gain.

The additional requirements of insensitivity to disturbances and noise reduction will be expanded in a later chapter, but it must be mentioned here that they should be central to design of controllers for multivariable systems. In particular the property of insensitivity to disturbances, would be very useful in the case of fatigue tests for the following reason. In these tests the controller is designed to give good performance for one particular test object. Test object changes however are frequent, and there is not always the case that the dynamics of all test objects are identical. Additionally the dynamics of the test object may change as the test progresses. One example of this change in test object dynamics is the change that can occur to a complex structure due to constituent component deterioration in a long test. System parameter variations due to change in the test object dynamics can however be modelled as disturbances. Good disturbance rejection properties of the control scheme, will therefore ensure good accuracy throughout the test. Furthermore it will eliminate the need to design different controllers for different test objects with similar dynamic characteristics. This property can only be achieved with a feedback control scheme as it will be shown later. The requirement for "robust" control schemes, as often such schemes are called, is of course as useful in the SISO case. The difference is that the assessment of robustness in the multivariable case is a lot more difficult and less accurate. Its importance however remains the same, and considerable advantages are to be gained, as it has already been indicated, by designing robust control systems for dynamic testing.

Another obvious requirement of any control system is that of stable operation. This requirement is usually extended to require stability even when certain components fail during operation, in other words it is asked of the control system to have "integrity", under component failure (usually failure of
transducers or actuators). The importance of this can be seen if we consider the tests of expensive prototype aerospace structures and the damage that could be caused to them when, say, a transducer fails, if a system is of poor integrity. Assessing the integrity of a multivariable control system however is a lot more complex than in the case of single-input single-output systems. The reason for this added complexity is that failure in one loop in a multivariable system, can cause instability in another loop. Clearly non-interacting multivariable systems do not present this difficulty in assessing the integrity of the system, since they can be regarded as a collection of SISO systems. In fact it will be shown later that if interaction is reduced beyond a certain level, high integrity systems can be designed relatively easily.

2.4.3. Current test practices

It has been seen that what makes multi-actuator tests different in control terms, from single actuator tests, is the mechanical coupling or interaction that exists between different inputs and outputs. If the mechanical coupling is non existent or weak, the problem can be treated as a collection of single actuator tests. In fact interaction is ignored all together in the majority of multi-actuator tests, and single loop controllers are tuned in each individual actuator's servo system, as each loop is sequentially closed. This approach needs a lot of experience in order to achieve "good" results in the relatively simple case where interaction between different actuators is low. In the presence of significant interaction however the accuracy that can be realistically achieved is poor. Stability margins can not be assessed with any degree of accuracy and as a result stability can, and sometimes is, lost. Also no serious attempt can be made to provide insensitivity and noise rejection when the
interaction present in the system is ignored.

It is therefore apparent that for multi-actuator tests where more accurate simulation of service conditions, and in general better performance is required, a control scheme that would take interaction into account should be used. Such a scheme was devised in the late seventies by MTS. It is usually referred to as the "Remote Parameter Control" (RPC) scheme, and is used today in the most sophisticated tests of automotive structures almost exclusively. The principles of this scheme are described below.

Consider a linear system which is described by a transfer function matrix \( G(s) \). In the case of structural testing \( G(s) \) will describe the combined dynamics of the test object, the actuators, and the transducers. Consider further that field measurements of loads at certain points in the structure have been made. Let the vector of the spectral densities of these loads be \( \Phi_{yy}(\omega) \). The aim of the test then is to reproduce such loads at the same points in the structure that will have spectral densities given by the vector \( \Phi_{yy}(\omega) \). If the control input that will produce such loads is \( u(t) \) with spectral density \( \Phi_{uu}(\omega) \) then the input output relationship is given in equation (2.17):

\[
\Phi_{yy}(\omega) = G(j\omega) \cdot \Phi_{uu}(\omega) \cdot G^*(j\omega)
\]

Now the spectral densities have been evaluated at discrete frequencies \( \omega_k \), and therefore (2.17) becomes:

\[
\Phi_{yy}(\omega_k) = G(j\omega_k) \cdot \Phi_{uu}(\omega_k) \cdot G^*(j\omega_k)
\]
and finally

$$\Phi_{uu}(\omega_k) = G^{-1}(j\omega_k) . \Phi_{yy}(\omega_k) . \{G^*(j\omega_k)\}^{-1}$$  \hspace{1cm} (2.24)

for each $\omega_k$

Once the spectral density of the input has been determined the input vector itself can be calculated in the time domain, as it is shown below. Let $u(j\omega_k)$ is the input vector description in the frequency domain. Now it is desired to find a transformation $\Xi(j\omega_k)$ such that:

$$u(j\omega_k) = \Xi(j\omega_k) . w(j\omega_k)$$  \hspace{1cm} (2.25)

where $w(j\omega_k)$ is a vector of uncorrelated white noise signals. Taking PSDs in equation (2.25) :

$$\Phi_{uu}(\omega_k) = \Xi(j\omega_k) . \Phi_{ww}(\omega_k) . \Xi^*(j\omega_k)$$  \hspace{1cm} (2.26)

Since the white noise inputs are uncorrelated, $\Phi_{ww}(\omega_k)$ is diagonal. If in addition the white noise inputs are generated each with unity spectral density, then

$$\Phi_{ww}(\omega_k) = I$$  \hspace{1cm} (2.27)

and (2.26) becomes

$$\Phi_{uu}(\omega_k) = \Xi(j\omega_k) . \Xi^*(j\omega_k)$$  \hspace{1cm} (2.28)
\( \Phi_{uu}(\omega_k) \) is however known and if \( \Xi(j\omega_k) \) is restricted to be triangular with positive diagonal elements, it can be calculated from equation (2.28). The frequency domain description of the input vector is obtained from equations (2.25) as \( u(j\omega_k) = \Xi(j\omega_k) \) since \( w(j\omega_k) \) is a vector of unit elements. Once \( u(j\omega_k) \) is known \( u(t) \) can be determined by inverse Fourier transforms.

Only the frequency response of the transfer function matrix \( G(s) \) is needed to apply the RPC method and this is determined experimentally. The calculation of the control input \( u(t) \), as described above is however only approximate. Errors are introduced in the identification process, in the inversion of the transfer function matrix and its transposed conjugate; and in the calculation of \( u(t) \) after \( \Phi_{uu}(\omega) \) has been determined. The errors will be greater particularly at higher frequencies, where frequency response measurements are noisier. Also at high frequencies the transfer function matrix becomes more and more ill-conditioned (or more nearly singular) and sometimes introduces substantial numerical errors in the inversion algorithm. To overcome this problem an iterative process is used to adjust the calculated \( u(t) \), so that greater accuracy in the reproduction of service loads is achieved.

The great strength of the RPC scheme is the automation it introduces in multi-actuator tests. Also the method effectively tries to decouple the system, and to the degree that it is successful achieves greater accuracy than traditional "single-loop" control schemes which ignore interaction. There are however certain disadvantages associated with this method as it will be shown in the next subsection, which could be alleviated by the use of frequency domain multivariable design techniques.
2.4.4 The need for multivariable control.

The RPC method provides a dramatic improvement in accuracy, in comparison with sequential loop closing. There are however a number of problems associated with it.

The identification process and the iterations to achieve the required accuracy need to be done every time the test object changes. This means that the test object is in effect pre-loaded before the test commences. It is argued that the effect this would have in a long test situation is limited. On the other hand we cannot run any test with no pre-loaded test object, since the identification and the iteration process has to be repeated every time. The pre-loading effect therefore cannot be quantified.

The main problem with RPC however, results from the fact that it is an open-loop scheme. Namely the uncertainty of the effect that changes of system parameters or temporary disturbances could have in system performance. To quantify this statement consider figure 2.2. The vector \( d(s) \) models changes in system parameters that occur due to model inaccuracies or due to change of parameters due to system time dependency, as well as other unpredicted temporary disturbances. The output vector is then given by:

\[
y(s) = G(s) . u(s) + d(s)
\]

(2.29)

From equation (2.29) we see that the disturbance is propagated to the output unaltered. For accuracy to be maintained, \( d(s) \) should be small for the whole duration of the test. We have however no means of attenuating \( d(s) \), should it become unacceptably large. To reduce this uncertainty high quality test equipment has to be used and this has obvious financial implications.
In $d(s)$ we could incorporate the system parameter change due to changes of test objects. If we had a way of attenuating $d(s)$ it would clearly be useful.

All these considerations point to a multivariable feedback design approach. Indeed the main reason for using feedback is to eliminate uncertainty. To see how this is achieved consider figure 2.3. Here $K(s)$ is the controller that should be designed to provide the required performance. The erroneous noise in the transducer signals is represented by the vector $m(s)$. There is also a prefilter $P(s)$ which is designed after $K(s)$ to provide the necessary accuracy. Next it will be shown how both unwanted quantities $d(s)$ and $m(s)$ can be attenuated by designing an appropriate forward path controller $K(s)$.

From figure 2.3:

\[ y(s) = d(s) + G(s)K(s)\left[P(s)r(s) - y(s) - m(s)\right] \quad (2.30) \]

or

\[ (I + G(s)K(s))y(s) = d(s) + G(s)K(s)P(s)r(s) - G(s)K(s)m(s) \]

Now define the sensitivity matrix:

\[ S(s) = \left[I + G(s)K(s)\right]^{-1} \quad (2.31) \]

and the loop transfer function matrix:

\[ T(s) = S(s)G(s)K(s) = \left[I + G(s)K(s)\right]^{-1}G(s)K(s) \quad (2.32) \]
then

\[ y(s) = S(s) \cdot d(s) + T(s) \cdot P(s) \cdot r(s) - T(s) \cdot m(s) \]  

(2.33)

Equation (2.33) shows that both \( d(s) \) and \( m(s) \) are multiplied by dynamic matrices. These matrices, \( S(s) \) and \( T(s) \) can therefore be used to reduce the effects of the disturbances and the transducer measurement noise. The controller \( K(s) \) should therefore be designed to give "small" \( S(s) \) at frequencies where \( d(s) \) is expected to be large, and "small" \( T(s) \) at frequencies where \( m(s) \) is large. Of course \( S(j\omega) \) and \( T(j\omega) \) are complex matrices for each specific frequency point \( \omega \), and no exact measurable and physically meaningful size is associated with matrices. Ways of assessing the "size" of a transfer function matrix are given in a later chapter. In the same chapter methods of designing the controller \( K(s) \) to give the required system performance are described.

The required accuracy is achieved by designing \( P(s) \) after \( K(s) \) is designed, so that

\[ T(s) \cdot P(s) \approx I \]  

(2.34)

The reference input will then be transmitted almost unaltered to the output.
Figure 2.1. LOW-HIGH, HIGH-LOW AND PSEUDO-RANDOM LOADING.
Figure 2.2 Block diagram of open-loop system

Figure 2.3 Block diagram of closed-loop system.
CHAPTER 3

THE TEST RIG

It has already been mentioned that the qualitative difference between a selection of single-input single-output systems and a true multivariable system is the presence of interaction in the latter. To test therefore the application of multivariable control theory in the field of structural testing an experimental rig is needed to give significant interaction in a multi-channel test situation. Such a rig was build in a previous project and was used in the course of the work that gave rise to this thesis.

The test rig consists of a test frame on which an elastic steel beam is clamped at one end with the other end free. This cantilever beam is forced into vibration by two electrodynamic shakers positioned at mid-point and its free end as shown in figure 3.1. The vibrators are fastened to the test frame and have their moving coil armature assembly attached to the cantilever via "push - rods". The forcing signals which could be random or sinusoidal are generated from signal generators and are applied to the vibrators through two power amplifiers. The force generated by the vibrators act in a horizontal plane thus eliminating gravitational effects. Two linear variable differential transformers (LVDT) are used to monitor the deflections on the beam at the points where the vibrators are attached to the beam.

3.1 THE TEST FRAME

The test frame is constructed out of two longitudinal 6"x 3" mild steel channels held together by angle piece
cross-members. A mild steel plate 8" wide and 1/2" thick is mounted on the upper surface of each longitudinal channel. One of these plates forms the basis on which the electrodynamic vibrators are fastened. A similar thickness plate but wider is fastened across the channels. This last plate further stiffens the test frame and forms a base for a solid mild steel block of dimensions 6"x 6"x 5.5" which is fastened down to the base by bolts. This mild steel block clamps solidly the clamped end of the cantilever. The whole structure weighs over 210 Kilograms forming a virtually rigid immovable reference. The whole assembly is shown on figures 3.2a and 3.2b where the positions of the vibrators and the displacement transducers are also marked.

Usually the motion between the test fixture and test item should depend only upon the input signal and the dynamics of the test object. Steel structures however have very little inherent energy dissipation facilities and thus are very susceptible to excitation into resonances. Any amount of energy therefore transmitted from the vibrator trunnion to the test frame while the vibrator is active is going to act as an excitation force to the frame. The frame can also be excited by vibrations of the floor onto which it is mounted, caused by a variety of sources such as nearby road traffic, building vibrations etc. These effects were quantified in a previous project by sine-sweeping each vibrator individually in the frequency range of 0 - 220 Hz. As it was expected the test frame was excited into resonant modes at different frequencies, the levels of which would have adversely affected the tests. The easiest way to alleviate this problem is to use some sort of vibration isolators between the test rig and the vibrators and the floor. The various types of isolators available in the market were tested and finally corrugated rubber mats where used to isolate the vibrators from the frame while anti-vibration felt pads were selected to isolate the test frame from the floor. In addition to this when the rig was first commissioned in the department of Electrical
Electronics and Systems Engineering at Coventry Polytechnic, the frame was bolted to the floor. This way the effective mass of the test frame was increased and the longitudinal channels where further stiffened. When the rig was transferred to LUT however, time and financial constraints caused this option to be abandoned.

One final problem associated with the rig arises from the fact that one end of the beam is rigidly clamped. This means that in response to an horizontal applied force the beam will not only translate, but it will also rotate as indicated in figure 3.3. If we were to succeed in measuring the displacement \( v \) due to the applied force alone, it is essential that the beam is allowed to rotate too without any restraints imposed by the attachment of the vibrators. It is in fact highly unlikely that an attached vibrator will be flexible enough in rotational and lateral directions to permit the necessary freedom of movement for the structure under test. One practical solution to this problem is to introduce between the vibrator and the structure a special drive rod which is stiff in the direction of excitation but flexible in all other directions. This is actually the solution adopted here. Although it is possible that resonances of the push rods may interfere with the measurements, these effects can be confined to relatively high frequencies by suitable choice of the drive rod material.

3.2 THE VIBRATORS

The two electrodynamic vibrators used in the test rig are the Ling Dynamic Systems 403 series model. The housing of each vibrator contains a permanent magnet. A magnetic field is thus produced in the annular gap that surrounds the armature drive coil. The armature and the drive coil assembly are suspended on two laminated fibre flexures. The flexures are attached to the vibrator body by support pillars.

The armature assembly consists of a cylindrical copper coil
bonded to a cast aluminium radial finned structure. The armature is located around the centre pole assembly and occupies part of the air gap in the magnet. The two laminated flexures bonded to the armature provide axial support for the armature as well as providing lateral and rotational restraint as shown in figure 3.4.

Armature drive current is supplied to the vibrator input socket and from there applied to the armature. The armature coil conductors are in right angles to the magnetic flux in the permanent magnet's air gap. When current is flowing therefore through the armature coil, a force is produced which is mutually perpendicular to the air gap flux and the direction of the armature current. Thus an alternating current produces an alternating force.

The electrical activity of an electrodynamic vibrator, can be modelled simply as an inductance and resistor in series representing the coil (figure 3.5), with an additional voltage drop representing the back e.m.f. due to the velocity of the moving coil through the magnetic flux of the permanent magnet [11].

The mechanical subsystem of the vibrator can be also simplified and considered to be an idealised mass-spring-damper system driven by a force proportional to the current (figure 3.6). The damping effects due to the flexure stiffness and the rubber seal can be ignored since they are negligible when compared to the damping created by the back e.m.f.

The force developed in the coil when current \( i \) is passed through is

\[ f = 2\pi n r B i = K_F i \quad (3.1) \]

where \( K_F = 2\pi n r B \) in N/A

- \( B \) = the magnetic flux density
- \( n \) = the number of coil turns
- \( r \) = the radius of the coil.
The back e.m.f. produced by the vibrator coil moving in the magnetic field is given by

\[ e = 2nrB \dot{v} \quad \text{where} \quad K_B = 2nrB \text{ in } \left[ \frac{V}{\text{ms}^{-1}} \right] \]  

From equation (3.1) and (3.2) we see that the force constant \( K_F \) is equal to the back e.m.f constant \( K_B \), for direct transfer of electrical power to mechanical power in the vibrator coil.

The above description of the vibrator dynamics can be represented by the the block diagram of figure 3.7, for no-load "condition". The block diagram of figure 3.8 describes the vibrator dynamics when driving a load.

### 3.3 CHOICE OF BEAM

It has already been mentioned that it was decided to configure the rig in such a way as to present the worst possible control problem to the designer. In order to investigate which combination of beam and vibrator dynamics gives this worst case problem is desirable to use a relatively simple mathematical model of the beam. Such a mathematical model is developed next by combining the dynamics of the vibrators with a simple lumped parameter model for the beam.

Under dynamic conditions, the cantilever alone should be treated as a continuous mass system. A mathematical model which adheres to this principle will be developed in another part of this thesis. This method however leads to complicated models where the mass and stiffness of the beam and the vibrators are distributed to the beam's natural modes. Such
models therefore do not lend themselves to the analysis required in order to establish what values the parameters of the beam should have, to present the desired "worst case problem". To arrive at a model which will serve our present purpose, small static deflections will be considered and the beam will be modelled as shown in figure 3.9. Note here that an acceptable approximation to the dynamic behaviour of the beam will only be given under the assumption that the deflections of the beam are kept small.

From figure 3.9 we have:

For \( f_1 = 0 \), \( y_1 = \frac{51^3}{48EI} f_2 \); \( y_2 = \frac{1^3}{3EI} f_2 \) = \( \frac{161^3}{48EI} f_2 \)

and

For \( f_2 = 0 \), \( y_1 = \frac{1^3}{24EI} f_1 \) = \( \frac{21^3}{48EI} f_1 \); \( y_2 = \frac{51^3}{48EI} f_1 \)

and by superposition

\[
\begin{bmatrix}
  y_1 \\
  y_2
\end{bmatrix} = \frac{1^3}{48EI} \begin{bmatrix}
  2 & 5 \\
  5 & 16
\end{bmatrix} \begin{bmatrix}
  f_1 \\
  f_2
\end{bmatrix}
\]

or

\[ y = F.f, \]

where \( F \) is the flexibility matrix for the beam.

and

\[ f = F^{-1}y = Sy \]

\[ (3.4) \]
where $S$ is the stiffness matrix which from equation (3.3) is given by

$$S = F^{-1} = \frac{48EI}{7l^3} \begin{bmatrix} 16 & -5 \\ -5 & 2 \end{bmatrix} = \beta \begin{bmatrix} 16 & -5 \\ -5 & 2 \end{bmatrix}$$

(3.5)

where

$$\beta = \frac{48EI}{7l^3}$$

(3.5a)

and finally

$$f = \begin{bmatrix} f_1 \\ f_2 \end{bmatrix} = \frac{48EI}{7l^3} \begin{bmatrix} 16 & -5 \\ -5 & 2 \end{bmatrix} \begin{bmatrix} y_1 \\ y_2 \end{bmatrix} = SY$$

(3.6)

Consider now the entire rig. The block diagram of the entire rig can be seen in figure 3.10. From this the transfer function of the entire rig is given by

$$Q(s) = \left[I + \frac{K_F K_B}{R} sG(s)\right]^{-1} \frac{K_F}{R} G_{bv}(s)$$

(3.7)

and

$$Q^{-1}(s) = \left(\frac{K_F}{R}\right)^{-1} G_{bv}(s) \left[I + \frac{K_F K_B}{R} sG(s)\right]$$

$$= \frac{R}{K_F} G_{bv}^{-1}(s) + sK_B$$

(3.8)

The transfer function $G_{bv}(s)$ which describes the dynamics of the beam including the concentrated stiffnesses due to the
vibrators and the concentrated masses due to the vibrator armatures is defined by

\[ y(s) = G_{bv}(s) f(s) \]

and its inverse will be defined by

\[ f(s) = G_{bv}^{-1}(s) y(s) \]

Therefore

\[ G_{bv}^{-1}(s) = M s^2 + S + K_v I \]  \hspace{1cm} (3.9)

where the beam stiffness matrix \( S \) is given by equation (3.5) and the mass matrix \( M \) from figure 3.9 and taking into account the mass of the vibrator armatures \( m_a \), is

\[ M = \begin{bmatrix} m_1 + m_a & 0 \\ 0 & m_2 + m_a \end{bmatrix} = \begin{bmatrix} m & 0 \\ 0 & m \end{bmatrix} \]  \hspace{1cm} (3.10)

Equation (3.8) now becomes

\[ Q^{-1}(s) = \frac{R}{K_F} \left[ M s^2 + S + K_v I \right] + K_B I s \]

\[ = \frac{R}{K_F} \left[ M s^2 + \frac{K_v K_B}{R} I s + S + K_v I \right] \]  \hspace{1cm} (3.11)
Or finally

\[
Q^{-1}(s) = \begin{bmatrix}
-5\beta & ms^2 + \frac{K_F K_B}{R}s + (16\beta + K_v) \\
-5\beta & ms^2 + \frac{K_F K_B}{R}s + (2\beta + K_v)
\end{bmatrix}
\]

(3.12)

The worst case control problem would be presented by the system configuration which does not allow the true multivariable system to be reduced to a collection of independent single-input single-output systems. In other words the worst case control problem will be given by the rig configuration that has significant interaction between different input/output pairs, which cannot be easily eliminated.

From equation (3.12) the interaction in the inverse complex plane can be assessed directly. To assess interaction in the direct plane put

\[
\hat{Q}(s) = Q^{-1}(s) = \begin{bmatrix}
\hat{q}_{11} & \hat{q}_{12} \\
\hat{q}_{21} & \hat{q}_{22}
\end{bmatrix}
\]

(3.13)

then

\[
\hat{Q}(s) = Q^{-1}(s) = \frac{1}{\hat{q}_{11}\hat{q}_{22} - \hat{q}_{12}\hat{q}_{21}} \begin{bmatrix}
\hat{q}_{22} & -\hat{q}_{12} \\
-\hat{q}_{21} & \hat{q}_{11}
\end{bmatrix}
\]

(3.14)

Note that in equation (3.13) and (3.14) the elements of the matrix \(Q(s)\) are functions of \(s\) as well but are written in the above form for brevity.

The interaction therefore in the direct complex plane is
given by the ratios

\[
\begin{align*}
 r_1 &= \frac{|-\hat{q}_{12}|}{|\hat{q}_{22}|} = \frac{|-\hat{q}_{21}|}{|\hat{q}_{22}|} \quad (3.15a) \\
 r_2 &= \frac{|-\hat{q}_{21}|}{|\hat{q}_{11}|} = \frac{|-\hat{q}_{12}|}{|\hat{q}_{11}|} \quad (3.15b)
\end{align*}
\]

The first ratios in equations (3.15a) and (3.15b) give the row-wise interaction while the column-wise interaction is given by the second ratios. The equalities are valid since from (3.12) we have

\[
\hat{q}_{12} = \hat{q}_{21} = -\beta
\]

The system will be non-interacting when \( r_1 = r_2 = 0 \). Conversely the greater the values of the ratios \( r_1 \) and \( r_2 \) the more interaction is present in the system. Also from equation (3.12) we see that interaction (row or column-wise) will be more severe in the first channel, i.e. \( r_1 > r_2 \). It is therefore justified to consider only \( r_1 \) in the analysis that follows.

In the frequency domain the interaction in the first channel is given by:

\[
\begin{align*}
 r_1(j\omega) &= \frac{|-q_{12}(j\omega)|}{|q_{22}(j\omega)|} = \frac{|5\beta|}{|(-m\omega^2 + 2\beta + K_v) + j(\frac{K_F K_B}{R} \omega)|} \\
 &= \frac{5\beta}{\sqrt{(2\beta + K_v - m\omega^2)^2 + (\frac{K_F K_B}{R} \omega)^2}}
\end{align*}
\]
Now since the parameters of the beam are constrained by the manufacturer the only way of altering the value of the ratio in equation (3.16) is by changing the dimensions and the stiffness of the beam, thereby changing the value of $\beta$. Three clear ranges for the value of $\beta$ emerge:

1. $2 \beta > K_v$. This corresponds to a very stiff beam and it will result in severe interaction. In this case however the effects of the frequency $\omega$ will be limited. Making the system non-interacting therefore at zero frequency will dramatically reduce interaction throughout the frequency range. The obvious choice for a decoupling controller will therefore be:

$$K = G^{-1}(0)$$

since $K$ is real and will completely decouple the system at zero frequency.

2. $\beta$ very small. This corresponds to a very soft beam. It can be seen from equation (3.16) that it is possible to choose a very soft beam to give negligible interaction.

3. $\beta$ is such that $r_1(0) = 1$. This corresponds to beams whose stiffness is between the two extreme cases. In this case the contribution of the frequency terms is significant over the frequency range. The interaction is also significant from zero frequency up to the system's bandwidth. From equation (3.16) then after substituting 0 for $\omega$ we get:

$$\frac{5 \beta}{2 \beta + K_v} = 1 \quad \text{at } \omega = 0 \text{ rad.s}^{-1}$$
this gives

$$\beta = \frac{K_v}{3} \tag{3.17}$$

In fact a process of trial and error shows that the most difficult situation occurs when $\beta$ is slightly greater than $K_v/3$ for a steel beam driven by a 400 series vibrator.

### 3.4 THE VIBRATOR PROTECTION CIRCUIT

Problems were encountered with the vibrator protection circuits from an early stage. Inadequate protection against current surges resulted in damaging the armature of one the vibrators in three different occasions. Although costly repairs of the vibrators were carried out by the manufactures, no reason was given why such damages should occur. Investigations into the electronic circuit of the power amplifiers used with the vibrators showed that two main reasons affected the satisfactory protection of the vibrators:

The first problem, which was the easiest to correct, was that the current cut-off level of the amplifiers clamp circuit was set by the manufacturers for force-cooling operation of the vibrators. This meant that the current supply to the vibrators was not cut-off before it rose to 18 Amp. when the maximum allowable current for safe operation was only 9 Amps.

The second problem was associated with the way the current was measured. Any potential damage to the vibrator would be the result of the heating effect of the current through vibrator armature coils. It is therefore the RMS value of the current that is important in this instant. In the series 300 amplifier however the RMS value of the current is approximated by rectifying and averaging the current waveform. The accuracy of the approximation therefore depends on the purity of the sinewave input. This clearly does not give adequate
protection since random input test are envisaged in the future.

One more practical problem was that whenever the amplifier entered the "clamp" mode, a latch is set that keeps the amplifier output clamped. The only way to reset the latch was to move the MASTER GAIN knob to the RESET position. The reason for this is that effectively eliminates the possibility of abrupt signal changes after reseting, since at this instant the MASTER GAIN of the amplifier is zero and it can only be brought up to its desirable level relatively slowly. This however results in poor test repeatability since it is very difficult to set the gain accurately in its prior position after a RESET has been effected. This is particularly a problem in closed loop operation when variation in gain will affect the dynamic behaviour of the system.

These considerations lead to the decision to design and build an external protection circuit whose operation would be based on the monitoring of true RMS current output. Furthermore the latch that effected the current clamp, should be able to be reset by a push-button mounted on the board of this circuit.

To this end true RMS current transducers (one for each amplifier) should be used to measure the current. Such transducers found in the market today do not have a useful frequency range extending from D.C. to some high frequency limit. They can only guarantee satisfactory operation in frequencies above 10 Hz.

To alleviate this problem it was decided to monitor the instantaneous value of the current at low frequencies and if that exceeded a given critical value the amplifier output would be clamped.

A schematic block diagram of the designed circuit is given in figure 3.12. Both instantaneous and RMS measurements are supplied by the same transducer for each of the two amplifiers whose current output are measured. Two 2nd order Sallen and Key low pass filters are used to ensure that instantaneous
current values cannot cause a clamp condition if they occur at relatively high frequencies. Both filters have a common corner frequency of 15Hz. Above this frequency the level of the RMS measurement becomes the important one. Either the RMS value or the instantaneous values of current can cause the output to be clamped. They are fed through comparators to an OR gate whose output can set the RS flip-flop. This, when set, outputs a LOW logical level signal to the amplifier's EXTERNAL CLAMP terminal causing it to cut-off the current supply to the vibrator. The clamp signal can only be reset by using the push button seen on figure 3.12. Safety against abrupt signal changes was enhanced by altering the value of the capacitor across the output of the amplifier's internal clamp circuit (Capacitor C15 in figure 3.11) from 470nF to 10μF. This will ramp-up the output signal to its set position slower, after the signal clamp is reset.

The detailed schematic diagram of the external circuit can be seen on figure 3.13. The amplifier vibrator and external clamp circuit interconnections are shown in figure 3.14.

The circuit was build on a PCB and mounted in the amplifier main racking cabinet. Satisfactory operation of the rig has been observed ever since.
Figure 3.1 Schematic diagram of experimental rig.
Figure 3.2a: Plan view of test rig frame

Figure 3.2b: Side view of test rig frame
Figure 3.3 Movement of beam in response to an horizontal force
1. Top Cover retaining screws
2. Dust Seal
3. Upper Flexure (Bonded to Moving Coil Assembly)
4. Lower Flexure (Bonded to Moving Coil Assembly)
5. Lower Flexure Support Pillars
6. Top Cover Retaining Pillars
7. Flexure Support Ring
8. Trunnion Attachment and Air Cooling Inlet
9. Magnet Housing
10. Cooling Air Exhaust Ports
11. Column Permanent Magnet
12. Moving Platform and Drive Coil Assembly
13. Shaker input Drive Socket
14. Electrical Connections
15. Upper Flexure Support Pillars
16. Top Cover

Figure 3.4. SECTIONAL VIEW OF SHAKER.
Figure 3.5 Vibrator electrical subsystem

Figure 3.6 Vibrator's mechanical subsystem
Figure 3.7  Block diagram of vibrator dynamics. No load condition

Figure 3.8  Block diagram of vibrator dynamics when driving a load.
Figure 3.9  Schematic diagram of beam model as a lumped parameter system.

Figure 3.10  Block diagram of the experimental rig.
Figure 3.11. INTERNAL CLAMP CIRCUIT OF SERIES 300 AMPLIFIER.
Figure 3.12  Block diagram of protection circuit
Figure 3.13 Schematic diagram of clamping circuit.
Figure 3.14 Interconnections between the external clamp circuit, the power amplifiers and the vibrators.
CHAPTER 4

MODELLING AND SIMULATION OF THE
EXPERIMENTAL RIG

4.1 INTRODUCTION

The problem of the mathematical modelling of the entire rig, is dominated by the method which will be used to derive the model of the cantilever beam. The reason for this is that the dynamics of the electrodynamic vibrators are relatively simple and well defined as it has already been established in the previous chapter.

The role of the mathematical model to be developed is primarily to give a better understanding of the dynamics of the system and the interconnections of the various subsystems. It should also lend itself to simulation of the rig. It would be desirable to simulate not only the system outputs, displacements in this case, for a given input function, but other variables as well, such as current through the vibrator, velocity at measurement points or load applied to the beam. Mathematical models of the rig will not be used in the controller design since the necessary frequency response data can easily be obtained experimentally. It would however be desirable to be able to obtain mathematical models and subsequently simulate with the minimum effort the behaviour of the system, when the number of vibrators and their positions, as well as the number of the system outputs and their positions, vary. This would facilitate extending the
investigation for to a number of cases which would result when varying the number and positions of shakers and LVDT's, without having to build or acquire the necessary hardware.

It has already been said in a previous chapter that the frequency response of the system completely describes the response to the inputs it is subjected to, in the various types of dynamic structural tests. It is therefore clear that the mathematical model need only be able to simulate the system relatively accurately in the frequency range of interest.

There are a number of different methods of obtaining a mathematical model for a cantilever beam. Two such methods have already been used in a previous projects [12], one giving a lumped parameter model of the beam and the other obtained using finite element analysis. The lumped parameter method gives a simple but rather crude model of the beam. Its accuracy could of-course be increased by increasing the number of lumped masses used, but that would increase the degree and complexity of the model considerably. The model obtained using the finite element method gives considerably more accurate results, but it produces higher order models. The common characteristic however of both methods is that their accuracy depends on the number of spatially distributed elements or lumped parameters considered. Adding or moving vibrators or LVDT's to different locations will therefore affect the choice of the number and/or location of these elements or lumped parameters. A new mathematical model of the beam, even when the beam itself is not altered, will therefore be necessary for each such addition or change in position of vibrators and LVDT's. This is the reason which makes the use of these methods impractical in investigating the effects of varying the number and position of shakers and LVDT's in the system's dynamic behaviour and therefore the controller design.

A method for obtaining mathematical models with less effort in these circumstances is developed.
method treats the beam as a continuous elastic body with an infinite number of natural modes. The number of modes which need to be considered for the model can be assessed directly by its frequency response. A program to automate the modelling and simulate the rig is also developed. The user needs only specify the parameters of the beam and the vibrators (e.g. mass stiffness etc) and the positions of the vibrators and LVDT's along the beam.

4.2 LATERAL FREE VIBRATION OF A UNIFORM CANTILEVER

In this section a mathematical model for the free vibration of a uniform Euler-Bernoulli cantilever beam is developed.

Consider the free body diagram of an element $dx$ of the cantilever beam shown in figure 4.1. From Newton's second law of motion the dynamic force equation in the lateral direction is

$$\frac{m}{dt^2} \frac{\partial^2 u(x,t)}{dx} = V - (V + \frac{dV}{dx}) \quad (4.1)$$

Where $m$ is the mass per length for the beam, $u(x,t)$ is the lateral deflection of the beam at a distance $x$ from the clamped end, and $V$ is the shear force.

Summing the moments $M$ about any point on the right face of the element $dx$ yields:

$$\frac{\partial M}{\partial x} dx - V dx = 0$$

-4.3 -
investigation for to a number of cases which would result when varying the number and positions of shakers and LVDTs, without having to build or acquire the necessary hardware.

It has already been said in a previous chapter that the frequency response of the system completely describes the response to the inputs it is subjected to, in the various types of dynamic structural tests. It is therefore clear that the mathematical model need only be able to simulate the system relatively accurately in the frequency range of interest.

There are a number of different methods of obtaining a mathematical model for a cantilever beam. Two such methods have already been used in a previous projects [12], one giving a lumped parameter model of the beam and the other obtained using finite element analysis. The lumped parameter method gives a simple but rather crude model of the beam. Its accuracy could of-course be increased by increasing the number of lumped masses used, but that would increase the degree and complexity of the model considerably. The model obtained using the finite element method gives considerably more accurate results, but it produces higher order models. The common characteristic however of both methods is that their accuracy depends on the number of spatially distributed elements or lumped parameters considered. Adding or moving vibrators or LVDT's to different locations will therefore affect the choice of the number and/or location of these elements or lumped parameters. A new mathematical model of the beam, even when the beam itself is not altered, will therefore be necessary for each such addition or change in position of vibrators and LVDT's. This is the reason which makes the use of these methods impractical in investigating the effects of varying the number and position of shakers and LVDT's in the system's dynamic behaviour and therefore the controller design process.

A method for obtaining mathematical models with the minimum of effort in these circumstances is developed here. This
method treats the beam as a continuous elastic body with an infinite number of natural modes. The number of modes which need to be considered for the model can be assessed directly by its frequency response. A program to automate the modelling and simulate the rig is also developed. The user needs only specify the parameters of the beam and the vibrators (e.g. mass stiffness etc) and the positions of the vibrators and LVDT's along the beam.

4.2 LATERAL FREE VIBRATION OF A UNIFORM CANTILEVER

In this section a mathematical model for the free vibration of a uniform Euler-Bernoulli cantilever beam is developed.

Consider the free body diagram of an element dx of the cantilever beam shown in figure 4.1. From Newton's second law of motion the dynamic force equation in the lateral direction is

\[
\frac{\partial^2 u(x,t)}{\partial t^2} \frac{dx}{m} = V - (V + \frac{dV}{dx})
\] (4.1)

Where \( m \) is the mass per length for the beam, \( u(x,t) \) is the lateral deflection of the beam at a distance \( x \) from the clamped end, and \( V \) is the shear force.

Summing the moments \( M \) about any point on the right face of the element \( dx \) yields:

\[
\frac{\partial M}{\partial x} dx - V dx = 0
\]
From elementary strength of materials, the beam curvature and the moments $M$ are related by

$$EI \frac{\partial^2 u(x,t)}{\partial x^2} = M \quad (4.3)$$

where $EI$ is the flexural stiffness of the beam expressed as a product of the Young's modulus and the second moments of area for the beam.

Equations (4.1) and (4.2) give

$$m \frac{\partial^2 u(x,t)}{\partial t^2} = - \frac{\partial V}{\partial x}$$

and

$$\frac{\partial^2 M}{\partial x^2} = \frac{\partial V}{\partial x}$$

combining the above two equations with equation (4.3) we obtain

$$m \frac{\partial^2 u(x,t)}{\partial t^2} = - \frac{\partial^2}{\partial x^2} (EI \frac{\partial^2 u(x,t)}{\partial x^2})$$
or

\[ m \frac{\partial^2 u(x,t)}{\partial t^2} = - EI \frac{\partial^4 u(x,t)}{\partial x^4} \]  \hspace{1cm} (4.4)

but since the flexural stiffness EI, and the mass per length m are constant for any given uniform beam, putting

\[ a^2 = \frac{EI}{m} \]  \hspace{1cm} (4.5)

gives

\[ \frac{\partial^2 u(x,t)}{\partial t^2} + a^2 \frac{\partial^4 u(x,t)}{\partial x^4} = 0 \]  \hspace{1cm} (4.6)

Equation (4.6) is the beam equation for its lateral vibration. In the special case of the cantilever beam the boundary conditions are:

Zero displacement, at the clamped end, which gives

\[ u(0,t) = 0 \]  \hspace{1cm} (4.7)

and zero slope at the clamped end which gives

\[ \frac{\partial}{\partial x} u(0,t) = 0 \]  \hspace{1cm} (4.8)
Zero moment at free end gives

\[ M(l, t) = EI \frac{\partial^2 u(l, t)}{\partial x^2} = 0 \implies \]

\[ \frac{\partial^2}{\partial x^2} u(l, t) = 0 \quad (4.9) \]

and finally zero shear force at the free end gives

\[ V(l, t) = EI \frac{\partial^3 u(l, t)}{\partial x^3} = 0 \implies \]

\[ \frac{\partial^3}{\partial x^3} u(l, t) = 0 \quad (4.10) \]

Now by the method of separation of variables (D'Alembert's solution) the formal solution of equation (4.6) is:

\[ u(x, t) = \sum_{i=1}^{\infty} \phi_i(x) q_i(t) \quad (4.11) \]

where \( \phi_i(x) \) is a function of the space variable alone and defines the shape of the vibration of the \( i \)th mode of the beam. The function \( q_i(t) \) is independent of \( x \), and defines the magnitude of the vibration of the \( i \)th mode as this changes with time.

Equation (4.6) can also be expressed in the following way

\[ M''(u) + L(u) = 0 \quad (4.12) \]

-4.6 -
Where $L$ and $M$ are linear differential operators involving only the space variable $x$. $L$ and $M$ are therefore of the form:

$$L = \alpha_0 + \alpha_1 \frac{\partial}{\partial x} + \alpha_2 \frac{\partial^2}{\partial x^2} + \ldots$$

and

$$M = \beta_0 + \beta_1 \frac{\partial}{\partial x} + \beta_2 \frac{\partial^2}{\partial x^2} + \ldots$$

Therefore

$$L(u(x,t)) = \alpha_0 u(x,t) + \alpha_1 \frac{\partial}{\partial x} u(x,t) + \alpha_2 \frac{\partial^2}{\partial x^2} u(x,t) + \ldots$$

and

$$M(\dddot{u}(x,t)) = \beta_0 \dddot{u}(x,t) + \beta_1 \frac{\partial}{\partial x} \dddot{u}(x,t) + \beta_2 \frac{\partial^2}{\partial x^2} \dddot{u}(x,t) + \ldots$$

A comparison between equation (4.4) and (4.12) gives the exact form of the linear operators:

$$L = \alpha_4 \frac{\partial^4}{\partial x^4} = EI \frac{\partial^4}{\partial x^4} \quad (4.13)$$

and

$$M = \beta_0 = m \quad (= \text{mass} / \text{length}) \quad (4.14)$$
Therefore the linear operator \( M \) is in this case reduced to a mere weighting factor.

The linear operators \( L \) and \( M \) are related to the beams kinetic and potential energy functions.

In particular since the inertial force is described by the operator \( M \) in equation (4.12), the kinetic energy \( T \) is

\[
T = \frac{1}{2} \int_0^L \dot{u} M(\ddot{u}(x,t)) \, dx \tag{4.15}
\]

Or

\[
T = \frac{1}{2} \int_0^L m \sum_{i=1}^{\infty} \phi_i(x) \dot{q}_i(t) \sum_{j=1}^{\infty} \phi_j(x) \dot{q}_j(t) \, dx
\]

\[
= \frac{1}{2} \sum_{i=1}^{\infty} \sum_{j=1}^{\infty} \dot{q}_i(t) \dot{q}_j(t) \int_0^L m \phi_i(x) \phi_j(x) \, dx \tag{4.16}
\]

But the normal modes are orthogonal functions, i.e.

\[
\phi_i(x) \phi_j(x) = \begin{cases} 
0 & \text{for } j \neq i \\
\phi_2^2(x), & \text{for } i = j
\end{cases}
\]

and also

\[
q_i(t) q_j(t) = \begin{cases} 
0 & \text{for } j \neq i \\
q_i^2(t), & \text{for } i = j
\end{cases}
\]
In view of the two orthogonal equations above, equation (4.16) becomes

\[
T = \frac{1}{2} \sum_{i=1}^{\infty} q_i^2(t) \int_0^L m \phi_i^2(x) \, dx = \frac{1}{2} \sum_{i=1}^{\infty} m_{ii} q_i^2(t) \tag{4.17}
\]

where

\[
m_{ii} = m \int_0^L \phi_i^2(x) \, dx \tag{4.18}
\]

is the generalised mass that corresponds to the ith mode.

The linear operator \( L \) describes the elastic properties of the beam. Hence the potential energy \( U \) is given by

\[
U = \frac{1}{2} \int_0^L u(x,t) L\{u(x,t)\} \, dx \tag{4.19}
\]

Substituting the value of \( u(x,t) \) from equation (4.11) and noting the orthogonal relation

\[
\int_0^L \phi_i(x) L\{\phi_j(x)\} \, dx = 0 \quad \text{for } i \neq j \tag{4.20}
\]

we obtain

\[
U = \frac{1}{2} \int_0^L \sum_{i=1}^{\infty} \phi_i(x) q_i(t) \int_0^L \left( \sum_{j=1}^{\infty} \phi_j(x) q_j(t) \right) \, dx
\]
Or
\[ U = \frac{1}{2} \sum_{i=1}^{\infty} \sum_{j=1}^{\infty} q_i(t) q_j(t) \int_0^L \phi_i(x) L_i(\phi_j(x)) \, dx \]

Or finally
\[ U = \frac{1}{2} \sum_{i=1}^{\infty} q_i(t)^2 \int_0^L \phi_i(x) L_i(\phi_i(x)) \, dx \]

\[ = \frac{1}{2} \sum_{i=1}^{\infty} k_{ii} q_i(t)^2 \] \quad (4.21)

where
\[ k_{ii} = \int_0^L \phi_i(x) L_i(\phi_i(x)) \, dx \] \quad (4.22)

is the generalised stiffness of the cantilever beam for the ith mode.

Now the Lagrange equation of motion for small oscillation of a uniform beam is
\[ \frac{d}{dt} \left( \frac{\partial T}{\partial q_i} \right) + \frac{\partial U}{\partial q_i} = 0 \] \quad (4.23)

Substituting the kinetic energy \( T \) and the potential energy \( U \) from equations (4.17) and (4.21) into equation (23) we get for the ith mode
\[ \frac{d}{dt} \left( m_i \dot{q}_i(t) \right) + k_{ii} q_i(t) = 0 \]
Or

\[ m_{ii} \ddot{q}_i(t) + k_{ii} q_i(t) = 0 \]  \hspace{1cm} (4.24)

Or

\[ \ddot{q}_i(t) + \omega_i^2 q_i(t) = 0 \]  \hspace{1cm} (4.25)

where

\[ \omega_i = \sqrt{\frac{k_{ii}}{m_{ii}}} \]  \hspace{1cm} (4.26)

is the \( i \)th natural frequency of the beam.

Now

\[ L(u_i(x,t)) = \alpha_4 \frac{\partial^4}{\partial x^4} (\phi_i(x) q_i(t)) = q_i(t) \alpha_4 \frac{\partial^4 \phi_i(x)}{\partial x^4} \]

\[ = q_i(t) L(\phi_i(x)) \]

and

\[ M(\dddot{u}_i(x,t)) = \beta_0 \frac{d^2}{dt^2} (\phi_i(x) q_i(t)) = \dddot{q}_i(t) \beta_0 \phi_i(x) \]

\[ = \dddot{q}_i(t) M(\phi_i(x)) \]
but

\[ M(\dddot{u}_i(x,t)) + L(\dot{u}_i(x,t)) = 0 \]

combining the three equations above gives

\[ \dddot{q}_i(t)M(\phi_i(x)) + \dot{q}_i(t)L(\phi_i(x)) = 0 \Rightarrow \]

\[ \frac{L(\phi_i(x))}{M(\phi_i(x))} = -\frac{\dddot{q}_i(t)}{\dot{q}_i(t)} = \omega_i^2 \Rightarrow \]

\[ EI \frac{\partial^4 \phi_i(x)}{\partial x^4} - \omega_i^2 \rho \phi_i(x) = 0 \Rightarrow \]

which finally gives

\[ \frac{\partial^4 \phi_i(x)}{\partial x^4} - b_1^4 \phi_i(x) = 0 \quad (4.27) \]

where

\[ b_1^4 = \frac{\omega_i^2 \rho}{EI} \quad (4.28) \]

The solution of the differential equation (4.27) is:

\[ \phi_i(x) = C_{i1} \sin b_1 x + C_{i2} \cos b_1 x + C_{i3} \sinh b_1 x + C_{i4} \cosh b_1 x \]

\[ (4.29) \]
where $C_{i1}$, $C_{i2}$, $C_{i3}$ and $C_{i4}$ are constants which can be calculated using the boundary conditions of the beam. In particular from equations (7), (8), (9) and (10) assuming $q_i(t)$ and its derivatives are not identically equal to 0, we have from equation (4.7):

$$\phi_i(0) = 0 \quad \Rightarrow$$

$$C_{i2} + C_{i4} = 0,$$

Or

$$C_{i2} = -C_{i4} \quad (4.30)$$

From equation (4.8)

$$\frac{\partial}{\partial x} (\phi_i(x) q_i(t)) = q_i(t) \frac{\partial}{\partial x} (\phi_i(x)) \bigg|_{x=0} = 0 \quad \Rightarrow$$

$$b_i \left( C_{i1} \cos b_i x - C_{i2} \sin b_i x + C_{i3} \cosh b_i x + C_{i4} \sinh b_i x \right) \bigg|_{x=0} = 0$$

Or

$$b_i \left( C_{i1} + C_{i3} \right) = 0 \quad \Rightarrow$$

$$C_{i1} + C_{i3} = 0$$
and finally

\[ C_{11} = - C_{13} \]  \hspace{1cm} (4.31)

From equation (4.9)

\[ \frac{\partial^2}{\partial x^2} (u(x,t)) = q_1(t) \frac{\partial^2}{\partial x^2} (\phi_1(x)) \bigg|_{x=L} = 0 \Rightarrow \]

\[ b_1^2 \left( -C_{11} \sin b_1 x - C_{12} \cos b_1 x + C_{13} \sinh b_1 x + C_{14} \cosh b_1 x \right) \bigg|_{x=L} = 0 \]

and because of (4.30) and (4.31)

\[ C_{11} \left( \sin b_1 L + \sinh b_1 L \right) + C_{12} \left( \cos b_1 L + \cosh b_1 L \right) = 0 \]  \hspace{1cm} (4.32)

Finally equation (4.10) gives

\[ \frac{\partial^3}{\partial x^3} (u(x,t)) \bigg|_{x=L} = q_1(t) \frac{\partial^3}{\partial x^3} (\phi_1(x)) \bigg|_{x=L} = 0 \Rightarrow \]

\[ b_1^3 \left( -C_{11} \cos b_1 x + C_{12} \sin b_1 x + C_{13} \cosh b_1 x + C_{14} \sinh b_1 x \right) \bigg|_{x=L} = 0 \]

and again because of (4.30) and (4.31)

\[ C_{11} \left( \cos b_1 L + \cosh b_1 L \right) - C_{12} \left( \sin b_1 L - \sinh b_1 L \right) = 0 \]  \hspace{1cm} (4.33)

Equations (4.32) and (4.33) form a system with two equations and two unknowns, namely \( C_{11} \) and \( C_{12} \). For this system to have a solution the determinant of the coefficients of \( C_{11} \)
and $C_{12}$ must equal zero.

Therefore,

$$
\begin{vmatrix}
\sin b_1 L + \sinh b_1 L & \cos b_1 L + \cosh b_1 L \\
\cos b_1 L + \cosh b_1 L & \sinh b_1 L - \sin b_1 L
\end{vmatrix} = 0
$$

Or

$$(\cos b_1 L + \cosh b_1 L)^2 - \sinh^2 b_1 L + \sin^2 b_1 L = 0$$

Or

$$(\cosh^2 b_1 L - \sinh^2 b_1 L) + (\cos^2 b_1 L + \sin^2 b_1 L) + 2\cos b_1 L \cosh b_1 L$$

$$= 1 + 1 + 2\cos b_1 L \cosh b_1 L = 0$$

Which finally gives the frequency equation for the cantilever beam:

$$\cos b_1 L \cosh b_1 L = -1 \quad (4.34)$$

Solving equation (4.34) for $b_1$ facilitates the calculation of the natural frequencies of the beam given in equation (4.28).

Solving the system of equations (4.30), (4.31), (4.32) and (4.33) for $C_{11}$, $C_{12}$, $C_{13}$ and $C_{14}$ and substituting into equation (4.29) gives

$$\phi_1(x) = A_1 (\cos b_1 x - \cosh b_1 x) + \sin b_1 x - \sinh b_1 x \quad (4.35)$$
where
\[
A_i = - \frac{\sin b_i L + \sinh b_i L}{\cos b_i L + \cosh b_i L} = \frac{\cos b_i L + \cosh b_i L}{\sin b_i L - \sinh b_i L}
\] (4.36)

From equations (4.35) and (4.36) the shape of the vibration, \( \phi_i(x) \), for each mode of the beam is determined. Since the natural frequencies of the beam can also be calculated from equations (4.28) and (4.34) it is only the magnitude functions \( q_i(t) \) which need be determined in order to obtain the displacement of the beam as given in (4.11). The functions \( q_i(t) \) can be calculated by solving the differential equation (4.25).

4.3 DEVELOPMENT OF THE STATE-SPACE MODEL OF THE BEAM-VIBRATOR SUBSYSTEM.

It has been seen in the previous chapter that the mechanical behaviour of an electrodynamic vibrator can be modelled by a simple mass-spring-damper system. The damping action of the mechanical subsystem of the vibrator is negligible when compared to the damping provided by the back e.m.f. and as such can be ignored. We can therefore model the effect of the mechanical behaviour of the vibrators on the beam by mass-spring suspensions attached to the beam. In figure 4.2, the schematic diagram of the existing rig is shown with the two electrodynamic vibrators modelled as mass and spring suspensions.

If the forces \( f_i, i=1,2,\ldots,1 \), produced by the vibrators cause displacement \( u(x_p,t) \) at distance, \( x_p \), along the beam where the pth vibrator is attached, the suspension due to this vibrator will produce a force that tries to oppose the
movement of the beam. This negative force can be further broken down to an inertial force due to the mass \( m_{pv} \) and a stiffness force due to the vibrator stiffness \( k_p \).

The negative force acting on the beam due to the \( p \)th vibrator stiffness is

\[
\tilde{f}_p(t) = -k_p u_p(x_p, t) = -k_p \sum_{i-1}^{\infty} \phi_i(x_p) q_i(t) \tag{4.37}
\]

and the negative inertial force acting on the beam due to the mass of the armature of the \( p \)th vibrator is

\[
f_{mp}(t) = -m_p \ddot{u}_p(x_p, t) = -m_p \sum_{i-1}^{\infty} \phi_i(x_p) \ddot{q}_i(t) \tag{4.38}
\]

The total force acting on the beam due to the \( p \)th vibrator will be

\[
f_{tp} = f_p + f_{mp} + f_{kp} \tag{4.39}
\]

Now when the beam is forced into vibration by an external force \( F(x,t) \) distributed along the beam equation (4.12) becomes

\[
M(u) + L(u) = F(x,t) \tag{4.40}
\]

and distributing \( F(x,t) \) to each mode

\[
F_i(x,t) = \int_0^L F(x,t) \phi_i(x) \, dx \tag{4.41}
\]
In this case however the forces that act on the beam are concentrated at the points where the vibrators are attached. To distribute these forces spatially along the beam we can use the Dirac delta function to define the total external force seen at a point $x$ on the beam:

$$F(x,t) = f_{t_1}(t)\delta(x-x_1) + \ldots + f_{t_n}(t)\delta(x-x_n)$$

or

$$F_i(x,t) = \sum_{p=1}^{n} \int_0^L f_{tp}(t)\phi_i(x)\delta(x-x_p)\,dx = \sum_{p=1}^{n} \phi_i(x_p)f_{tp}(t) \quad (4.43)$$

Substituting (4.42) into (4.41) we obtain

$$F_i(x,t) = \sum_{p=1}^{n} \int_0^L f_{tp}(t)\phi_i(x)\delta(x-x_p)\,dx = \sum_{p=1}^{n} \phi_i(x_p)f_{tp}(t) \quad (4.43)$$

Taking into account equation (4.39) we can express the above relationship as

$$F_i(x,t) = \sum_{p=1}^{n} \left( f_p(t) + f_{mp}(t) + f_{kp}(t) \right)\phi_i(x_p)$$

or

$$F_i(x,t) = f^i(t) + f^i_{m}(t) + f^i_{k}(t) \quad (4.44)$$
Where

\[ f^1(t) = \sum_{p=1}^{l} f_p(t) \phi_i(x_p) \]  
(4.45)

and

\[ \sum_{p=1}^{l} f(k) \phi_i(x_p) \]  
(4.46)

Now we have

\[
\begin{bmatrix}
    f^1(t) \\
    \vdots \\
    f^n(t)
\end{bmatrix} =
\begin{bmatrix}
    \phi_1(x_1) & \phi_1(x_2) & \cdots & \phi_1(x_n) \\
    \vdots & \vdots & \cdots & \vdots \\
    \phi_n(x_1) & \phi_n(x_2) & \cdots & \phi_n(x_n)
\end{bmatrix}
\begin{bmatrix}
    f_1(t) \\
    \vdots \\
    f_n(t)
\end{bmatrix}
\]

if we put

\[
f =
\begin{bmatrix}
    f^1(t) \\
    \vdots \\
    f^n(t)
\end{bmatrix}
\quad \text{and} \quad
f =
\begin{bmatrix}
    f_1(t) \\
    \vdots \\
    f_n(t)
\end{bmatrix}
\]

(4.47)
Then if

\[ \Phi_F = \begin{bmatrix}
\phi_1(x_1) & \phi_1(x_2) & \ldots & \phi_1(x_1) \\
\vdots & \vdots & \ddots & \vdots \\
\phi_n(x_1) & \phi_n(x_2) & \ldots & \phi_n(x_1)
\end{bmatrix} \]

we have

\[ f = \Phi_F \mathbf{f} \quad (4.49) \]

Equation (4.49) gives the force for each normal mode of the beam due to the combined effect of all the forces produced by the vibrators. We need to calculate equivalent relationships for the forces due to the combined effects of all the inertial and stiffness forces (or constraint forces) generated by the suspensions of figure 4.2, which model the behaviour of the mechanical subsystems of the vibrators.

First the force due the spring action of the suspensions will be calculated. From equation (4.37) and (4.46) we have

\[ f_k^i(t) = \sum_{p=1}^N f_{k_p}(t) \phi_i(x_p) = -\sum_{p=1}^N k_p \phi_i(x_p) u(x_p,t) \]
or

\[
\dot{x}_k(t) = - \begin{bmatrix} k_1 \phi_1(x_1) & k_2 \phi_1(x_2) & \ldots & k_1 \phi_1(x_1) \end{bmatrix} \begin{bmatrix} u(x_1, t) \\ u(x_2, t) \\ \vdots \\ u(x_1, t) \end{bmatrix}
\]

and since

\[
\begin{bmatrix} u(x_1, t) \\ \vdots \\ u(x_1, t) \end{bmatrix} = \begin{bmatrix} \phi_1(x_1) & \phi_2(x_1) & \ldots & \phi_n(x_1) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_1(x_1) & \phi_2(x_1) & \ldots & \phi_n(x_1) \end{bmatrix} \begin{bmatrix} q_1(t) \\ \vdots \\ q_n(t) \end{bmatrix} = \Phi_F^T q
\]

where

\[
q = \begin{bmatrix} q_1(t) \\ \vdots \\ q_n(t) \end{bmatrix}
\]

(4.50)
Therefore if we put

\[
f_k = \begin{bmatrix} f_k^1(t) \\ \vdots \\ f_k^l(t) \end{bmatrix} \quad \text{and} \quad K_s = \begin{bmatrix} k_1 & 0 & \cdots & 0 \\ 0 & k_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & k_1 \end{bmatrix}
\]

we have, for the distributed to the beam's natural modes, spring forces of the suspension

\[
f_k = -\Phi_f K_s \Phi_f^T q \tag{4.51}
\]

Similarly putting

\[
f_m = \begin{bmatrix} f_m^1(t) \\ \vdots \\ f_m^l(t) \end{bmatrix} \quad \text{and} \quad M_g = \begin{bmatrix} m_1 & 0 & \cdots & 0 \\ 0 & m_2 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & m_1 \end{bmatrix}
\]

and combining with equations (4.38) and (4.47) we obtain

\[
f_k = -\Phi_f M_s \Phi_f^T q \tag{4.52}
\]

Having established relationships for all the component
forces which cause the vibration of the beam in terms of \( q_1(t) \) and \( \phi_1(x) \), we can now develop the state-space model for the beam-vibrator subsystem. Equation (4.40) can now be written as

\[
\ddot{q}(t) + \Omega q(t) = M^{-1} \left[ f(t) + f_k(t) + f_m(t) \right]
\]  

(4.53)

where \( \Omega \) is the matrix of the squared natural frequencies of the beam.

\[
\Omega = \begin{bmatrix}
\omega_1^2 & 0 & \ldots & 0 \\
0 & \omega_2^2 & \ldots & 0 \\
0 & 0 & \ddots & \ddots \\
0 & 0 & \ldots & \omega_n^2
\end{bmatrix}
\]  

(4.54)

and \( M \) is the diagonal matrix of the generalised masses

\[
M = \begin{bmatrix}
m_{11} & 0 & \ldots & 0 \\
0 & m_{22} & \ldots & 0 \\
0 & 0 & \ddots & \ddots \\
0 & 0 & \ldots & m_{nn}
\end{bmatrix}
\]  

(4.55)
Substituting (4.49), (4.51) and (4.52) into equation (4.53) we have

$$\ddot{q}(t) + \Omega q(t) = M^{-1}\Phi_f f(t) - M^{-1}\Phi_f K_s \Phi_f^T q(t) - M^{-1}\Phi_f M_s \Phi_f^T q(t)$$

or

$$\ddot{q}(t) = -\left[I + M^{-1}\Phi_f M_s \Phi_f^T\right]^{-1}\left[I + M^{-1}\Phi_f K_s \Phi_f^T\right] q(t)$$

$$+ \left[I + M^{-1}\Phi_f M_s \Phi_f^T\right]^{-1} M^{-1}\Phi_f f(t)$$

(4.56)

Or finally

$$\ddot{q}(t) = -P_q q(t) + P_f f(t)$$

(4.57)

where

$$P_q = \left[I + M^{-1}\Phi_f M_s \Phi_f^T\right]^{-1}\left[I + M^{-1}\Phi_f K_s \Phi_f^T\right]$$

(4.58)

and

$$P_f = \left[I + M^{-1}\Phi_f M_s \Phi_f^T\right]^{-1}\Phi_f$$

(4.59)

The state-space model of the beam-vibrator subsystem can easily be obtained by letting

$$q_1(t) = q(t) \quad \text{and} \quad q_2(t) = \dot{q}(t)$$

-4.24-
Defining the state vector

\[ q_{ss}(t) = \begin{bmatrix} q_1(t) \\ q_2(t) \end{bmatrix} = \begin{bmatrix} q(t) \\ \dot{q}(t) \end{bmatrix} \]  \hspace{1cm} (4.60)

we obtain from equation (4.57)

\[ \dot{q}_{ss}(t) = \begin{bmatrix} 0 & I \\ -p & 0 \end{bmatrix} q_{ss}(t) + \begin{bmatrix} 0 \\ p_f \end{bmatrix} f(t) \]  \hspace{1cm} (4.61)

Recalling that

\[ u(x,t) = \sum_{i=1}^{n} \phi_i(x) q_i(t) \]

Define the matrix

\[ \Phi_u = \begin{bmatrix} \phi_1(x_{11}) & \phi_2(x_{11}) & \ldots & \phi_n(x_{11}) \\ \vdots & \vdots & \ddots & \vdots \\ \phi_1(x_{1n}) & \phi_2(x_{1n}) & \ldots & \phi_n(x_{1n}) \end{bmatrix} \]  \hspace{1cm} (4.62)

where \( x_{1i} \) is the distance of the \( i \)th LVDT from the clamped end of the beam.
The output displacement vector is then given by

\[
\mathbf{u}(t) = \begin{bmatrix}
\mathbf{u}(x_{11}, t) \\
\vdots \\
\mathbf{u}(x_{1m}, t)
\end{bmatrix} = \begin{bmatrix}
\Phi_u & 0_{n \times m}
\end{bmatrix} \mathbf{q}_{ss}(t)
\]  \hspace{1cm} (4.63)

Equations (4.61) and (4.63) can be rewritten in the standard state-space form

\[
\dot{\mathbf{q}}_{ss}(t) = \mathbf{A}_{bv} \mathbf{q}_{ss}(t) + \mathbf{B}_{bv} \mathbf{f}(t)
\]  \hspace{1cm} (4.64)

\[
\mathbf{u}(t) = \mathbf{C}_{bv} \mathbf{q}_{ss}(t) + \mathbf{D}_{bv} \mathbf{f}(t)
\]

where

\[
\mathbf{A}_{bv} = \begin{bmatrix}
0_{n \times n} & \mathbf{I}_n \\
-p_{q} & 0_{n \times n}
\end{bmatrix}
\]  \hspace{1cm} (4.65)

\[
\mathbf{B}_{bv} = \begin{bmatrix}
0_{n \times 1} \\
p_{f}
\end{bmatrix}
\]  \hspace{1cm} (4.66)

\[
\mathbf{C}_{bv} = \begin{bmatrix}
\Phi_u & 0_{n \times m}
\end{bmatrix}
\]  \hspace{1cm} (4.67)

and finally

\[
\mathbf{D}_{bv} = 0_{n \times 1}
\]  \hspace{1cm} (4.68)
If the output is to be velocity instead of displacement then since

\[ v(x,t) = \dot{u}(x,t) = \sum_{i=1}^{n} \phi_i(x)q_i(t) \]

equation (4.64) becomes

\[ \dot{q}_{ss}(t) = A_{bv}q_{ss}(t) + B_{bv}f(t) \]

\[ v(t) = C_{v,bv}q_{ss}(t) + D_{bv}f(t) \]  \hspace{1cm} (4.69)

where \( v(t) \) is the vector of velocities at the measurement points \( x_{1i} \). \( A_{bv}, B_{bv} \) and \( D_{bv} \) remain the same but \( C_{bv} \) now is given by

\[ C_{v,bv} = \begin{bmatrix} 0_{nxm} & \Phi_u \end{bmatrix} \]  \hspace{1cm} (4.70)

### 4.4 INTRODUCING DAMPING

The model of the beam-vibrator subsystem developed above is based on the assumption that the vibration of the beam is undamped. This can be justified by the fact that the damping produced by the vibrators' back e.m.f. is much greater than the damping of the beam alone which is therefore usually ignored. There are however cases where the effect of the damping of the beam could be significant. Damping in a vibrating beam can be caused by a number of factors such as internal molecular friction or sliding friction. Generally the mathematical description of these forms of friction is quite complicated and not suitable for vibration analysis. Thus only
the simplified viscous damping will be used here to model the damping forces acting on the vibrating beam. The damping need also be modelled in such a way as not to alter the natural frequencies of the beam. This can be achieved by introducing a damping factor $\zeta_i$ for each mode of the beam. Equation (4.25) will now be written as

$$\ddot{q}_i(t) + 2\zeta_i\omega_i\dot{q}_i(t) + \omega_i^2q_i(t) = 0$$  \hspace{1cm} (4.71)$$

and equation (4.53) becomes

$$\ddot{\mathbf{q}}(t) + 2Z\Omega^{1/2}\dot{\mathbf{q}}(t) + \Omega\mathbf{q}(t) = M^{-1}[f(t) + f_k(t) + f_m(t)]$$  \hspace{1cm} (4.72)$$

where

$$Z = \begin{bmatrix} \zeta_1 & 0 & \ldots & 0 \\ 0 & \zeta_2 & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & \zeta_n \end{bmatrix}$$  \hspace{1cm} (4.73)$$

Equation (4.57) also changes to

$$\ddot{\mathbf{q}}(t) = -P_d\dot{\mathbf{q}}(t) - P_q\mathbf{q}(t) + P_f\mathbf{f}(t)$$  \hspace{1cm} (4.74)$$
where

\[ P_d = 2 \left[ I + M^{-1} \Phi_F M_s \Phi_F^T \right]^{-1} Z \Omega^{1/2} \] (4.75)

Developing the state-space model for the beam-vibrator subsystem with viscous damping from equation (4.74) involves the same process as before. In fact the state-space model matrices \( B_{bv} \), \( C_{bv} \) and \( D_{bv} \) remain unchanged and only \( A_{bv} \) is altered to

\[ A_{bv,d} = \begin{bmatrix} O_{n \times n} & I_n \\ -P_q & -P_d \end{bmatrix} \] (4.76)

4.5 ALGORITHM FOR CALCULATION OF THE STATE-SPACE MODEL.

The mathematical model of the beam-vibrator subsystem, as this was developed in the preceding sections, need to be summarised in an algorithm which could be programmed in a digital computer to calculate the subsystem's state-space model. Once this is done the calculation of the state-space model or transfer function matrix of the entire rig, can be accomplished relatively easily from the block diagram of figure 4.3. The outputs of the rig can be chosen to be one of the following variables: displacement, velocity, load or current.

The algorithm for the calculation of the state-space of the beam-vibrator subsystem is given below.
Algorithm 1

Step 1:

Use the equation

$$\cos b_i L \cosh b_i L = -1$$

to form a look-up table of values of

$$\lambda_i = b_i L$$

and subsequently of

$$b_i = \frac{b_i L}{L}$$

for \( i = 1, 2, \ldots, n_{\text{max}} \) where \( n_{\text{max}} \) is the maximum number of modes that can be considered.

Step 2:

Use the equation

$$A_i = \frac{\sin b_i L + \sinh b_i L}{\cos b_i L + \cosh b_i L} = \frac{\cos b_i L + \cosh b_i L}{\sin b_i L - \sinh b_i L}$$

to form a look-up table of values of \( A_i \) for \( i = 1, 2, \ldots, n_{\text{max}} \)

Step 3:

Specify the number of modes to be considered \( n \), the mass \( m_b \), the length \( L \) and the flexural stiffness \( E I \) of the beam. Specify also for each of the vibrators, the armature mass, resistance, stiffness, force and back e.m.f. as well as the distance from the clamped end
of the beam. Finally specify the positions of the LVDTs relative to the clamped end of the beam.

Step 4:

Assuming a uniform beam calculate its mass per length

\[ m = \frac{m_b}{L} \]

Then calculate the generalised masses of the first \( n \) modes by evaluating the integral:

\[ m_{ii} = \frac{m}{L} \int_0^L \phi_i^2(x) \, dx \]

where

\[ \phi_i(x) = A_i (\cos b_i x - \cosh b_i x) + \sin b_i x - \sinh b_i x \]

The generalised mass for each mode can be calculated using numerical integration, or by evaluating the integral analytically. This last approach gives:

\[ m_{ii} = \frac{m}{b_i} \left\{ A_i^2 \left[ \lambda_i + \frac{1}{4} \sin 2\lambda_i + \frac{1}{2} \cosh \lambda_i \sinh \lambda_i - \cosh \lambda_i \sin \lambda_i \\
- \sinh \lambda_i \cos \lambda_i \right] + A_i \left( \sinh \lambda_i - \sin \lambda_i \right)^2 + \sinh \lambda_i \cos \lambda_i \\
- \cosh \lambda_i \sin \lambda_i - \frac{1}{4} \sin 2\lambda_i + \frac{1}{2} \cosh \lambda_i \sinh \lambda_i \right\} \]
Step 5:

Calculate the natural frequency of the first \( n \) modes, from the equation

\[
\omega_i^2 = \frac{b_i^4 EI}{m}
\]

and subsequently form the matrix

\[
\Omega = \begin{bmatrix}
\omega_1^2 & 0 & \cdots & 0 \\
0 & \omega_2^2 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & \omega_n^2
\end{bmatrix}
\]

Step 6:

Use the equation for \( \phi_1(x) \) in step 4 to calculate the matrices.

\[
\Phi_F = \begin{bmatrix}
\phi_1(x_1) & \phi_1(x_2) & \cdots & \phi_1(x_1) \\
\vdots & \vdots & \ddots & \vdots \\
\phi_n(x_1) & \phi_n(x_2) & \cdots & \phi_n(x_1)
\end{bmatrix}
\]

where \( x_i \) is the distance of the \( i \)th vibrator from the clamped end of the cantilever beam.
Also form the matrix
\[
\Phi_u = \begin{bmatrix}
\phi_1(x_{11}) & \phi_2(x_{11}) & \cdots & \phi_n(x_{11}) \\
\vdots & \vdots & \ddots & \vdots \\
\vdots & \vdots & \ddots & \vdots \\
\phi_1(x_{1m}) & \phi_2(x_{1m}) & \cdots & \phi_n(x_{1m})
\end{bmatrix}
\]

where \(x_{1i}\) is the distance of the \(i\)th LVDT from the clamped end of the cantilever beam.

Step 7:

Specify the damping ratio \(\zeta\) for each of the \(n\) modes considered and form the matrix
\[
\zeta = \begin{bmatrix}
\zeta_1 & 0 & \cdots & 0 \\
0 & \zeta_2 & \cdots & 0 \\
\vdots & \vdots & \ddots & \vdots \\
0 & 0 & \cdots & \zeta_n
\end{bmatrix}
\]

Step 8:

Finally form the matrices:
\[
P_q = \left[ I + M^{-1} \Phi_p M^T \right]^{-1} \left[ \Omega + M^{-1} \Phi_p K \Phi_p^T \right]
\]

-4.33-
and

\[ P_f = \left(I + M^{-1} \Phi F M_g \Phi F^T\right)^{-1} \Phi F \]

and

\[ P_d = 2 \left(I + M^{-1} \Phi F M_g \Phi F^T\right)^{-1} 2 \Omega^{1/2} \]

and use them to form the state-space matrices

\[ A_{bv,d} = \begin{bmatrix} O_{n \times n} & I_n \\ -P_q & -P_d \end{bmatrix} \]

\[ B_{bv} = \begin{bmatrix} O_{n \times 1} \\ P_f \end{bmatrix} \]

\[ C_{bv} = \begin{bmatrix} \Phi_u & O_{n \times m} \end{bmatrix} \]

if the required output is displacement, or

\[ C_{v, bv} = \begin{bmatrix} O_{n \times m} & \Phi_u \end{bmatrix} \]

if the required output is velocity. Finally

\[ D_{bv} = O_{m \times 1} \]

-4.34-
A MATLAB function to implement the above algorithm has been developed and the relevant listing is given in appendix 1.

The state-space model of the entire rig can be calculated from the block diagram of figure 4.3. This is accomplished using the Multivariable Frequency Domain toolbox for use with MATLAB. A MATLAB program whose listing is also given in appendix 1, has been written to automate this calculation. The same program provides simulation of the time response of displacement, velocity, load or current for any given combination of inputs in open or closed-loop configurations.

4.6 EVALUATION OF CONTINUOUS ELASTIC BODY MODEL

The main motivation for attempting this type of mathematical modelling of the rig, was to develop a flexible modelling process which would produce accurate and relatively simple models. Flexible here is used in the sense that the user should be able to obtain models with little effort for different configurations of beams and vibrators.

The continuous body approach to modelling does indeed lend itself to such treatment. The relatively short MATLAB program mentioned above, only requires from the user to specify the parameters of the beam and the vibrators and the locations of the vibrators and the LVDT's, in order to calculate the state-space model of the rig. Subsequent simulation of a range of output variables is also provided for any given input time series.

Moreover the order of the continuous model is only governed by the number of modes that are considered and is therefore independent of the number of vibrators and LVDTs and their relative locations. As an example considered the case of two different rig configurations, one with two vibrators and two LVDTs and the other with three vibrators and three LVDTs. In
both cases the LVDTs are attached to different points on the beam from those where the vibrators are attached. A finite element analysis model would need to consider at least 4 nodes for the first configuration and 6 for the second. This would produce models of at least 8th and 12th order for the first and second configuration respectively. A lumped parameter model would be of the same order since again at least 4 lumped masses for the first configuration and 6 for the second should be considered. Note here that in both cases these are the least orders of the models, which may not provide the necessary accuracy, and more nodes or lumped masses may have to considered. We see therefore that the order of the produced models depends not only on the number of system input and outputs but also on their location, relative to each other, as well as the required accuracy.

The continuous body model on the other hand is frequency orientated and its order depends solely upon the number of modes that will be considered to give the required accuracy in the frequency range of interest. Of course as the number of vibrators increases the structure will be stiffened and more modes may need be considered for accurate modelling. The number and positions of LVDTs however does not affect the complexity of the model.

Finally the easy addition of damping in each mode, provides for more realistic simulation. This is possible because the dynamics of the system at each mode are described by a second order differential equation whose damped frequency response is so well defined in classical control theory.

Frequency responses produced by the continuous system model with three modes considered, are plotted together with experimental frequency responses in the graphs of figure 4.4. Although the accuracy of the model compares well with lumped parameter or finite element models, it does not provide a perfect match to the experimental frequency response. The discrepancy between actual and simulated frequency responses

-4.36-
can be attributed to the following factors.

(a) The values of the Young's modulus of the beam used is the nominal one obtained from standard tables. Also nominal values of the vibrator parameters (armature resistance, and force constant) are used.

(b) The beam is assumed to be absolutely uniform having constant flexural stiffness $E I$ and mass per length $m(x)=m$.

(c) Heat losses in the vibrator armature coil are ignored and instead direct transfer of electrical to mechanical power is assumed giving $k_B = k_F$. The mechanical damping of the vibrators are also ignored.
Figure 4.1 Free body diagram of an element \( dx \) of the cantilever beam.

Figure 4.2 Schematic diagram of rig with the vibrators modelled as mass and spring suspensions.
Figure 4.3 Block diagram of the entire rig.
Figure 4.4. Magnitude Bode array of experimental frequency responses and frequency responses produced by a continuous system model with the first three modes considered.
5.1 INTRODUCTION

In chapter 2 it was shown that a multivariable feedback system approach is needed in the design of control systems for fatigue/structural tests. It was also shown that the desired loads and therefore the input reference signals, are described in terms of their frequency composition. Such description is the obvious choice for sinusoidally based signals. Stochastic type signals on the other hand are described by their power spectral density function which is again a frequency domain function. Furthermore specific requirements were developed in chapter 2 for the control system in terms of the shape of its frequency response, in order to achieve accurate reproduction of the results. It is clear, in the light of what is said above, that a multivariable frequency domain method should be used to design the required control system.

In this chapter the two main multivariable frequency response methods, the Nyquist array methods [13-41] and the characteristic locus method [42-44], together with recent extensions, will be described. Also ways of assessing the robustness and the performance of the design of the multivariable feedback system will be given. Before this however it is useful to summarise the main input-output relationships and the requirements of the multivariable
control system as they were developed in chapter 2.

Refer to figure 5.1. The input-output relationship is

\[ y(s) = S(s) \cdot d(s) + T(s) \cdot P(s) \cdot r(s) - T(s) \cdot m(s) \]  

(5.1)

where \( S(s) \) is the sensitivity function which is defined as

\[ S(s) = \left[ I + G(s) \cdot K(s) \right]^{-1} \]  

(5.2)

and \( T(s) \) is the loop transfer function matrix which is given by

\[ T(s) = S(s) \cdot G(s) \cdot K(s) = \left[ I + G(s) \cdot K(s) \right]^{-1} G(s) \cdot K(s) \]  

(5.3)

Now

\[ S^{-1}(s) = I + G(s) \cdot K(s) \Rightarrow G(s) \cdot K(s) = S^{-1}(s) - I \]

and

\[ T(s) = S(s) \cdot G(s) \cdot K(s) = S(s) \left[ S^{-1}(s) - I \right] \Rightarrow \]

\[ T(s) = I - S(s) \]  

(5.4)

Because of equation (5.4) the loop transfer function matrix is sometimes referred to as the complementary sensitivity function.

The requirements for a control system that will be used in a fatigue or dynamic structural test can be summarised in the following

(a) Stability under all possible operating conditions. This requirement will usually include stability
under all perturbations up to a certain magnitude. It can also mean that the integrity of the system in the case of components failure is also required.

(b) Accurate reproduction of the reference signal at the output. As it is shown in chapter 2 this requirements can be translated into elimination of interaction and flat frequency responses of the transfer function matrix $T(s)P(s)$, with approximately unity gain in the desired bandwidth.

(c) To guarantee the above requirement for accuracy, the system should have adequate disturbance and noise rejection properties. From equation (5.1) we see that this translates to a "small" sensitivity function and at the same time a "small" loop transfer function. These requirements however according to equation (5.4) are conflicting. In practice the conflict is resolved by the fact that $d(s)$ and $m(s)$ are relatively large at different frequency ranges and therefore $S(s)$ and $T(s)$ need not be small at the same time.

The above requirements for the design of the control system for dynamic structural tests, also provide the criteria against which a design method must be evaluated. The description and the evaluation of both the Nyquist array methods and the characteristic locus, follow in this chapter. Although chronologically the Nyquist array methods where developed earlier, the characteristic locus method is described here first. The reason for this is that certain concepts which are defined in the characteristic locus method give considerable insight when studying the inverse or the direct Nyquist array technique.

Ways of quantifying the robustness of the performance of each design are also given towards the end of this chapter.
5.2 THE CHARACTERISTIC LOCUS METHOD.

Consider the closed loop system of figure 5.2. Here the effects of the possible disturbance \( d(s) \) and the measurement noise \( m(s) \) are initially ignored. In the characteristic locus method the controller \( K(s) \) is used to manipulate the eigenvalues of the forward path transfer function matrix \( Q(s) = G(s)K(s) \) and in this way achieve the desired performance.

The eigenvalues or the characteristic gain functions, of a transfer function matrix \( G(s) \) are defined as the solutions \( g(s) \) of the equation

\[
|g(s).I - G(s)| = 0
\]  \( (5.5) \)

Note that the characteristic functions \( g_i(s) \) are not in general rational. They may be considered as defined on a Riemann surface consisting of a number of copies of the complex plane appropriately joined together.

The eigenvectors \( w_i(s) \) of \( G(s) \) are called its characteristic directions. Define then the matrices \( W(s) \) and \( V(s) \) such that

\[
W(s) = [w_1(s) \ w_2(s) \ ... \ w_m(s)]
\]  \( (5.6) \)

and

\[
V(s) = W^{-1}(s)
\]  \( (5.7) \)

then the transfer function matrix \( G(s) \) is given by the relationship

\[
G(s) = W(s). \text{diag}\{g_1(s)\}.V(s)
\]  \( (5.8) \)
The terms characteristic gain functions and characteristic directions can be justified if you consider the system output in response to an input vector equal to $w_j(s)$

$$y(s) = g_j(s) . w_j(s) = G(s) . w_j(s)$$  \hfill (5.9)

Since $g_j(s)$ and $w_j(s)$ form an eigenpair of $G(s)$. Equation (5.9) shows that if the system input has the same direction with an eigenvector (or characteristic direction) of $G(s)$, then the system output will have the same direction as the input, but would have been scaled by a scalar equal to the corresponding eigenvalue (or characteristic gain).

The characteristic gain functions can be used to assess the closed-loop stability of the system. They also give an important insight into the system dynamics which was used by McFarlane and Kouvaritakis to form the "Characteristic Locus" design method. These properties of the characteristic gains and the characteristic locus design procedure are discussed in the next two subsections.

### 5.2.1 Closed-loop stability assessment using characteristic gains.

A multivariable system is said to be stable if, as in the SISO case, all its poles lie in the open left-half of the complex plane. In the SISO case closed loop stability is assessed from the Nyquist diagram of the open-loop transfer function. The characteristic gains provide the facilities for generalising the Nyquist criterion to the multivariable case.

First rewrite equation (5.5)

$$| g(s) . I - G(s) | = \Delta_1(g,s) . \Delta_2(g,s) \ldots \Delta_k(g,s) = 0$$  \hfill (5.10)
where each factor $\Delta_i(g,s)$ is irreducible over the field of rational functions. The factors $\Delta_i(g,s)$ are polynomials in $g(s)$ with coefficients which are rational functions in $s$. The set of the characteristic gains will then be the set of the solutions of the equations $\Delta_i(g,s) = 0$. Each characteristic gain function is an algebraic function. This is a generalisation of the simple function of a complex variable, and has as its domain a Riemann surface on which a single-valued complex function is defined. A set of poles and zeros can then be associated with this single-valued complex function. If the polynomial $\Delta_i(g,s)$ is of degree $p$ then the Riemann surface will consist of $p$ copies of the complex plane.

The generalised Nyquist criterion can now be stated as follows:

**Theorem**: The $m$-input, $m$-output linear system of figure 5.2, will be stable if and only if each characteristic locus associated with an irreducible factor $\Delta_i(q,s)$ of $\det\{q(s).I - Q(s)\}$, encircles the critical point $(-1+j0)$, $P_{Oi}$ times, where $P_{Oi}$ is the number of right half plane poles of the irreducible factor $\Delta_i(q,s)$.

A physical interpretation of that can be given if it is noticed that the $\det\{q(s).I - Q(s)\}$ has the same common denominator with $Q(s)$. Therefore the set of the poles of all $\Delta_i(q,s)$ is the set of the poles of $Q(s)$.

In practical situations therefore the closed-loop system's stability can be assessed by using C.A.D. to obtain the $m$ Nyquist diagrams that correspond to the $m$ eigenvalues $q_i(s)$ of $Q(s)$. These Nyquist diagrams that are obtained from the
characteristic gain functions of the transfer function matrix are called the system's characteristic loci. Closed loop-stability is obtained if, and only if, the sum of the encirclements of the critical point (-1+j0) by the characteristic loci is $p_0$, where $p_0$ is the number of unstable poles of the open-loop transfer function matrix $Q(s)$.

The integrity of the closed-loop system can also be assessed by breaking the loop at different points (simulating corresponding component failures), and assessing the stability of the resulting open-loop transfer function matrices. In this case the number of encirclements of the origin is used to indicate open-loop unstable poles just as in the case of SISO systems.

This similarity with the classical control method of Nyquist diagrams is extended to the characteristic locus design method as is shown in the next two subsections.

5.2.2. The commutative controller.

Before the design procedure is described it is useful to explore some of the properties of the characteristic gain functions.

In the light of what has already been stated about the characteristic loci, it is clear that given a system with transfer function matrix $G(s)$, we need to design a controller which will give certain characteristic gains $q_1(s)$ for the open-loop transfer function matrix $Q(s) = G(s) \cdot K(s)$. There is no useful relationship however, which would link directly the functions $q_1(s)$, $q_1(s)$ and $k_1(s)$ and would thus provide guidelines to the designer about the form of $K(s)$. Here $q_1(s)$, $g_1(s)$ and $k_1(s)$ are the characteristic gains of $Q(s)$, $G(s)$ and $K(s)$ respectively.
The way around this problem is to restrict the controller $K(s)$ to the form

$$K(s) = W(s) \cdot \text{diag}\{k_i(s)\} \cdot V(s) \quad (5.11)$$

then

$$Q(s) = G(s) \cdot K(s)$$

and recalling equation (5.8)

$$Q(s) = W(s) \cdot \text{diag}\{g_i(s)\} \cdot V(s) \left[ W(s) \cdot \text{diag}\{k_i(s)\} \cdot V(s) \right]$$

and since $V(s) = W^{-1}(s)$,

$$Q(s) = W(s) \cdot \text{diag}\{k_i(s) \cdot g_i(s)\} \cdot V(s) \quad (5.12)$$

The forward loop transfer function matrix $Q(s)$ therefore has eigenvalues or characteristic gains

$$q_i(s) = k_i(s) \cdot g_i(s) \quad (5.13)$$

The controller $K(s)$ of equation (5.8) has the same modal structure as the plant transfer function matrix $G(s)$. As such will commute with $G(s)$ i.e.

$$G(s) \cdot K(s) = K(s) \cdot G(s)$$

This is the reason why $K(s)$ is called a *commutative* controller.
Now let

\[ Q(s) = W(s) \cdot \text{diag}\{q_i(s)\} \cdot V(s) = W \cdot \Lambda_q \cdot V = Q \]  \hspace{1cm} (5.14)

then

\[ Q^{-1} = [W \Lambda_q V]^{-1} = V^{-1} \Lambda_q^{-1} W^{-1} = W \Lambda_q V \]

and

\[ [I+Q]^{-1} = W [I+\Lambda_q]^{-1} V \]

The closed-loop transfer function matrix \( T(s) \) will therefore be

\[ T(s) = [I+Q]^{-1} Q = W [I+\Lambda_q]^{-1} V \cdot \{W \Lambda_q V\} = W \{ [I+\Lambda_q]^{-1} \Lambda_q \} V \]

or

\[ T(s) = W(s) \cdot \text{diag}\{\frac{q_i(s)}{1+q_i(s)}\} \cdot V(s) \]  \hspace{1cm} (5.15)

Having established the open and closed-loop system relationship in terms of the systems modal composition, we can proceed with examining the interaction of the closed-loop system.
5.2.3 Interaction and system eigenstructure.

From equation (5.15) we can see that there are two possible ways of eliminating interaction in the closed loop system. The first which is most suitable at low frequencies is to make

\[ q_i(s) = k_i(s) \cdot g_i(s) \gg 1 \]

by choosing an appropriate commutative controller with large characteristic gain functions \( k_i(s) \).

Then

\[ \text{diag}\left\{ \frac{q_i(s)}{1+q_i(s)} \right\} \approx I \]

and the closed loop transfer function matrix will then be

\[ T(s) \approx W(s) V(s) = I \]

which of-course is non-interacting.

The second way of reducing interaction can be used at high frequencies where, for a proper system, \( q_i(s) \ll 1 \). Then

\[ 1+q_i(s) \approx 1 \quad \text{and} \quad \text{diag}\left\{ \frac{q_i(s)}{1+q_i(s)} \right\} \approx \text{diag}\{q_i(s)\} \]

and the closed loop transfer function matrix

\[ T(s) \approx W(s) \text{diag}\{q_i(s)\} V(s) = Q(s) \quad (5.16) \]
will be non-interacting if the open-loop transfer function matrix, $Q(s) = G(s)K(s)$, is made non-interacting by an appropriate choice of the controller $K(s)$. In this case we can see from (5.16) that we need to choose $K(s)$ to make $Q(s) = I$, or

$$W(s) \sim I \iff V(s) = W^{-1}(s) \sim I$$

Another way of expressing this last relationship is to make the eigenvectors of $Q(s)$ aligned, at high frequencies, with the natural basis vectors $e_j$, or

$$\lim_{s \to \infty} \{ w_j(s) \} = e_j$$

where the vector $e_j$ is the standard basic vector i.e.

$$e_j(i) = \begin{cases} 1 & \text{for } i=j \\ 0 & \text{for } i \neq j \end{cases}$$

A measure of misalignment between the eigenvectors of the open-loop system and the natural basis vectors, can therefore be used to measure interaction at high frequencies for the open-loop as well as the closed-loop system.

Such a measure is defined by the relationship

$$\cos \theta_i(j\omega) = \frac{|<w_i(j\omega), e_i>|}{||w_i(j\omega)||} \quad (5.17)$$

clearly interaction is minimised when $\cos \theta_i(j\omega)$ approach
the value of 1 and conversely when $\theta_{i}(j\omega)$ are near to 0°. The angles $\theta_{i}(j\omega)$ are called the "misalignment angles" and are used to measure interaction at high frequencies.

5.2.4 Physical interpretation of system's eigenstructure.

In equation (5.15) is seen that the modal composition of the system is preserved under unity feedback. Also the characteristic gain functions of the closed-loop system is given by the same relationship that connects open and closed loop SISO systems, namely

$$t_{i}(s) = \frac{q_{i}(s)}{1+q_{i}(s)}$$

(5.18)

To expand on this last statement consider now the response of the closed-loop system to a demand signal, $r(s)$

$$y(s) = T(s).r(s) = W(s).\text{diag}\left\{\frac{q_{i}(s)}{1+q_{i}(s)}\right\}.V(s).r(s)$$

$$= W(s) \sum_{j=1}^{m} \frac{q_{i}(s)}{1+q_{i}(s)} e_{j}e_{j}^{T} V(s).r(s)$$

since

$$e_{j}e_{j}^{T} = \{e_{pq}\} = \begin{cases} 1 & \text{for } p=q=j \\ 0 & \text{for } p,q \neq j \end{cases}$$

-5.12-
thus the previous input output relationship can be written as

\[
y(s) = \sum_{j=1}^{n} w_j(s) \frac{q_i(s)}{1+q_i(s)} \left\{ e_j^T V(s) r(s) \right\}
\]

Since

\[
W(s)e_j = w_j
\]

The physical interpretation of equation (5.19) is that the system output can be constructed as a linear combination of the characteristic directions, scaled by the complex valued function

\[
y_j(s) = \frac{q_i(s)}{1+q_i(s)} \left\{ e_j^T V(s) r(s) \right\}
\]

These last functions are sometimes called the "characteristic outputs" of the system. They can be seen as the responses of the scalar system

\[
t_i(s) = \frac{q_i(s)}{1+q_i(s)}
\]

to demand signals

\[
r_j(s) = e_j^T V(s) r(s)
\]

The output of the system therefore can be seen as a linear
combination of the "characteristic outputs", which can be thought of as responses of SISO systems as described above. An insight to the generalised Nyquist criterion is obtained by observing that stability is maintained only as long as all closed-loop characteristic gains \( t_i(s) \) are stable. This of-course can be assessed if we apply the classical Nyquist criterion on every one of the open loop characteristic gain functions \( q_i(s) \).

5.2.5. The design method

The desired controller should give the system certain characteristics to provide adequate stability margins, accuracy and reduction of interaction. In particular high system gains should be achieved at low frequencies for good accuracy. This high gain will also provide reduction of interaction, at low frequencies. Stability margins can be achieved by phase compensating the loci around the critical point \((-1+j0)\), at medium frequencies. High frequency interaction can not be achieved however by relying on high loop gains. The controller therefore should provide alignment of the characteristic directions of \( Q(s) \) with the natural basis vectors \( e_j \).

The characteristics of the controller \( K(s) \) therefore are different at high medium and low frequencies. The characteristic locus design method takes account of this fact and proceeds with the design of the controller \( K(s) \) in three distinct stages.

First a controller \( K_h \) is designed to reduce interaction at high frequencies. This can be achieved by diagonalising the transfer function matrix at some high frequency \( \omega_h \).
Then the forward path transfer function matrix at $\omega_n$ becomes

$$G_1(j\omega_n) = G(j\omega_n)K_h = D(j\omega_n).$$

Where $D(j\omega_n)$ is the resulting diagonal transfer function matrix. Thus the desired controller is

$$K_h = G^{-1}(j\omega_n)D(j\omega_n).$$

The matrix $K_h$ however as described above is complex. Therefore it should either be dynamic thus increasing the overall complexity of the controller, or a real approximation to its desired complex value should be used. Such a real approximation can be provided by the ALIGN algorithm. Interaction can be assessed by plotting the misalignment angles $\theta_i(j\omega)$ at high frequencies.

In the second stage of the design process, a commutative controller $K_m$, is designed to provide phase compensation around the critical point, to the transfer function matrix $G_1(s) = G(s)K_h$, at some medium frequency $\omega_m$. Again it is desirable to design the controller $K_m(s)$ to have a relatively simple form, and to this end the ALIGN algorithm is used again to provide real approximations $A_m$ and $B_m$ to the complex eigenvector matrices $W_1(j\omega_m)$ and $V_1(j\omega_m)$ of $G_1(j\omega_m)$. Then

$$K_m(s) = A_m \text{diag}\{k_{m_1}(s)\} B_m$$  \hspace{1cm} (5.20)

where $k_{m_1}(s)$ are the transfer functions that will provide the necessary phase compensation to the characteristic gain.
functions of $G_1(s)$. Also in order not to upset the low interaction at high frequencies achieved by $K_h$, we should have

$$k_m(s) \overset{s \to j\omega_h}{\leftrightarrow} 1 \quad \Leftrightarrow \quad K_m(s) \overset{s \to j\omega_h}{\leftrightarrow} I$$  \hspace{1cm} (5.21)

In the third and last stage a controller is designed for low frequency compensation, of the transfer function matrix

$$G_2(s) = G_1(s) K_m(s) = G(s) K_h K_m(s)$$

This low frequency controller will have the form

$$K_1(s) = \frac{\alpha}{s} A_1 \text{diag}\{k_i\} B_1 + I$$  \hspace{1cm} (5.22)

Where $A_1$ and $B_1$ are ALIGN approximations to the complex eigenvector and reciprocal eigenvector matrices, of $G_2(j\omega_1)$. The gains $k_i$ are used to adjust the characteristic loci of $G_2(s)$ in order to provide low interaction at low frequencies and accuracy. The parameter $\alpha$ determines the duration of the integral action in the frequency domain. As the frequency increases the first term of the right hand side of equation (5.22) becomes negligible and the controller $K_1(s)$ tends to unity. In this way the effect of $K_1(s)$ on the action of $K_m(s)$ and $K_h$ can be made negligible.

The overall controller $K(s)$ is given by

$$K(s) = K_h K_m(s) K_1(s)$$  \hspace{1cm} (5.23)
and the open-loop transfer function matrix can now be written as

\[ Q(s) = G(s)K(s) = G(s)K_m(s)K_1(s) \]  \hspace{1cm} (5.24)

The success of the design will depend on the accuracy of the approximations provided by the ALIGN algorithm as well as the choice of characteristic gain functions of \( K_m(s) \) and \( K_1(s) \) and the integral action parameter \( \alpha \).

5.3 THE INVERSE AND THE DIRECT NYQUIST ARRAY METHOD.

5.3.1 Introduction

In the characteristic locus design method, input output relationships and stability criteria in terms of the system's eigenstructure are utilised to achieve the required performance. In effect the design process tries to alter explicitly the internal input-output mapping mechanism of the system by changing the system's eigenstructure. In this process the system is always seen as a true multivariable system.

The philosophy of the Nyquist array methods is different. Here the first step is to reduce interaction sufficiently and thus be able to regard the system as a collection of single-input, single-output systems. Single loop controllers are then designed for each channel to give the required performance using well known classical control techniques. There are two Nyquist array methods the inverse Nyquist array (INA) and the direct Nyquist array (DNA) method. The main difference between them is that the former uses the inverse forward path transfer function matrix whereas the latter is
based on the direct open-loop transfer function matrix.

Before describing the Nyquist array design methods however, it will be useful to introduce some of the concepts and the terminology which will be used in this section.

Consider again the system block diagram of figure 5.2. Let now the controller \( K(s) \) be the product of two matrices

\[
K(s) = K_d(s) F 
\]  

(5.25)

where \( F \) is a diagonal matrix of gains,

\[
F = \text{diag}\{f_i\} 
\]  

(5.26)

The block diagram of figure 5.2 can then be redrawn as in figure 5.3. Note here that only the transfer function matrices inside the loop affect closed-loop stability. The loop transfer function matrix of the closed-loop system is therefore given by

\[
T(s) = [I + Q(s) F]^{-1} Q(s) 
\]  

(5.27)

where

\[
Q(s) = G(s) K_d(s) 
\]

Define now the transfer function matrices

\[
\hat{Q}(s) = \{\hat{q}_{ij}(s)\} = Q^{-1}(s) 
\]  

(5.28)

and

\[
\hat{T}(s) = \{\hat{t}_{ij}(s)\} = T^{-1}(s) 
\]
From equation (5.27) then we have
\[ T^{-1}(s) = Q^{-1}(s) \left[ I + Q(s)F \right] \]
\[ = Q^{-1}(s) + F \]
and finally
\[ \hat{T}(s) = \hat{Q}(s) + F \] (5.29)

and
\[ \hat{t}_{ij}(s) = \begin{cases} 
q_{ii}(s) + f_i & \text{for } i=j \\
q_{ij}(s) & \text{for } i \neq j 
\end{cases} \] (5.30)

The simplicity of equations (5.29) and (5.30) is the reason that the inverse Nyquist array method was developed first, and still preferred by many control engineers.

5.3.2 Diagonal dominance and graphical criteria for stability.

It has already been mentioned that the strategy adopted by the Nyquist array methods is to reduce interaction to such a level that will make it possible for single-loop controllers to be designed to achieve the required performance for any input-output pair. In designing such controllers care must be taken to ensure closed loop stability. If interaction is eliminated altogether the assessment of stability for every loop can be made on an individual loop basis. This, on the other hand, will not be possible if some interaction is present in the forward path transfer function matrix, since the effects of its off-diagonal elements can not be ignored.
Complete elimination of interaction is not however always possible and even when it can be achieved it will often result in complicated high order controllers. There is therefore the need to be able to assess closed-loop stability in each individual loop when the system is nearly decoupled but has not been made diagonal. This would facilitate single-loop controller design, for the system provided that interaction has been reduced below a certain critical level in each loop. This critical level of interaction is defined in the Nyquist array methods to occur when the sum of the magnitudes of the off-diagonal terms in every row (column) of the transfer function matrix, is less than the magnitude of the corresponding diagonal element. Such systems are not in general diagonal but are said to be diagonally dominant by row or column. To express this through a mathematical relationship, a system described by a transfer function matrix \( Q(s) \) is said to be diagonally dominant by row when

\[
|q_{ii}(s)| - \sum_{j=1, j \neq i}^{m} |q_{ij}(s)| > 0
\]  

(5.31)

and diagonally dominant by column when

\[
|q_{ii}(s)| - \sum_{i=1, i \neq j}^{m} |q_{ij}(s)| > 0
\]  

(5.32)

The reason for having diagonal dominance as a precondition for assessing closed-loop stability is a well known theorem due to Gershgorin. According to this theorem the eigenvalues of a complex \( m \times m \) matrix lie within the union of \( m \) discs in the complex plane where the \( i \)th disc has centre the \( i \)th diagonal element and radius the sum of the magnitudes of the
off-diagonal elements in the same row (column).

The significance of this theorem is that it locates the characteristic loci of a diagonally dominant system to bands in the complex plane which are defined in term of the Nyquist diagrams of the diagonal elements of the transfer function matrix.

Indeed suppose that the array of the Nyquist diagrams of the diagonal elements of the transfer function matrix are plotted and circles are drawn corresponding to every frequency point \( \omega_k \), with

centre at \( q_{ii}(j\omega_k) \) and radius \( d_i(\omega_k) \) such that

\[
d_i(\omega_k) = \sum_{j=1 \atop j \neq i}^{m} |q_{ij}(j\omega_k)|, \text{ (for row dominance)}
\]

or

\[
d_i(\omega_k) = \sum_{i=1 \atop i \neq j}^{m} |q_{ij}(\omega_k)|, \text{ (for column dominance)}
\]

Then according to Gershgorin's theorem the eigenvalues of \( Q(j\omega_k) \) will lie within the union of the circles, which are called the Gershgorin circles, that correspond to the frequency point \( \omega_k \). It follows from this that the band swept by the Gershgorin circles drawn for each point frequency point, will contain the system's characteristic loci. A Nyquist array of a 2x2 system with Gershgorin circles superimposed is shown in figure 5.4.

If the system is diagonally dominant, the Gershgorin band will not contain the origin. The number of encirclements of the origin by the characteristic loci can in this case be

\[\text{---5.21---}\]
determined by counting the number the Gershgorin bands encircle the origin. The number of encirclements of the origin by the Gershgorin bands, will of course be the same with the number of encirclements by the Nyquist diagrams of the diagonal elements of the transfer function matrix. If the system is diagonally dominant therefore stability can be assessed by a Nyquist type criterion using the Nyquist plots of the diagonal elements of the transfer function matrix, instead of the characteristic loci. This is formally expressed in the following theorem due to Rosenbrock.

**Theorem**: Let both the open-loop transfer function matrix $Q(s)$ and the closed-loop transfer function matrix $T(s)$ be dominant on the Nyquist contour $D$. Let $q_{ii}(s)$ map $D$ onto $\Gamma_{q_{i}}$ and $t_{ii}(s)$ map $D$ onto $\Gamma_{t_{i}}$. Let these encircle the origin $N_{q_{i}}$ and $N_{t_{i}}$ times respectively. Then if the open-loop system has $p_{o}$ poles in the closed right half plane, the closed-loop system is asymptotically stable if and only if

$$\sum_{i=1}^{m} N_{t_{i}} - \sum_{i=1}^{m} N_{q_{i}} = p_{o} \quad (5.34)$$

Note here that the original proof of the theorem as this was offered by Rosenbrock was not accomplished in terms of characteristic loci.

If in the discussion above the transfer function matrices $Q(s)$ and $T(s)$ are replaced by their inverses the stability criterion described in the last theorem is replaced by an analogous criterion corresponding to the inverse Nyquist stability criterion in classical control.
In particular equation (5.34) becomes

\[
\sum_{i-1}^{m} N_{q_i} - \sum_{i-1}^{m} N_{t_i} = P_0 \tag{5.35}
\]

where the sums on the left hand side represent the number of encirclements of the inverse Nyquist array of \( Q(s) \) and \( T(s) \).

The justification of this last statement comes from the fact that the Gershgorin bands of the inverse transfer function matrix contain the inverses of the characteristic loci of the direct transfer function matrices. This is true because the eigenvalues of

the inverse of a matrix are the reciprocals of the eigenvalues of this matrix, i.e

\[
\lambda_i [Q(s)] = \frac{1}{\lambda_i [Q^{-1}(s)]} = \frac{1}{\lambda_i [Q(s)]} \tag{5.36}
\]

In order to develop a graphical criterion that will enable us to assess the closed-loop stability from the open-loop Nyquist loci we need to be able to bound the transfer function \( t_{ii}(s) \) to an area around the transfer function \( q_{ii}(s) \). This result has been provided by Ostrowski who has shown that

\[ t_{ii}^{-1}(s) - f_i \neq q_{ii}(s) \text{ (in general)} \]

is contained in the Gershgorin band of \( \hat{q}_{ii} \).

The following simple graphical criterion for assessing the system's closed-loop stability can now be stated. If the Gershgorin bands of the inverse Nyquist loci do not cross the real axis between 0 and \((-f_{i\text{max}}, 0)\) in the \( m \) complex planes then
the feedback system is stable (if \( p_o = 0 \)) for all gains \( f_i \) such that
\[
0 \leq f_i \leq f_{i \text{max}}
\]

In fact Ostrowski's result can be used to locate the transfer function \( t_{ii}^{-1}(s) - f_i \), in a concentric but narrower band than the Gershgorin band. The radius of the circles that form this new band, which is called the Ostrowski band, is:

\[
d_i(s) = \left[ \max_{j \neq i} \frac{d_j(s)}{f_j + d_j(s)} \right] d_i(s)
\]

(5.37)

The Ostrowski band being narrower than the Gershgorin band, can be used to provide a measure of gain margin, once all the gains \( f_i \) are chosen for stability using the Gershgorin band.

Note also that the transfer function
\[
t_i^{-1}(s) = t_{ii}^{-1}(s) - f_i
\]

(5.38)

which is contained in the Ostrowski band is the transfer function seen in the ith loop, (between input \( u_i(s) \) and output \( y_i(s) \), when all loops but the ith are closed. The Ostrowski bands therefore can be used as the inverse Nyquist diagrams on which the designs of the single loop controllers are based. However in the case of the direct Nyquist array, although Ostrowski circles can still be drawn, they no longer provide bounds to the transfer function seen between the ith input and the ith output when all loops but the ith are closed. The design in this case will be based solely on the direct Nyquist array with Gershgorin bands.

The graphical criterion described above for the inverse
Nyquist array can be used in the case of the direct Nyquist array too, if \( f_i \) and \( f_{i \text{max}} \) are replaced by their reciprocals.

Following the analysis above, the design technique can be outlined as follows. First the controller \( K_d(s) \) is designed to make diagonally dominant the transfer function matrix

\[
Q(s) = G(s)K(s)
\]

if the direct Nyquist array method is used, or the transfer function matrix

\[
\hat{Q}(s) = \hat{K}(s)\hat{G}(s) \quad (= Q^{-1}(s) = K^{-1}(s)G^{-1}(s) )
\]

if the inverse Nyquist array is used. The loop gains \( f_i \) are then chosen to give the required stability margins while at the same time are high enough to eliminate closed-loop interaction.

Finally single-loop controllers are designed using classical control techniques to give the required performance to the SISO system seen between the \( i \)th input and \( i \)th output. Note that the transfer function of this system is given by equation (5.38) and if the inverse Nyquist array is used it is contained in the Ostrowski bands.

### 5.3.3 Achieving diagonal dominance

It is clear that the main difficulty with the Nyquist array design methods is to design the first controller \( K_d(s) \) that will make the forward path transfer function diagonally dominant. Indeed it is not guaranteed that such a controller even exists. In this section a number of different techniques
for designing the controller $K_d(s)$ to make the system diagonally dominant are described.

**Diagonalisation at a single frequency point**

A first attempt would be to try to diagonalise the forward path transfer function matrix at one specific frequency and hope that this will make it diagonally dominant over the frequency range of interest. If such a controller is designed to make $Q(j\omega_k)$ diagonal at a specific frequency point $\omega_k$, it will be given by

$$K_d(j\omega_k) = G^{-1}(j\omega_k)$$

(5.39)

since

$$Q(j\omega_k) = G(j\omega_k)K_d(j\omega_k)$$

Now $G^{-1}(j\omega_k)$ will only be real if $\omega_k = 0$. It is therefore reasonable the first trial is done with the controller $K_d = G^{-1}(0)$. In any other frequencies $K_d$ is calculated to be a real approximation to $G^{-1}(j\omega_k)$. One way of calculating this real approximation to the complex matrix $G^{-1}(j\omega_k)$ is to use the ALIGN algorithm.

Clearly however this approach will only be successful in a limited number of cases.

**Use of permutation matrices for simple row and column operations**

An alternative and equally simple method is to design $K_d$ to
be the product of some permutation matrices \( K_{1d}, K_{2d}, \ldots, K_{pd} \). The process will be carried out in an interactive computing environment in which the designer will have the Nyquist array displayed on the screen and decide which elementary row or column operation will improve the dominance of the forward path transfer function matrix (or its inverse). The elementary row or column operation will then be accomplished by the permutation matrix \( K_{qd} \) which will improve interaction of the transfer function matrix

\[
G_{q-1}(s) = G(s) K_{1d} K_{2d} \cdots K_{(q-1)d} \quad (5.40)
\]

The overall controller is then given by

\[
K_d = K_{1d} K_{2d} \cdots K_{pd} \quad (5.41)
\]

Although this approach gives a useful and systematic way of designing real controllers for diagonal dominance, it becomes very tedious for large systems with many inputs and outputs.

**Pseudodiagonalisation** [15].

Another way of designing controllers for diagonal dominance is to use some function optimisation technique to minimise the sum of the off diagonal elements of the transfer function matrix. The first such method was developed by Hawkins, and is widely referred to as pseudodiagonalisation.
Hawkins proposed to minimise the measure

\[ J_j = \sum_{k=1}^{N} W_k \left\{ \sum_{i=1}^{m} \left| q_{ij}(j\omega_k) \right|^2 \right\} \]

subject to \( \| k_j \| = 1 \)

(5.42)

Where \( q_{ij}(j\omega_k) \) is the \((i,j)\)th element of the transfer function matrix \( Q(j\omega_k) = G(j\omega_k)K(j\omega_k) \) and \( k_j \) is the jth column of \( K \). \( W_k \) is a weighting factor that will attach different significance to different frequency points. The \((i,j)\)th element of \( Q \) can now be written as

\[ q_{ij}(j\omega) = g_i^T(j\omega) k_j \]

(5.43)

and

\[ J_j = \sum_{k=1}^{N} W_k \left\{ \sum_{i=1}^{m} \left| g_i^T(j\omega_k) k_j \right|^2 \right\} \]

subject to \( \| k_j \| = 1 \)

(5.44)

Using Lagrange multipliers this leads to an eigenvalue problem that can be solved easily with existing numerical methods.

The problem with pseudodiagonalisation is that in the process of minimising the off diagonal elements it does not prevent the diagonal element \( q_{jj}(j\omega) \) from becoming very small and therefore diagonal dominance may not be achieved.
Pseudodecoupling [16]

To overcome the problem associated with pseudodiagonalisation, Ford and Daly suggested an alternative method called pseudodecoupling. In this method the measure

$$J_j = \frac{\sum_{k=1}^{N} w_k \left\{ \sum_{i=1}^{m} |q_{ij}(j\omega_k)|^2 \right\}}{\sum_{k=1}^{N} w_k |q_{jj}(j\omega_k)|^2}$$

is minimised instead. The cost function $J_j$ is however the reciprocal of the cost function maximised by the ALIGN algorithm, which provides the numerical solution to the problem. The method can be extended to design dynamic compensators to achieve diagonal dominance if no real compensator can be found which will successfully accomplish this task.

Perron-Frobenius theory of positive matrices [17].

It is often the case that only some of the channels will be non-dominant in a system, whereas the rest could have a varying degree of diagonal dominance. In these cases it would be useful to know whether dominance can be achieved by means of a scaling diagonal compensator which would redistribute
dominance between the different channels of the system. Such a diagonal compensator would be particularly useful in the case of the inverse Nyquist array since its inversion will be a simple task even when it is dynamic.

A simple test which determines whether diagonal dominance can be achieved by a diagonal precompensator is given by the Perron-Frobenius theory of positive matrices [18]. Furthermore if such a controller exists, the theory provides the "best" or optimal diagonal compensator which will achieve dominance at each specific frequency point. This information can be used as it will be described later to design compensators for diagonal dominance over the whole frequency range of interest.

Before the Perron-Frobenius test is described, we need to define two matrices. The first, $\Omega(G)$ is a diagonal matrix which has as diagonal elements the absolute values of the diagonal elements of the argument matrix $G(s)$. I.e.

$$\begin{align*}
\Omega(G) = \{\Omega_{ij}\} &= \begin{cases} 
\Omega_{ii} &= |g_{ii}(s)| \\
\Omega_{ij} &= 0, \ i\neq j
\end{cases}, \quad (5.46)
\end{align*}$$

Define also the matrix $M(s)$ such that

$$M(s) = \{m_{ij}(s)\} = \{|g_{ij}(s)|\} \quad (5.46a)$$

Now define the normalised comparison matrix $\Gamma(s)$

$$\Gamma(s) = \Omega^{-1}(G) M(s) \quad (5.47)$$

for the case of the direct Nyquist Array and

$$\Gamma(s) = \hat{G}(s) \Omega^{-1}(\hat{G}) \quad (5.48)$$
for the case of the Inverse Nyquist Array.

By definition the normalised comparison matrix will be primitive (i.e. there will exist a real scalar \( q \) so that the matrix \( \Gamma\mathcal{C}(s) \) will have all its elements greater than zero), at each specific frequency point. According to the Perron-Frobenius theory of positive matrices, \( \Gamma(s) \), will have a real and positive eigenvalue, \( \lambda_p\{\Gamma(s)\} \), called the Perron-Frobenius eigenvalue, whose modulus is larger than the modulus of all the eigenvalues of \( \Gamma(s) \). Mees has shown that if

\[
\lambda_p\{\Gamma(j\omega_k)\} < 2
\]

the system can be made diagonally dominant at \( \omega_k \) by means of a diagonal compensator. The diagonal precompensator that will produce the most diagonally dominant system at a frequency \( \omega_k \) in the case of the direct Nyquist array, is the right eigenvector of \( \Gamma(j\omega_k) \) which corresponds to the Perron-Frobenius eigenvalue. The same result will be achieved by the left (but not the right) Perron-Frobenius eigenvector of \( \Gamma(j\omega_k) \), if the Inverse transfer function matrix is to be made diagonally dominant at the frequency point \( \omega_k \). In general however we will need to make the system diagonally dominant over a range of frequencies rather than one particular frequency point. There are two ways to achieve this using diagonal compensators derived from the Perron-Frobenius theory.
The first way is to create a worst case normalised comparison matrix $T$, defined as

$$ T = \{t_{ij}\}, \text{ with } t_{ij} = \max_{\omega} \{\gamma_{ij}(j\omega)\} \quad (5.49) $$

where $\gamma_{ij}(j\omega)$ is the $(i,j)$th elements of $\Gamma(j\omega)$.

If the Perron-Frobenius eigenvalue of $T$, $\lambda_p(T)$, is less than 2, then the Perron-Frobenius eigenvector (right for the direct array and left for the inverse), will make the system diagonally dominant. In general however,

$$ \lambda_p(T) > \sup \{\lambda_p(\Gamma(j\omega))\} \quad (5.50) $$

Therefore, if the controller does not need to be real, it will be unnecessarily conservative. The alternative approach will be to plot the value of each eigenvector element against frequency. Provided that

$$ \sup \{\lambda_p(\Gamma(j\omega))\} < 2 $$

these curves may be fitted by the gain curves of the frequency responses of some scalar transfer functions $v_i(s)$. The desired compensator is then given by

$$ K(s) = \text{diag}\{v_i(s)\} \quad (5.51) $$

The main disadvantage of the Perron-Frobenius method is that it produces row dominance in the direct transfer function matrix and column dominance in the inverse transfer function matrix. This is problematic since we design our controllers to post-multiply the direct transfer function matrix and
pre-multiply the inverse. Therefore even if the single-loop controllers are diagonal, they will in general disturb the dominance achieved by the Perron-Frobenius compensator. Dominance will not be affected only if all single-loop controllers are equal.

On the other hand the plot of the Perron-Frobenius eigenvalue against frequency gives a measure of the overall dominance of the system. This is an obvious improvement to the situation up to now, whereby any dominance information was restricted to individual channels. In addition to this information, provided that the Perron-Frobenius eigenvalue is less than 2, the method provides the optimal scaling matrix that will achieve diagonal dominance. This scaling matrix is of-course diagonal, a fact that makes it particularly suitable for the INA method where the inversion of the designed compensators is always a problem.

5.4 ROBUST STABILITY AND PERFORMANCE [18,19,20]

The principal reason for using feedback control schemes is their ability to deal with uncertainty in the system. The uncertainty may be due to modelling errors, parameter variations with time, system non-linearities, input or output perturbations or measurement noise. The uncertainty is modelled in figure 5.1, by the output perturbation function \(d(s)\), which models non-linearities, modelling-errors, and time dependency as well as output perturbations and the measurement noise function \(m(s)\). It has already be shown that

\[
y(s) = S(s)d(s) + T(s)P(s)r(s) - T(s)m(s) \quad (5.52)
\]
where

\[
S(s) = (I + G(s)K(s))^{-1}
\]

\[
T(s) = (I + G(s)K(s))^{-1} G(s)K(s) = I - S(s)
\]

For good system performance we need

1. \( S(s) \) small where \( d(s) \) is large to reduce sensitivity to parameter variations.
2. \( T(s) \) small where \( m(s) \) is large to reduce the effects of measurement noise.
3. \( T(s)P(s) \approx I \) for good tracking in the bandwidth of interest.

and

4. Above all stability under all possible parameter variations.

The concept of the size of a matrix has not however been defined. In single input single output systems the size of the transfer function at a specific frequency point is the gain of a system which is a scalar. In multivariable systems the gain will depend upon the direction of the input vector as well as its size (length or norm). It is for this reason that a multivariable system does not have a single gain, but we can only hope to be able to put bounds to the range of values its gain can take. An upper bound to the system gain can be
obtained through the spectral matrix norm which is defined as follows (note here that for simplicity the function argument \((j\omega)\) is often omitted in what follows in this section).

\[
\| G(s) \|_s = \sup_{q \neq 0} \frac{\| Gq \|}{\| q \|} 
\]  
(5.53)

Where \(q\) is any appropriately dimensioned vector.

If the vector norms in (5.53) are taken to be Euclidean norms then the spectral norm is equal to the maximum singular value of \(G(j\omega)\). \([44]\)

\[
\bar{\sigma}(\omega) = \sup_{q \neq 0} \frac{\| G(j\omega)q(j\omega) \|_2}{\| q(j\omega) \|_2} 
\]  
(5.54)

In this way the maximum singular value of \(G(j\omega)\) provides an upper bound to its gain. A lower bound to the gain is provided by the minimum singular value of \(G(j\omega)\) as is shown next.

Consider the singular value decomposition of the transfer function matrix \(G\):

\[
G = \Sigma \Sigma U^H \Rightarrow G^{-1} = U \Sigma^{-1} Y^H 
\]

\[
\Rightarrow \| G^{-1}(j\omega) \|_s = \frac{1}{\bar{\sigma}(\omega)} 
\]  
(5.55)

Or

\[
\frac{\| G^{-1}(j\omega) y(j\omega) \|_2}{\| y(j\omega) \|_2} \leq \frac{1}{\bar{\sigma}(\omega)} 
\]
but

\[ y(j\omega) = G(j\omega)u(j\omega) \]

\[ \frac{\|u(j\omega)\|_2}{\|G(j\omega)u(j\omega)\|_2} \leq \frac{1}{\sigma(\omega)} \]

Or

\[ \frac{\|G(j\omega)u(j\omega)\|_2}{\|u(j\omega)\|_2} \geq \sigma(\omega) \] (5.56)

Substituting \( u(j\omega) \) for \( q(j\omega) \) in (5.54) and combining with (5.56) gives

\[ \sigma(\omega) \leq \frac{\|G(j\omega)u(j\omega)\|_2}{\|u(j\omega)\|_2} \leq \bar{\sigma}(\omega) \] (5.57)

Equation (5.57) shows that the maximum and minimum singular values can be used to define the upper and lower bounds of the gain of a transfer function matrix. Having established these bounds the design requirements for robust stability and performance can be analysed.

5.4.1. Robust stability.

Suppose that a physical system \( G_r(s) \) is described by the transfer function model \( G_n(s) \). The error between \( G_r(s) \) and \( G_n(s) \), due to parameter variation with time, modelling errors,
non-linearities and unpredicted perturbation, can be expressed as a percentage of $G_r(j\omega)$ at each frequency point $\omega$. If the error factor is denoted by $\Delta(s)$ then,

$$G_r(j\omega) = G_n(j\omega) + \Delta(j\omega) G_n(j\omega) = \left[ I + \Delta(j\omega) \right] G_n(j\omega) \quad (5.58)$$

If we make the assumption that the perturbation $\Delta(s)$ and the system model $G_n(s)$ are both stable, and the loop is designed to be stable for $\Delta(S) = 0$ the system would be unstable if $\det[I+G_r(s)K(s)]$ encircles the origin at least once. In this case there exist a $\Delta(s)$ such that $\det[I+G_r(s)K(s)] = 0$. By the same argument for the closed loop system to be stable, $\det[I+G_r(j\omega)K(j\omega)] \neq 0$ for all possible values of $\Delta(j\omega)$ and any $\omega$ on the Nyquist contour. This is true only if $I+G_r(j\omega)K(j\omega)$ has full rank, which in turn means that

$$\sigma[I + G_r(j\omega) K(j\omega)] > 0 \quad (5.59)$$

for all $\omega$ and all possible $G_r(j\omega)$.

or

$$\sigma[I+G_n(j\omega) K(j\omega) + \Delta(j\omega) G_n(j\omega) K(j\omega)] > 0 \quad (5.60)$$
It can be shown that equation (5.60) leads to the relationship

\[ \bar{\sigma}[\Delta(j\omega)] < \frac{1}{\bar{\sigma}[G_n(j\omega) K(j\omega)]} \]

\[ = \frac{1}{\bar{\sigma}[T(j\omega)]} \quad (5.61) \]

The smallest perturbation \( \Delta(j\omega) \) that causes instability at a frequency \( \omega \), is therefore

\[ \bar{\sigma}[\Delta_m(j\omega)] = \frac{1}{\bar{\sigma}[T(j\omega)]} \quad (5.62) \]

Equation (5.61) gives in effect a multivariable gain margin for the system. If the largest possible perturbation of \( \Delta(j\omega) \), \( \delta_m(j\omega) \) is known, then the controller \( K(s) \) is designed so that

\[ \bar{\sigma}[T(j\omega)] < \frac{1}{\delta_m(j\omega)} \quad (63) \]

Stability then is guaranteed for all possible perturbations.

5.4.2 Relationships between open-loop and closed-loop principal gains

The objectives of the closed loop system are to provide good tracking of the reference signal, and to reduce sensitivity and noise propagation. The controller however is usually designed to give certain characteristics to forward
path transfer function which in its turn will give the required performance to the closed-loop system. It is therefore very useful to establish relationships between the open and closed loop characteristics of the system which will act as guidelines to the designer when designing the forward path controller. Such relationships are developed next for sensitivity, noise propagation and tracking of the reference signal.

(a) Sensitivity

From equation (5.52) we see that we require the sensitivity function to be small where \( d(s) \) is large. This translates to small

\[
\tilde{\sigma}[S(j\omega)] = \tilde{\sigma}[(I+G(j\omega)K(j\omega))^{-1}]
\]

but

\[
\tilde{\sigma}[(I+G(j\omega)K(j\omega))^{-1}] = \frac{1}{\tilde{\sigma}[I+G(j\omega)K(j\omega)]} \\
\leq \frac{1}{\tilde{\sigma}[G(j\omega)K(j\omega)] - 1} = \frac{1}{\tilde{\sigma}[G(j\omega)K(j\omega)]}
\]

if \( \tilde{\sigma}[G(j\omega)K(j\omega)] \gg 1 \)

Therefore for low sensitivity we need:

large \( \tilde{\sigma}[G(j\omega)K(j\omega)] \), where \( d(j\omega) \) is large.

(b) Noise Propagation

Equation (5.52) shows that in order to reduce noise propagation the complementary sensitivity function \( T(s) = \)
I-S(s) should be made small. In other words we need to have small

\[ \bar{\sigma}[T(j\omega)] = \bar{\sigma}[I-S(j\omega)] = \bar{\sigma}[I-(I+G(j\omega)K(j\omega))^{-1}] \\
= \bar{\sigma}[I+(G(j\omega)K(j\omega))^{-1}] = \frac{1}{\bar{\sigma}[I+(G(j\omega)K(j\omega))^{-1}]} \]

Therefore for reduced noise propagation we want \( \sigma(I+(GK)^{-1}) \) as large as possible. Or \( \sigma((GK)^{-1}) \) as large as possible. Or, finally, we want

\( \bar{\sigma}(G(j\omega)K(j\omega)) \) small at frequencies where \( m(j\omega) \) is large.

This requirement conflicts with the requirement for low sensitivity. In practice however this conflict is usually resolved because the frequency bands where \( m(s) \) and \( d(s) \) are large are quite distinct.

The final design objective is

(c) Good Tracking of Reference signal

Again from equation (5.52) we see that if the designer has only "one degree of freedom", i.e. \( P(s) = I \), then for good tracking we need \( T(s) \equiv I \). Or

\[ T(j\omega) = I-S(j\omega) = I-(I+G(j\omega)K(j\omega))^{-1} \equiv I \]

Or \((I+GK)^{-1}\) must be very small.

I.e.

\[ \bar{\sigma}[(I+G(j\omega)K(j\omega))^{-1}] \ll 1 \]
or

\[ \sigma(I + G(j\omega)K(j\omega)) \gg 1 \]

or finally

\[ \sigma(G(j\omega)K(j\omega)) \gg 1 \]
Figure 5.1 Block diagram of closed-loop system.

Figure 5.2 Block diagram of closed-loop system with the effects of noise and disturbance ignored.
Figure 5.3  Block diagram of closed-loop system.
Equivalent to the block diagram of figure 5.2

Figure 5.4 Nyquist array with Gershgorin circles superimposed at $\omega_0$
CHAPTER 6

THE CONTROLLER DESIGN

6.1 INTRODUCTION

It has already been shown in chapter 2 that the ideal performance of the test rig would be achieved by a controller which would give an interaction free system, with flat frequency responses for each of its input-output pairs. It has also be shown that this requirement can be achieved by open-loop control. Any such scheme however will not be able to maintain its performance in the case of change in the system dynamics, as the test progresses, and it will propagate any disturbance unaltered to its output. Disturbance in this context, is a function which describes the unmodelled system dynamics and changes in system dynamics due to time dependency, as well as real external disturbances.

In this chapter the design process is described which produced controllers for closed-loop systems using both the Nyquist array and the characteristic locus method. The designs are evaluated against the criteria of accuracy as well as their ability to minimise the effects of uncertainty.

The main obstacle to designing such controllers is the interaction present in the system. The size of the problem can be quantified by examining the magnitude of the experimental frequency responses between all combinations of input output pairs of the test rig as this is shown in figure 6.1. Element [1,1] here is the frequency response produced by driving only the vibrator attached at the free end of the beam while measuring the displacement at the same
point. Measuring the displacement at the mid-point of the beam produces the frequency response of element \([1,2]\). Driving only the vibrator at the middle of the beam produces the frequency response element \([2,2]\) and \([2,1]\) when measuring displacement at mid-point and the free-end of the beam respectively. The frequency responses of the elements \([1,2]\) and \([2,1]\) constitute the unwanted interaction between the vibrator-LVDT channels at the mid-point and free-end of the beam. It is seen from figure 6.1 that interaction is over 30% at low frequencies in channel 1, and more than 100% in channel 2. Moreover from the magnitude curves of the Bode-plot in figure 6.2, it seen that interaction worsens at high frequencies. It is against this background that controllers must be designed.

The primary data for the controller design are experimentally obtained frequency responses. This minimises the modelling errors that would be present in any frequency response produced by analytical models. It also shows that frequency domain techniques can be used to successfully design controllers, without the need for analytical mathematical models. This is particularly important in structural testing, where the dynamics of the test objects are often too complex to make analytical modelling a practical proposition. Frequent changes of test objects add to the need to by-pass analytical modelling.

This is made possible by the use of C.A.D., in particular the Multivariable Frequency Domain (M.F.D.) toolbox of MATLAB [21,22], to assist with frequency response calculations. By this method open or closed-loop compensated frequency responses are calculated directly from the frequency responses of the system and the controller, without the need to calculate the transfer function matrix or state-space model of the compensated system. The design is done entirely in the frequency domain and all that is needed in terms of primary data is the "plant's" frequency
response which in this case was obtained experimentally.

6.2 NYQUIST ARRAY DESIGN

The first step in the Nyquist array method is to ensure that interaction is sufficiently low for the system to be diagonally dominant in the direct or the inverse complex plane. Note here that for 2-input 2-output systems, dominance in a row sense in one plane is possible if and only if, there is column dominance in the other. Similarly column dominance in one of the planes is ensured and is only possible, if there is row dominance in the other. In other words examining the dominance characteristics of a 2x2 system in the direct plane also examines its dominance in the inverse plane and vice-versa. Furthermore the row (column). dominance levels in the direct plane are the same as the column (row) dominance levels in the inverse plane.

The direct Nyquist array of the rig with column Gershgorin bands superimposed, is shown in figure 6.3, while the Nyquist array of the system with row Gershgorin bands is shown in figure 6.4. It is seen in both these figures that the system is very non-dominant in channel 2. A controller which will make the system diagonally dominant must therefore be designed.

A number of approaches to designing controllers for diagonal dominance, which is the most difficult part of the design, have been described in chapter 5. If the inverse Nyquist array method is to be used, it would be beneficial to have a diagonal controller. In this case the controller \( K_d(s) \) is designed to make the transfer function matrix

\[
\hat{Q}(s) = Q^{-1}(s) = \hat{K}_d(s) \hat{G}(s) = K_d^{-1}(s) G^{-1}(s)
\]  

(6.1)
diagonally dominant, where

\[ G(s) = \{g_{ij}(s)\} \]

is the transfer function matrix of the plant (in this case the test rig).

The forward path transfer function matrix will be

\[ Q(s) = G(s)K_d(s) \]  \hspace{1cm} (6.2)\]

Therefore, in order to obtain the actual controller, the controller designed to make \(Q^{-1}(s)\) diagonally dominant must be inverted. Clearly if a dynamic controller is designed its inversion is going to be difficult unless it is diagonal.

The test rig forms a two-input two-output system and the controller \(K_d(s)\), could be designed in the direct plane to make both \(Q(s)\) and its inverse diagonally dominant. In this case \(K_d(s)\) does not need to be diagonal, since no inversion is going to take place. It was felt however that the method used here should be applicable to the general dynamic structural test configuration which is not limited to two channel systems. This meant that only a combination of scalar and diagonal dynamic compensators would be acceptable. Such compensators which will make the system diagonally dominant can be designed using the Perron-Frobenius theory of positive matrices, as it is described in chapter 5.

In the first step of this process the Perron-Frobenius eigenvalue, (or largest positive real eigenvalue) of the normalised comparison matrix [24]

\[ \Gamma(j\omega) = \Omega^{-1}(G)M(j\omega) \]  \hspace{1cm} (6.3)\]

-6.4 -
must be examined as \( \omega \) varies in the frequency range of interest.

Note here that

\[
\Omega(G) = \{ \Omega_{1j} \} = \begin{bmatrix}
\Omega_{11} = |g_{11}(j\omega)| \\
\Omega_{ij} = 0, \quad i \neq j
\end{bmatrix} \quad (6.4)
\]

and

\[
M(j\omega) = \{ m_{ij}(j\omega) \} = \{ |g_{ij}(j\omega)| \} \quad (6.5)
\]

A diagonal compensator which would scale the system into diagonal dominance at some frequency \( \omega = \omega_k \), exists only when the value of the Perron-Frobenius eigenvalue of the normalised comparison matrix is below 2 at this particular frequency.

The graph of the Perron-Frobenius eigenvalue of \( \Gamma(j\omega) \) against frequency is shown in figure 6.5. Since the value of the Perron-Frobenius eigenvalue of the normalised comparison matrix, exceeds the value of 2 at approximately 330 rad/s, no diagonal compensator exists that will make the system diagonally dominant over the whole frequency range.

To give some physical meaning to the curve of figure 6.5, termed the P-F dominance graph, consider the system's Nyquist array, shown in figures 6.3 and 6.4. It can be seen in these figures that although the interaction in channel 2 is severe, channel 1 is diagonally dominant. Figure 6.5 shows that at most frequencies the system can be scaled by a diagonal post-multiplying matrix in such way that dominance from channel 1 will be transferred to channel 2 to
make the whole system diagonally dominant. At around 330 rad/s channel 1 becomes nearly non-dominant while channel 2 remains non-dominant as this can be seen in the magnitude curve of the Bode plot of figure 6.2. It is at this frequency that the system can not be scaled into diagonal dominance. Although this result could perhaps be induced from figures 6.2, 6.3 and 6.4, it is only the P-F dominance graph of figure 6.5 that quantifies this physical property of the system and conveys precise information as to whether a diagonal scaling compensator can be used to give diagonal dominance. Furthermore if the system can be scaled into dominance, the diagonal compensator which provides the best results will have for its main diagonal the eigenvector that corresponds to the Perron-Frobenius eigenvalue of $\Gamma(j\omega)$ as is shown in chapter 5.

Once the physical meaning of the P-F dominance graph is understood, it becomes clear that it would be considerably easier to design a scalar matrix that will bring the P-F dominance curve to values below 2 at all frequencies, than to design a scalar compensator to make the system diagonally dominant. The design of $K_d(s)$ can therefore be broken down to two stages. In the first stage, if the value of the Perron-Frobenius eigenvalue of $\Gamma(j\omega)$ exceeds the critical value of 2 at some frequencies, a scalar controller is designed to bring the P-F dominance curve down to acceptable levels. In the second stage a scalar or dynamic compensator based on the Perron-Frobenius eigenvectors of $\Gamma(j\omega)$ is designed to make the system diagonally dominant. This is exactly the approach adopted in this project and the design of the actual controllers is described next.

The scalar controller of the first stage can be designed using a number of techniques. Function minimisation methods such as the pseudodiagonalisation and pseudodecoupling could
be used to try to diagonalise the system at a single or range of frequencies. The ALIGN algorithm could also be used to diagonalise the system at one particular frequency, carefully selected, so that the P-F dominance level is brought down to values below 2, over the entire frequency range. The simplest method however for small systems is to use appropriate permutation matrices to perform row or column operations. This is the approach adopted here.

Examining the magnitude curves of the Bode array of figure 6.1, we see that no action need taken for channel one which is already dominant. For the second channel now, if column 1 of the array is subtracted from column 2, a greater quantity, (element \([1,1]\)), will be subtracted from element \([1,2]\), than the quantity, (element \([2,1]\)), subtracted from element \([2,2]\). This operation should improve the interaction characteristics of the system. Since however the element \([1,2]\) is over 3 times smaller than the element \([1,1]\) in low frequencies, column 2 should be multiplied by a number greater than 3.5 before column 1 is subtracted from it. In doing this the second column is prevented from changing sign and interaction is further reduced. The post multiplying matrix which leaves column 1 unaltered and multiplies column 2 by a factor of 3.62, before subtracting column 1 from it is

\[
K_1 = \begin{bmatrix} 1 & -1 \\ 0 & 3.62 \end{bmatrix}
\]

(6.6)

This is the scalar controller of the first stage of the design for diagonal dominance.

The Nyquist array of \(G(s) \cdot K_1\) with row Gershgorin circles superimposed, is shown in figure 6.6. This clearly shows that postmultiplying the system by \(K_1\), has not made it
diagonally dominant. Dominance is lost in channel 1 at high frequencies; channel 2 however is made very dominant at the same frequencies. At low frequencies both channels are dominant. It comes therefore as no surprise that figure 6.7 shows the P-F dominance of \( G(s).K_1 \) to be below 2 at all frequencies. Its worst value of 1.87, still below 2, is at 330 rad/s again.

We can now proceed with designing the diagonal Perron-Frobenius controller \( K_{pf}(s) \), which will make the system \( G(s).K_1 \) diagonally dominant. The overall controller that will make the open-loop system diagonally dominant will then be

\[
K_d(s) = K_1 K_{pf}(s) \tag{6.7}
\]

The second element of the right eigenvector of \( \Gamma(j\omega) \) which corresponds to the Perron-Frobenius eigenvalue is plotted against frequency in figure 6.8. The eigenvector has been normalised so as to have its first element set to 1. The ideal \( K_{pf}(s) \) will be a diagonal matrix of the form

\[
K_{pf} = \begin{bmatrix} 1 & 0 \\ 0 & v(s) \end{bmatrix} \tag{6.8}
\]

where \( v(s) \) is a single-input single-output transfer function whose frequency response magnitude curve fits exactly the curve of figure 6.8. In practice achieving an exact fit is very rarely possible. Care should be taken in this case to achieve a good match at the frequencies where the Perron-Frobenius eigenvalue is nearer to the critical value of 2. In the case of the test rig these frequencies are between 300 and 400 rad/s.
The method used here to calculate the transfer function \( v(s) \) is a least square fitting technique based on the so-called "modified Youle - Walker" equations [21]. This algorithm has been coded into a MATLAB function which is supplied with the signal processing toolbox. This function requires as inputs the magnitude values of the curve to be fitted at several key frequency points as well as the order of the fitting transfer function. By a method of trial and error the best fit was found to be produced by the 4th order transfer function

\[
v(s) = \frac{0.1279s^4 + 146.936s^3 + 6.7447 \times 10^4 s^2 + 1.681 \times 10^7 s + 1.0995 \times 10^9}{s^4 + 545.459s^3 + 1.0649 \times 10^5 s^2 + 8.98169 \times 10^6 s + 2.78409 \times 10^8}
\]

(6.9)

The magnitude curve of the frequency response of \( v(s) \) together with the ideal frequency response magnitude curve as this is defined by plotting the 2nd element of the Perron-Frobenius eigenvector of \( \Gamma(j\omega) \), is given in figure 6.9. It can be seen in this figure that a nearly perfect match is achieved at the difficult frequencies between 300 and 400 rad/s. The fit is not as close, but still good, at the rest of the frequency range where the values of the Perron-Frobenius eigenvalue of the normalised comparison matrix is considerably lower thus permitting some degree of error in the fit. The direct Nyquist array of the compensated transfer function matrix

\[
Q_d(s) = G(s)K_d(s) = G(s)K_1K_p(s)
\]

(6.10)

with row Gershgorin circles, can be seen in figure 6.10. Figure 6.11 gives an expanded view of the region near the
origin. From both figures it can be seen that row diagonal dominance has been achieved over the entire frequency range.

The final stage in the design process is to decide what the system gain will be, and to design the closed loop controllers which will give the required performance to the system. The highest value of system gain which will ensure stability can be assessed by examining the inverse Nyquist array of the already compensated system. Since row diagonal dominance has been achieved in the direct plane, the I.N.A. will be diagonally dominant columnwise. This is shown in the graphs of figures 6.12, 6.13 and 6.14. The two latter graphs are close-ups of the origin of the graph in figure 6.12.

It is desirable as it has already been shown in chapter 5, to have high system gains in the closed-loop system. This will ensure sufficient reduction in interaction levels and the sought after flat frequency responses, while at the same time creating good disturbance rejection characteristics. Since disturbance in this sense incorporates unmodelled system dynamics and time dependency, high system gain will also mean robust system performance. From figure 6.14, however, it can be seen that stability in the closed-loop system can only be guaranteed for system gain up to the value of 0.4. This is unacceptably low. The situation can however be rectified if we notice that it is due to the relatively high magnitude of the system at high frequencies. We need therefore a controller which reduces the high frequency magnitude while at the same time leaving undisturbed the levels of interaction already achieved. Dominance is row-wise in the direct plane and therefore will not be disturbed by any diagonal post-compensator with equal diagonal elements.
A simple lead-lag compensator is used here to accomplish this function:

\[
K_s(s) = \begin{bmatrix}
\frac{0.001s+1}{s+1} & 0 \\
0 & \frac{0.001s+1}{s+1}
\end{bmatrix}
\]  

(6.11)

The inverse Nyquist array of the resulting open-loop system is shown in figures 6.15, 6.16, and 6.17. It is seen in figure 6.16 that the maximum gain which will give a stable closed-loop system is now approximately 80. Here the gain was chosen to be 60. This ensures good system performance over the frequency range of interest while giving sufficient gain margins to provide robust stability in the closed loop system.

The magnitude curves of the Bode array of the closed-loop system can be seen in figure 6.18. Comparing this with the Bode array of the uncompensated open-loop system of figure 6.1, a vast improvement of performance is immediately apparent. The design, however, will be thoroughly evaluated later in this chapter.

6.3 THE CHARACTERISTIC LOCUS DESIGN.

The characteristic locus design process can be broken down to three distinct stages. These three stages, which amount to designing controllers for high, intermediate and low frequency compensation, have already been outlined in chapter 5. However, before embarking on the design process, the characteristic loci of the uncompensated system must be examined. A polar plot of the characteristic loci of the uncompensated rig can be seen in figure 6.19. It is clear that one of the loci has considerably smaller magnitude than
the other. This imbalance is an undesirable and it would be useful if it could be corrected. Recalling Gershgorin's theorem, the characteristic loci can be located, within bounds, to the Nyquist diagrams of the diagonal elements of the transfer function matrix. These bounds shrink as interaction reduces. The characteristic loci will therefore come closer together at the frequencies where the transfer function matrix, modified by a compensator, has little interaction and approximately equal diagonal elements. A compensator which achieves this at low frequencies is given by the permutation matrix.

\[
K_{sc}(s) = \begin{bmatrix} 0.53 & -1 \\ 0 & 3.8 \end{bmatrix}
\] (6.12)

The compensator \(K_{sc}(s)\) was designed following a similar process with the design of the permutation matrix \(K_1\) in the Nyquist array design. The resulting characteristic loci are now balanced as it can be seen in figure 20.

Having balanced the system's characteristic loci, we can now proceed to the first stage of the design process to produce the high frequency compensator. The sole role of this compensator is to reduce interaction at high frequencies. The easiest way to do this is to try to diagonalise the system at some high frequency using the ALIGN algorithm. A compensator which nearly diagonalises the system at 850 rad/s is

\[
K_{hf} = \begin{bmatrix} 1.397 & 1.4282 \\ 0.0699 & 0.3085 \end{bmatrix}
\] (6.13)

The characteristic loci of the of the system after high frequency compensation can be seen in figure 6.21. From
these it can be seen that only locus 1, exhibits satisfactory gain and phase margins, approximately 2.5 and 57° respectively. The second characteristic locus needs intermediate frequency compensation since it has far too great a gain margin and infinite phase margin. The Nyquist diagram of this locus with frequency points indicated on it is shown in figure 6.22. Examining this graph it was decided to use a lag compensator to give 75° of phase lag at 372 rad/s while injecting some gain at the same time to bring the magnitude at the critical point to approximately 0.4 which is the same value as for locus 1. The intermediate frequency compensator should be approximately 1 at the frequency at which the high frequencies compensator is designed, in order to avoid cancelling the effects of the high frequency compensator. The single-input single-output controller which compensates the Nyquist diagram of the second characteristic locus is

\[
 k_{mf}(s) = \frac{0.0032s + 12}{0.0155s + 1} \quad (6.14)
\]

The compensated by \( k_{mf}(s) \) 2nd characteristic locus is shown in figure 6.23. The polar plot of the same locus together with the first characteristic loci of the system with up to high frequency compensation is seen in figure 6.24. It can be seen that the gain margin is 2.5 for both loci now, whereas the phase margin of the second is changed to approximately 95°. These of course would be the results if direct compensation of characteristic loci was possible.
In practice the compensation is achieved through a commutative controller which in this case should be

\[
K_{mf}(s) = A_m \begin{bmatrix} 1 & 0 \\ 0 & k_{mf}(s) \end{bmatrix} B_m
\]  

(6.15)

where \( A_m \) is a real approximation to the generally complex matrix of the eigenvectors of

\[
Q_{mf}(j\omega_k) = G(j\omega_k) K_{sc} K_{hf}, \quad \omega_k = 372 \text{ rad. s}^{-1}
\]

(6.16)

and \( B_m \) a real approximation to its reciprocal eigenvector matrix. The algorithm used to calculate the commutative controller \( K_{mf}(s) \) produces first its state-space model and from it the transfer function matrix of the controller itself. The condition number of the state-space matrix \( A \) for the commutative controller of equation (6.15) (this is the ratio of the largest to the smallest singular value of \( A \)) is very great. This means that the controller will be very sensitive to even small numerical changes in the coefficients. This could provide erroneous results in the simulation of the system's responses but more importantly could cause severe problems in any future implementation of the controller. This situation was rectified by changing equation 6.15 to

\[
K_{mf}(s) = A_m \begin{bmatrix} 0.01s+1 & 0 \\ 0.01s+1 & 0 \\ 0 & k_{mf}(s) \end{bmatrix} B_m
\]

(6.17)

This change does not affect the frequency response of the
intermediate compensator and reduces the condition number dramatically.

The intermediate compensator of equation (6.17) in a transfer function matrix form is given by

\[
K_{mf}(s) = \begin{bmatrix}
0.0007s^2 + 0.365s + 29.231 & 0.0006s^2 - 0.5049s - 56.8065 \\
0.0002s^2 - 0.1698s - 19.1227 & 0.0005s^2 + 0.5853s + 53.8724
\end{bmatrix} \frac{1}{D_{mf}(s)}
\]

(6.18a)

where the common denominator is

\[
D_{mf}(s) = 0.001s^2 + 0.1645s + 6.451
\]

(6.18b)

The polar plot of the actual characteristic loci of the medium frequency compensated system is given in figure 6.25 which closely relates to the predicted loci of figure 6.24.

In the final stage of the design a low frequency commutative controller will be designed. This controller should produce high gains at low frequencies while at the same time it should not disturb the intermediate frequency compensation. This can be done by using a proportional plus integral commutative controller of the form

\[
K_{lf}(s) = \alpha \frac{s}{A_1} \begin{bmatrix}
k_{11} & 0 \\
0 & k_{12}
\end{bmatrix} B_1 + I
\]

(6.19)

The matrices \(A_1\) and \(B_1\) are ALIGN approximations to the complex eigenvector and reciprocal eigenvector matrices of

\[
Q_{lf}(j\omega) = G(j\omega) K_{sc} K_{hf} K_{mf}(j\omega)
\]

(6.20)
where $\omega_1$ is the lowest frequency permitted by the available data. The diagonal elements $k_{11}$ and $k_{12}$ are used to balance the characteristic loci at low frequencies. The most appropriate value for each one of these is therefore the reciprocal of the respective characteristic gain at $\omega_1$ rad/s.

Depending on the value of the integral constant $\alpha$, as the frequency becomes large, the first term in the left hand side of equation (6.19) tends to zero and therefore $K_{1f}(s)$ will tend to the unit matrix $I$. In this way high gains can be achieved at low frequencies, while the gain and phase margins achieved in the previous stage remain largely unaltered. The actual low frequency compensator is

$$K_{1f}(s) = \frac{1}{s} \begin{bmatrix} s+50.8553 & 8.2193 \\ -0.6328 & s+54.5041 \end{bmatrix}$$  \hspace{1cm} (6.21)

The final design open-loop matrix will be

$$Q(s) = G(s) K_{sc} K_{hi} K_{mf}(s) K_{1f}(s)$$  \hspace{1cm} (6.22)

The characteristic loci of the open-loop matrix are shown in figures 6.26 and 6.27. It can be seen from this that the low frequency gain is increased dramatically, ensuring good system performance while there are still adequate gain and phase margins. In particular the gain margins of the two loci are approximately 2 and 2.5 and the phase margins are $65^\circ$ and $45^\circ$ respectively.

The magnitudes of the closed-loop system's Bode array can be seen in figure 6.28. From this graph it can be seen that
flat frequency responses of the diagonal elements have been achieved and at the same time interaction is effectively dealt with for most of the useful frequency range.

6.4 EVALUATION OF THE NYQUIST ARRAY AND CHARACTERISTIC LOCUS DESIGNS.

In chapter 5 it was shown how the largest and smallest singular values of the open-loop system can be used to assess the performance of the closed-loop system. In particular relationships were developed for assessing the closed-loop system's tracking ability, sensitivity, and noise propagation from the singular values of the open-loop system. The same relationships provided the guidelines for a successful controller design. These guidelines and performance indices are summarised below.

For small sensitivity and good tracking of the closed-loop system, the open-loop system need have

$$\sigma[G(j\omega)K(j\omega)] >> 1$$  \hspace{1cm} (6.23)

Whereas for noise filtering the open-loop system must have

$$\bar{\sigma}[G(j\omega)K(j\omega)] << 1$$  \hspace{1cm} (6.24)

Where $G(j\omega)$ is the transfer function matrix of the uncompensated open-loop system, and $K(j\omega)$ denotes the transfer function matrix of the overall controller for either the Nyquist array or the characteristic locus design.

There is an apparent contradiction when comparing equations (6.23) and (6.24). The singular values in both equations however are functions of frequency. Also the
frequencies at which significant disturbance (be it due to system parameter variations or real external disturbance), and measurement noise occur, are in different ranges. Significant disturbance occur at low frequencies whereas measurement noise becomes a problem usually at relatively high frequencies. The conflict between equations (6.23) and (6.24) is therefore resolved by designing for large minimum singular value of the open-loop system at lower frequencies to provide good tracking and sensitivity, and for small maximum singular value at high frequencies to filter out transducer noise.

The graph of the maximum and minimum singular values against frequency of the compensated open-loop system for the Nyquist array design is shown in figure 6.29. The graph for the design based on the characteristic locus is shown in figure 6.30. From both figures we see that indeed high values of the minimum singular value at low frequencies have been achieved and at the same at frequencies beyond the system bandwidth (approximately 150 and 300 rad/s respectively) the maximum singular value rapidly becomes small to filter out measurement noise. Also, it is seen that the difference between the minimum and the maximum singular values remain small at all frequencies for both designs. Recalling that the maximum and minimum singular values provide an upper and a lower bound to the system's characteristic loci, it can be deduced that there are no significant differences between the performance of individual channels. Also, the increase in the slope of the singular values in figures 6.29 and 6.30 at frequencies beyond the bandwidth assists the increase of bandwidth without sacrificing the good noise rejection properties of the design.
In chapter five it was shown that the system would remain stable under disturbances smaller than \( \delta_m(j\omega) \), if

\[
\sigma[T(j\omega)] < \frac{1}{\delta_m(j\omega)} \quad (6.25)
\]

Therefore the maximum allowable perturbation for stability is given by

\[
\delta_m(j\omega) = \frac{1}{\sigma[T(j\omega)]} \quad (6.26)
\]

The graphs of the singular values of the closed-loop system for the Nyquist array and the characteristic locus designs are seen in figures 6.31 and 6.32 respectively. From these graphs it is seen that \( \delta_m(j\omega) \) is about 1 (or 100% perturbation) at low frequencies for both designs, growing at the bandwidth frequencies to approximately 0.56 and 0.48 (or 56% and 48%) for the Nyquist array and characteristic locus designs respectively. It would take a massive disturbance to make the system unstable at high frequencies.

To compare the performance achieved by the two designs consider the figures 6.18, 6.28, 6.29, 6.30, 6.31 and 6.32. From these we see that robust performance and stability have been achieved by both designs. Additionally the design requirement for flat frequency responses of the diagonal elements of the closed-loop transfer function matrix and minimum interaction have been achieved by both designs. The Nyquist array design provides better overall interaction characteristics with levels never above 30% in the useful frequency range and less than 10% below 80 rad/s. This is a marked improvement from the uncompensated system where
interaction exceeded 100% in the same range. The characteristic locus design also reduces interaction considerably to less than 10% up to 150 rad/s and remains below 15% up to 300 rad/s in the second channel. In the first channel, however, interaction is kept below 10% only up to 40 rad/s grows to 30% at 80 rad/s and finally exceeds 100% at 200 rad/s. These differences between the two designs are accounted for firstly by the diagonal dominance requirement of the Nyquist array design method which ensures that interaction levels never exceed 100%. Secondly the dominance sharing concept of the Perron-Frobenius controller gives approximately the same interaction levels in both channels.

The diagonal dominance requirement of the Nyquist array in effect forces the designer to work with bounds for the characteristic loci, given by the Gershgorin bands, rather than the characteristic loci themselves. This provides for more robust stability of the system, 56% against 48% for the characteristic locus design. On the other hand the conservative nature of the diagonal dominance concept gives lower limits to the system gain across the frequency range than the characteristic locus method. This accounts for the lower bandwidth of the Nyquist array design as well as the lower interaction levels of the characteristic locus design at low frequencies.

The interaction levels could be further reduced and the bandwidth extended to higher frequencies by using a pre-filter, $P(s)$ to filter the reference signal $r(s)$. Ideally this prefilter should approximate the inverse of the closed-loop transfer function matrix in the frequency range of interest. The philosophy of this approach does not differ from this of the RPC method described in chapter 2. The pre-filter however in this case will be combined with a closed-loop scheme which will ensure robust performance and stability.
Figure 6.1. Bode array of experimental frequency response of uncompensated system.

Figure 6.2. Magnitude graph of the Bode plot of the uncompensated system.
Figure 6.3. DIRECT NYQUIST ARRAY OF UNCOMPENSATED SYSTEM
WITH COLUMN GERSHGORIN BANDS SUPERIMPOSED.

Figure 6.4. DIRECT NYQUIST ARRAY OF UNCOMPENSATED
SYSTEM WITH ROW GERSHGORIN BANDS
SUPERIMPOSED.
Figure 6.5. **P-F Dominance graph of uncompensated system.**

Figure 6.6. **Direct Nyquist array of \( G(s)K_1 \), with row Gershgorin bands superimposed.**
**Figure 6.7.** P-F Dominance Graph of $G(s)K_L$

**Figure 6.8.** Second element of the normalised eigenvector of $G_{inv}$. 

P-F Dominance graph

FREQUENCY (rad/s)

MAG dB

FREQUENCY
Figure 6.9. Magnitude of actual and ideal p-e compensators against frequency.

Figure 6.10. Direct Nyquist array of $Q(s) = G(s)K_1K_2(s)$, with row Gershgorin bands superimposed.
Figure 6.11. Direct Nyquist array of $Q(s)=G(s)K_{s}K_{p_{c}}(s)$, with row Gershgorin bands superimposed. Magnified at the origin.

Figure 6.12. I.N.A. of $Q(s)$ with column Gershgorin bands.
Figure 6.13. I.N.A. of $Q(s)$ with column Gershgorin bands magnified at the origin.

Figure 6.14. I.N.A. of $Q(s)$ with column Gershgorin bands with further magnification at the origin.
Figure 6.15. I.N.A. of $Q(s)K_s(s)$ with Column Gershgorin Bands.
Figure 6.16. L.N.A. of $Q(s)K_s(s)$ with column Gershgorin bands magnified at the origin.

Figure 6.17. L.N.A. of $Q(s)K_s(s)$ with column Gershgorin bands with further magnification at the origin.
Figure 6.18. Bode array of closed-loop system with Nyquist array based compensator.
Figure 6.19. CHARACTERISTIC LOCI OF UNCOMPENSATED SYSTEM.

Figure 6.20. CHARACTERISTIC LOCI OF G(u)YK_sc.
Figure 6.21. Characteristic loci of 
\( G(s) K_{se} K_{sf} \).

Figure 6.22. Second characteristic locus 
of \( G(s) K_{1} K_{sf} \).
Figure 6.23. CHARACTERISTIC LOCUS OF $G(s)K_S V_{Uf}$ WITH IDEAL PHASE COMPENSATION.

Figure 6.24. CHARACTERISTIC LOCI OF $G(s)K_S V_{Uf}$ WITH THEORETICAL COMPENSATION.
Figure 6.25. CHARACTERISTIC LOCI OF SYSTEM AFTER INTERMEDIATE FREQUENCY COMPENSATION WITH COMMUTATIVE CONTROLLER.

Figure 6.26. CHARACTERISTIC LOCI OF OPEN-LOOP SYSTEM WITH "FINAL DESIGN" COMPENSATOR.
Figure 6.27. Characteristic loci of compensated open-loop system, magnified at the origin.

Figure 6.28. Body array of closed-loop system, compensated with characteristic locus based controller.
Figure 6.29. **Principal Gains of Open-Loop System with Nyquist Array Based Controller.**

Figure 6.30. **Principal Gains of Open-Loop System with Characteristic-Locus Based Controller.**
Figure 6.31. Principal gains of closed-loop system with Nyquist array based controller.

Figure 6.32. Principal gains of closed-loop system with characteristic locus based controller.
CHAPTER 7

SOFTWARE DEVELOPMENT

7.1 INTRODUCTION

The analysis and design of multivariable control systems involves a computational burden which makes computer assistance essential. Even until a few years back this assistance was provided by CAD suites mounted on main-frame computers. This coupled with the lack of a user friendly environment meant that not only they have not been widely accessible to the control engineering community, but also when access was provided their use was often problematic. These problems provided the motivation to develop a CAD program to run on small personal computers.

The first attempt to write such a program developed the CADMVS version for the BBC microcomputer[25]. However even with memory expansion this computer was only able to support software for 2-input 2-output systems and for a limited range of facilities. Additionally the constraint in code size degrade the structure of the program and made it less amenable to modification. These considerations led to the development of CADMVS2 [26], to run on MS-DOS based machines. This package which is designed to have a modular structure to facilitate future expansions, will be described in some detail in this chapter.

In the last three years PC-based CAD packages for the analysis and design of multivariable systems have become commercially available. The most widely used of them is MATLAB with its "control", "Robust control", and "Multivariable frequency domain" toolboxes. This package provides a flexible and user friendly environment combined with very reliable numerical algorithms. New facilities can be easily added to the
original program, by user provided subroutines.

The facilities, the ease of use, and the ease with which extra facilities can be added to this package supersedes these of CADMVS2. The focus was therefore moved from developing further the CADMVS2 to tailoring MATLAB toolboxes to the needs of our project. A number of routines were thus developed for use with MATLAB which are briefly described towards the end of this chapter.

**7.2 DESCRIPTION OF CADMVS2**

The package consists of two distinct programs although this is transparent to the user; one operating in the time domain and the other in the frequency domain. The two programs support different system models and CAD facilities, although some CAD options such as open and closed-loop simulation are shared by both programs.

The time-domain program supports state-space models of the form

\[
\dot{x} = Ax + Bu \\
y = Cx
\]  

(7.1)

with either state or output feedback.

The frequency domain program supports the control system configuration seen in the block diagram of figure 7.1. The forward path transfer function matrix of this block diagram is

\[
Q(s) = LG(s)K
\]  

(7.2)

where K and L are pre- and post compensator matrices respectively. The matrix F in the block diagram of figure 7.1 is the matrix of the feedback gains.

The maximum number of inputs and outputs in both state-space
and transfer function matrix descriptions is 5 whereas the maximum system order is 10. These values however are not imposed by limitations of the numerical algorithms used. They were chosen rather arbitrarily to provide a reasonable figure for the minimum memory requirement of the package. They could therefore be easily changed to higher values if memory size does dictate otherwise.

CADMVS2 runs on IBM PC/XT/AT or compatibles with an EGA graphics card and on Research Machines Nimbus computers running MS-DOS version 2.0 or later. At least 320K of RAM is needed in order to run the package satisfactorily. Hard copies can be obtained on any EPSON or EPSON compatible printers.

7.2.1 CAD options.

(a) Transformations

The transfer function matrix may be derived from a state-space description. Additionally for the state-space form the closed-loop model can be obtained for either state or output feedback. The transformation from the transfer function matrix to state-space model is also available. This transformation however results in observable but not controllable state-space models.

(b) Simulation

Time responses may be simulated for systems described either in a state-space or in a transfer function matrix form for both open and closed-loop configurations. It is worth noting here that a number of different time-functions can be used as inputs. The range of available inputs include impulse, step, ramp, sinusoidal and parabolic functions.

(c) Design methods

Time domain as well as frequency-domain techniques are offered. For state-space models, pole allocation is achieved by dyadic state-feedback controllers[27]. Note here that a controllability test on the resulting equivalent single-input system is automatically performed before the feedback controller is designed. The controllability test can be performed
irrespective of whether this particular design method is used.

For systems described by a transfer function matrix, the Multivariable Nyquist Array design methods are available. Either Gershgorin or Ostrowski circles can be superimposed on direct or inverse Nyquist plots. Diagonal dominance can be achieved by Perron-Frobenius based pre- or post-compensators or similarity transformations. Graphical transformations between INA and DNA plots can also be performed[28].

7.2.2 User Interface

The package is command driven but limited on screen help facilities also exist. Each command line can be followed by the command options. If some or all of the options are not specified, the default settings come into operation. These are read from a preset file. The file can be formed by either running a separate program, or by invoking the appropriate command while running the package. Transfer function or state-space models as well as frequency and time response data can be saved into files and retrieved when needed.

7.2.3 Expandability

Special attention has been given to the structure of the program, so that new facilities and options can be added with the minimum effort. The programming language used is mainly FORTRAN 77, although parts of the program have been written in Pascal. Standard versions of these languages are used so that the software can be transported easily to any computer having FORTRAN and Pascal compilers and graphics capabilities.

7.3 NEW MATLAB FUNCTIONS

It has already been mentioned that a considerable part of this project was dedicated to developing CADMVS2. The arrival, however, of the Multivariable Frequency Domain (MFD) toolbox for
the MATLAB package which offer a wider range of CAD facilities and easy addition of new ones, meant that further development of CADMVS2 was abandoned.

There are however certain disadvantages associated with this package too. The main one being the requirement that transfer function matrices must always be specified in terms of a numerator polynomial matrix and a common denominator polynomial. Calculating the common denominator of a large and/or high order transfer function matrix involves a lot of effort and is bound to introduce numerical errors. To overcome this problem a whole suit of MATLAB function was written which will accept the standard non-common denominator configuration of transfer function matrix and interface it to MFD. These functions are listed in appendix 2.

A number of other MATLAB functions were written to improve the graphical display facilities of MFD, and aid with various design techniques. The listings of these function can be found in appendix 2 as well.
Figure 7.1  Block diagram supported by the frequency domain program of CADMVS2
CHAPTER 8

CONCLUSIONS

The characteristics of the ideal control system for multi-channel dynamic structural testing have been examined and current tests practices have been analysed. It has been found that the system which will provide the required accuracy for dynamic testing must have little or no interaction between the different input-output pairs and frequency response with constant magnitude over the useful frequency range. Furthermore, the system which will exhibit such behaviour will be equally suited for tests with sinusoidal as well as random inputs. It was also shown that robust performance is desirable which can only be achieved by closed-loop control resulting from a multivariable system approach.

To test the applicability of multivariable control theory to dynamic structural testing, a purpose built experimental test rig was used. This consists of a cantilever beam which can be excited into vibration by two electrodynamic shakers. Displacement is measured at the points where the vibrators are attached to the beam thus forming a two-input two-output multivariable system. The rig is configured to provide the worst possible coupling between different inputs and outputs.

Analytical modelling of the rig has been achieved by modelling the beam as a continuous elastic body and combining it with models of the dynamics of the vibrators. The modelling is done in such a flexible way as to produce models of the rig with a varying number of vibrators and output sensors. The position of both the vibrators and transducers can also be varied along the beam. This led to
open and closed-loop simulation of the beam by MATLAB based computer programs which utilise all the features of the modelling method described above. Any time series can be used as inputs and the output variable can be selected amongst displacement, velocity force or current through the vibrators armature circuit. Analytical modelling of the rig was undertaken in order to provide a better insight into the systems dynamics and to simulate the rig as a larger system where more than two vibrators and two output sensors are used. The accuracy of these models has been validated by comparing frequency responses derived from it to experimentally obtained ones.

The multivariable methods used to design controllers for the experimental test rig are the characteristic locus and the Nyquist array extended by the use of the Perron-Frobenius theory of positive matrices. The primary data for the designs was provided by experimental frequency responses obtained from the test rig. It was shown that no analytical modelling is required in order to use multivariable control methods in dynamic structural testing. It was seen from the experimental frequency responses that interaction levels in the uncompensated system exceeded 100% in the useful frequency range. Simulated frequency responses of the Nyquist array design reduce this figure to 10% below 80 rad/s while it is never exceeding 30% in frequencies up to the system's bandwidth. The characteristic locus design exhibited even better levels of interaction in the first channel but in the second channel interaction was kept below 10% only up to 40 rad/s. During the design process emphasis was given to the production of controllers which would provide robust performance and stability and this has been achieved. In particular the system could only be made unstable by a perturbation of 100% in low frequencies for both the characteristic locus and the Nyquist array designs, whereas perturbations of 56% and 48% respectively at bandwidth frequencies would cause instability. It would take
a massive disturbance at high frequencies to make the system unstable. Therefore multivariable controllers provide accuracy as well as robust performance and stability.

In the initial stages of the project a CAD program for multivariable control was developed. Considerable effort was given to producing an efficient user interface and an easily expanding programming structure. However the development of the package was later abandoned since it could not compete with the range of options other commercial packages, such as the Multivariable Frequency Domain (MFD) toolbox for MATLAB, could offer.
FUTURE WORK

Future work on this project should concentrate mainly on two areas; the implementation of the designed controllers and the use of the developed model for simulation studies.

The implementation of the controllers should, preferably, be done digitally. In order to achieve the performance predicted for the continuous controllers a high sampling frequency must be used. Experience suggests that a sampling frequency of at least 6 to 10 times the system bandwidth must be chosen. The inclusion of anti-aliasing filters should also be considered.

The full potential of the model of the rig developed in chapter 4, could only be realised in simulation studies which would utilise varying numbers of shakers and output sensors, while at the same time varying their relative positions along the beam. This way the problem of designing controllers for larger structural test systems could be investigated.
REFERENCES


35. Irwin, C.A.K. "Static and repetitive loading tests on full-scale pre-stressed concrete bridge beams," TRRL report LR802, Transport and Road Research Laboratory, Crowthorne,1977

36. Howells, H. and Raithb, K.D. " Static and repeated loading tests on lightweight pre-stressed concrete bridge beams," TRRL report LR804, Transport and Road Research Laboratory, Crowthorne,1977


APPENDIX 1

MATLAB FUNCTIONS FOR THE SIMULATION OF THE EXPERIMENTAL RIG
% A script file that produces a mathematical modelling for
% the rig with output a user defined variable (current, force
% or displacement). The necessary data can be entered
% interactively from the keyboard, or loaded from the file
% 'rigdata.mat' which can be read from the hard or the floppy
% disk.
%
% % %% ----------------------------------
% % %% Input Data % % % % % % % %
% % %% ----------------------------------

mess1 = 'Interactively from keyboard';
mess2 = 'From the data file rigpars.mat, in hard disk';
mess3 = 'From the data file rigpars.mat, in floppy disk';
k = menu(' Input Data : ',mess1,mess2,mess3);
if k==1
    keybrdat
elseif k==2
    load rigpars
elseif k==3
    load a:rigpars
end
%
% % %% Get model for beam-vibrator subsystem % % % %

EI = Eb*Eb;
b = bi*L/L;
vb_dat = [xv,mv,kv];
bm_dat = [n,El,mb,L];
fprintf('

... Working .... Please Wait ... 

');
[Abv,Bbv,Cbv,Dbv] = bm_vbmdl(bi,Ai,bm_dat,vb_dat,xl,zeta);
%
% % % % Get model for the whole rig % % % %
%
% % First define the blocks of the block diagram
mvnblks = [1 2 3 5];
ssnblks = 4,
num1 = eye(1); comden1 = 1;
num2 = inv(diag(Ra)); comden2 = 1;
num3 = diag(kF); comden3 = 1;
a4 = Abv; b4 = Bbv; c4 = Cbv; d4 = Dbv;
num5 = []; T_temp = 1e-6;
for k = 1:1
    num_r = zeros(1,2*1);
    num_r(2*k-1) = kB(k);
    num5 = [num5;num_r];
end
comden5 = [T_temp 1];
verbose = 1;
mvblkbi;

% Specify the interconnections of the different blocks

O = [1 0 0
     2 1 -5
     3 2 0
     4 3 0
     5 4 0];
lu = 1;

% Get the output variable specified and produce state-space model
mess1 = 'Displacement'; mess2 = 'force'; mess3 = 'current';
k = menu('Simulated Output : ',mess1,mess2,mess3);
if k==1
    ly = 4; [Ar,Br,Cr,Dr] = mvcon(a,b,c,d,O,lu,ly,sz);
    strk = 'displacement.';
elseif k==2
    ly = 3; [Af,Bf,Cf,Df] = mvcon(a,b,c,d,O,lu,ly,sz);
    strk = 'force.';
elseif k==3
    ly = 2; [Acur,Bcur,Ccur,Dcur] = mvcon(a,b,c,d,O,lu,ly,sz);
    strk = 'current.';

A1.2
end

% ------- Produce simulation if required -------

mess1='System input and time vectors (U_sys,t_sys) exist.';
mess2='Define system inputs and time vector interactively.';
mess3 = 'Quit.';
k1 = menu(' Time Response ',mess1,mess2,mess3);
if k1 == 2
    U_sys = input('Give the system inputs : ');
    t_sys = input('Give the time vector : ');
    [n_U,m_U] = size(U_sys);
    if m_U ~= 1
        U_sys = U_sys';
    end
end
if (k1 == 1) || (k1 == 2)
    if k==1
        Yd = lsim(Ar,Br,Cr,Dr,U_sys,t_sys);
        plot(t_sys,Yd),grid,ylabel('Displacement'),xlabel('time'),title('Displacement Time Response');
    elseif k==2
        Yf = lsim(Af,Bf,Cf,Df,U_sys,t_sys);
        plot(t_sys,Yf),grid,ylabel('Force'),xlabel('time'),title('Load Time Response');
    elseif k==3
        Your = lsim(Aour,Bour,Cour,Dour,U_sys,t_sys);
        plot(t_sys,Your),grid,ylabel('Current'),xlabel('time'),title('Current Time Response');
end
end
function [A,B,C,D] = bm_vbmdl(bi,Ai,bm_dat,vb_dat,xl,zeta)
% [Abv,Bbv,Cbv,Dbv] = bm_vbmdl(bi,Ai,bm_dat,vb_dat,xl,zeta)
%
% Function to calculate the state-space model of the
% Beam-Vibrator subsystem.
% *** Input Variables ***
% bi : Is the vector bil/l, l the length of the beam.
% Ai : The vector of the Ai coefficients needed in the
% calculation of the mode shape functions phi(x).
% bm_dat : Data relating to beam : bm_dat = [n,EL,mb,L]',
% where n is the number of modes to be considered,
% EL is the flexural stiffness of the beam. The mass
% and length of the beam are mb and L respectively.
% vb_dat : Data relating to vibrators. vb_dat = [xv,mv,kv],
% where xv is the vector of the distances of the
% vibrators from clamped end. mv and kv are the
% vectors of the vibrators' armature masses and
% stiffnesses.
% xl : The vector of the distances of the LVDTs from the
% clamped end;
% zeta : The damping ratio.
%
% The function outputs are the beam-vibrator subsystem's
% state-space matrices.
%
% ----- STEPS 1 & 2 ----- %
% Retrieve data for beam and calculate Omega.
% L = bm_dat(4); mpl = bm_dat(3)/L; n = bm_dat(1);
% EI = bm_dat(2);
% bi = bi(1:n);
% Omega = diag((bi.^4)*EI/mpl);
%
% ----- STEP 3 ----- %
% Calculate the matrix of the generalized masses M.
%
\[ M = []; \]
\[
\text{for } k = 1 : n \\
\quad b = bi(k)*L; \\
\quad a_i = Ai(k); \\
\quad m_{ii} = 4*b*\sin(2*b)+2*cosh(b)*\sinh(b)-4*cosh(b)*\sin(b) - 4*\sinh(b)*\cos(b); \\
\quad m_{ii} = m_{ii} + a_i^2/4; \\
\quad m_{ii} = m_{ii} + a_i*(\sin(b)*\sin(b)+\sinh(b)*\sinh(b)-2*\sin(b)*\sin(b)); \\
\quad m_{ii} = m_{ii} + (2*cosh(b)*\sinh(b)+4*\sinh(b)*\cos(b)-4*cosh(b)*\sin(b)) - \sin(2*b))/4; \\
\quad m = m\cdot m_{ii}/bi(k); \\
\quad M = [M;m_{ii}]; \\
\text{end} \\
\text{M = diag(M);} \\
\%
\%
\% ----- STEP 4 ----- 
\%
\% Calculate the matrices \text{Phi}_F and \text{Phi}_Y 
\%
\text{xv = vb_dat(:,1);} \\
\text{l = length(xv);} \\
\text{m = length(xl);} \\
\text{Phi_F = [];} \\
\text{Phi_Y = [];} \\
\text{for } k = 1:n \\
\quad b = bi(k); \\
\quad a = Ai(k); \\
\quad Phi_F_row = []; \\
\quad Phi_Y_row = []; \\
\quad for kl = 1:1 \\
\quad \quad phi_{el} = \text{phix}(a,b,xv(kl)); \\
\quad \quad Phi_F_row = [Phi_F_row,phi_{el}]; \\
\quad end \\
\quad for km = 1:m \\
\quad \quad phi_{el} = \text{phix}(a,b,xl(km)); \\
\quad \quad Phi_Y_row = [Phi_Y_row,phi_{el}]; \\
\quad end \\
\text{Phi_F = [Phi_F,Phi_F_row];} \\
\text{Phi_Y = [Phi_Y,Phi_Y_row];} \\
\]
end
Phi_F = [Phi_F;Phi_F_row];
Phi_Y = [Phi_Y,Phi_Y_col];
end

% % ------ STEP 5 ------
% Calculate the matrices Ms, Ks Pq and Pf
% Ms = vb_dat(:,2);Ms = diag(Ms);
Ks = vb_dat(:,3);Ks = diag(Ks);
Pf = inv(eye(n)+inv(M)*Phi_F*Ms*Phi_F');
Pq = Pf*(Omega + inv(M)*Phi_F*Ks*Phi_F');
Pd = 2*Pf*zeta*sqrt(Omega);
Pf = Pf*inv(M)*Phi_F;
%
% % ------ STEP 6 ------
% Calculate the matrices A,B,C and D;
% A = [zeros(Pq),eye(Pq);-Pq,-Pd];
B = [zeros(Pf);Pf];
C = [Phi_Y, zeros(Phi_Y)];
D = zeros(m,1);
% ------- END -------
This is a script file that enters interactively the rig data needed by the RigSimul program.

```matlab
fprintf('* * * Rig Data Input Routine * * *
');
fprintf('
');
fprintf('* * * Input vibrator parameters * * *
');
l = input('Number of vibrators : ');
mv = input('Vibrator armature masses : ');
kv = input('Vibrator stiffness coefficients : ');
kF = input('Vibrator force coefficients : ');
Ra = input('Vibrator armature resistances : ');
xv = input('Vibrator distances from clamped end : ');
La = input('Vibrator armature coil inductances : ');

fprintf('
');
fprintf('* * * Input Beam Parameters * * *
');
Eb = input('Young's modulus : ');
Ib = input('Second moments of area : ');
mb = input('mass of beam : ');
L = input('length of beam : ');
n = input('number of natural modes to be considered : ');

fprintf('
');
fprintf('* * * Transducer Parameters * * *
');
xl = input('Distances of transducers from clamped end : ');
ml = input('Masses of moving parts of transducers : ');
Ai = [-1.362220574E+000
    -9.818675391E+001
    -1.000761053E+000
    -9.999644787E+000
    -1.000001449E+000
    -9.999993434E-001
    -1.000000027E+000
    -9.999998829E-001
    -1.000000050E+000
    -9.999999782E-001 ];
```
\[ \begin{align*} 
\text{biL} &= \begin{bmatrix} 1.87510406871196e+000 \\
4.69409113297418e+000 \\
7.85475743823761e+000 \\
1.09955407348755e+001 \\
1.41371683910465e+001 \\
1.72787595320882e+001 \\
2.04203522510413e+001 \\
2.3561949018065e+001 \\
2.6703537555183e+001 \\
2.9845130291028e+001 \\ 
\end{bmatrix}; 
\end{align*} \]
% A script file that creates the data file rigpars needed by
% the rigsimul program. It generates the saves the variables :
% n,m,l : the number of modes of the beam to be considered,
% and the number of LVDT's and vibrators respectively.
% mv : The vector with the masses of the vibrator armature.
% kv : The vector with the vibrator stiffnesses
% kF : The vector with the vibrator force coefficients
% Ra : The vector with the vibrator armature resistances
% xv : The vector with the vibrator distances from the
% beam's clamped end.
% Eb : The beams Young's modulus
% Ib : The beam's second moments of area
% mb : The beam's mass
% L : The beam's length
% biL : The vector of the first 10 solutions of the
% frequency equation for cantilever beams.
% Ai : The coefficients of the shape functions phi(x) for
% the first 10 modes.

% ------ Define beam parameters ------

n = 3; m = 2; l = 2;
mb = 1; L = 0.65; Eb = 205e09; Ib = 1.667e-09;
Ai = [-1.36222055748513e+000
      -9.81867539174730e-001
      -1.00077610535498e+000
      -9.99966447874070e-001
      -1.00000144989345e+000
      -9.99999937344375e-001
      -1.000000270760e+000
      -9.999999982994e-001
      -1.000000000506e+000
      -9.999999999782e-001 ];
biL = [ 1.87510406871196e+000
4.69409113297418e+000
7.85475743823761e+000
1.0995407348755e+001
1.41371683910465e+001
1.72787595320882e+001
2.04203522510413e+001
2.35619449018065e+001
2.67035375555183e+001
2.98451302091028e+001];

% ----- Define the vibrators' parameters ----- 

mv = 0.229*[1 1]';
kv = 1.41e04*[1 1]';
kF = 7.7*[1 1]';
Ra = 0.53*[1 1]';
xv = [0.65/2 0.65]';

% ----- LVDT locations ----- 

xl = xv;

save rigpars Ai L kF m n Eb Ra kv mb xl Ib biL l mv xv
clear Ai L kF m n Eb Ra kv mb xl Ib biL l mv xv
APPENDIX 2

MISCELLANEOUS MATLAB FUNCTIONS
function [A,B,C,D] = ncdtf2ss(NUM,DEN,Nio)

% [A,B,C,D] = ncdtf2ss(NUM,DEN,Nio)
%
% NCDTF2SS converts the transfer function matrix stored in
% NUM and DEN as described below, into the state - space
% system [A,B,C,D]. A minimum order state - space realization
% is calculated. However in some instances uncontrollable
% and/or unobservable modes are present in the state - space
% model returned.
% NUM : is a matrix containing the polynomial numerators
% of the T.F.M. The polynomial numerators are stored
% in NUM in the following way :
% the (i,j) element of the T.F.M. is stored in the
% m*(i-1)+j row, where m is the number of inputs.
% Therefore for a 2x2 T.F.M., the element (1,1) is
% stored in row 1, the element (1,2) in row 2, the
% element (2,1) in row 3, and the element (2,2) in
% row 4. The number of columns in NUM is given by
% the maximum degree of the polynomial numerators of
% the T.F.M.
% DEN : is the matrix containing the polynomial
% denominators of the T.F.M. The denominators are
% stored in DEN in the same as the way the
% numerators are stored in NUM.
% Nio : is a element vector. Nio(1) is the number of
% system I/Ps and Nio(2) is the number of system
% outputs.
% A,B,C,D : The system state - space matrices.

[nmxl,mno] = size(NUM);
[dmxl,mdo] = size(DEN);
C = [];
D = [];
B = [];
m = Nio(1);
1 = Nio(2);

% Handle constant case

if mno == 1 & mdo == 1
    A = 0;
    B = zeros(1,m);
    C = zeros(1,1);
    for i = 1:1
        for j = 1:m
            D(i,j) = NUM(m*(i-1)+j)/DEN(m*(i-1)+j);
        end % for j = 1:m
    end % for i = 1:1
    return
end % if mno == 1 & mdo == 1

orda = mdo - 1;

Ac = 0;
for i = 1:1
    for j = 1:m
        Ac = Ac + 1;
        index = m*(i-1) + j;
        gn = NUM(index,:);
        gd = DEN(index,:);
        [a,b,c,d] = tf2ss(gn,gd);
        if length(a) < orda
            nzs = orda - length(a);
            temp = a;
            a = zeros(orda,orda);
            a(nzs+1:orda,nzs+1:orda) = temp;
            b = [zeros(nzs,1);b] ;
            c = [zeros(1,nzs),c] ;
        end % if length(a) < orda
        if Ac == 1
            A = [a,zeros(orda,(m*1-Ac)*orda)];
        else
            A=[A;zeros(orda,(Ac-1)*orda),a,zeros(orda,(m*1-Ac)*orda)];
        end % if Ac == 1

A2.2
% Calculate the B matrix.
    if j == 1
        B = [B;b,zeros(orda,m-1)];
    else
        B = [B;zeros(orda,j-1),b,zeros(orda,m-j)];
    end  % j == 1

% Calculate the ith row of the C matrix.
    if (j == 1) & (i== 1)
        Crow = c;
    elseif j == 1
        Crow = [zeros(1,(i-1)*m*orda),c];
    elseif j < m
        Crow = [Crow,c];
    else
        Crow = [Crow,c,zeros(1,(nmxl-m*i)*orda)];
    end  % if ( j == 1 ) & ( i == 1 )

% Calculate the ith row of the D matrix.
    if j == 1
        Drow = d;
    else
        Drow = [Drow,d];
    end  % if j == 1
end  % for j = 1:m
C = [C,Crow];
D = [D,Drow];
end  % for i = 1 : 1
[a,b,c,d,nrs] = ssrnreal(A,B,C,D);
if nrs == 0
    A = a;
    B = b;
    C = c;
D = d;
end % if nrs ~= 0
end   % ncdtf2ss
function Gjw = ncd2fr(Gnum,Gden,Nio,w)

% GJW = ncd2fr(GNUM,GDEN,NIO,W)
% Function to calculate the multivariable frequency response
% matrix (compatible with the MFD toolbox), given the
% non-common denominator Transfer Function Matrix stored in
% the vectors GNUM GDEN and NIO.
%
% GNUM : is a matrix containing the polynomial numerators
% of the T.F.M. The polynomial numerators are stored
% in GNUM in the following way:
% the (i,j) element of the T.F.M. is stored in the
% m*(i-1)+j row, where m is the number of inputs. So
% for a 2x2 T.F.M., the element (1,1) is stored in
% row 1, the element (1,2) in row 2, the element
% (2,1) in row 3, and the element (2,2) in row 4.
% The number of columns in Gnum is given by the
% maximum degree of the polynomial numerators of the
% transfer function matrix.
% GDEN : is the matrix containing the polynomial
% denominators of the T.F.M.. The denominators are
% stored in GDEN in the same way the numerators are
% stored in GNUM.
% NIO : is a element vector. NIO(1) is the number of
% system I/Ps and NIO(2) is the number of system
% O/Ps.
% W : is the frequency vector (in radians).
% GJW : is the 'MFD format', Frequency responce matrix.

[dlxm,ndo] = size(Gden);  
[dlxm,nnc] = size(Gnum);  
ni = Nio(1);  
no = Nio(2);  
nw = length(w);  
imj = sqrt(-1);  
for k = 1:nw
\[ jw = w(k)imj; \]
for i = 1:no
  for j = 1:ni
    gn = Gnum(ni*(i-1)+j,:);
    gd = Gden(no*(i-1)+j,:);
    Gjw(i+no*(k-1),j) = polyval(gn,jw)/polyval(gd,jw);
  end
end
end
function [A,B,C,D] = ncdfb(NUM, DEN, FNUM, FDEN, NIO)

% [A,B,C,D] = ncdfb(NUM, DEN, FNUM, FDEN, NIO).
% ncdfb combines the two non-common denominator systems in a
% closed loop negative feedback system, whose state-space
% model is given by the [A,B,C,D] matrices. If the function is
% invoked as: [MVNUM,Comden] = ncdfb(NUM, DEN, FNUM, FDEN, NIO)
% the closed loop negative feedback system is defined by an
% MFD format transfer function matrix (see MFD Toolbox manual).
% NUM : Is the vector that stores the numerators of the
% N.C.D. (non-common denominator) forward path transfer
% function matrix.
% DEN : Is the vector that stores the denominators of NCD
% forward path transfer function matrix.
% FNUM : Is the vector that stores the numerators of the
% N.C.D. feedback transfer function matrix.
% FDEN : Is the vector that stores the denominators of NCD
% feedback transfer function matrix.
% NIO : A two element row vector giving the number of inputs
% ( NIO(1) ) and the number of outputs ( NIO(2) ) to
% the system.

[a,b,c,d] = ncdtf2ss(NUM,DEN,NIO);
FNIO = [ NIO(2), NIO(1) ];
[fa,fb,fc,fd] = ncdtf2ss(FNUM,FDEN,FNIO);
if nargout == 4
    [A,B,C,D] = mvfb(a,b,c,d,fa,fb,fc,fd);
else
    [a,b,c,d] = mvfb(a,b,c,d,fa,fb,fc,fd);
    [A,B] = mvss2tf(a,b,c,d);
end % nargout == 4
end % ncdfb
function [A,B,C,D] = ncdpar(NUM1, DEN1, NUM2, DEN2, NIO)

% [A,B,C,D] = ncdpar(NUM1, DEN1, NUM2, DEN2, NIO) forms a true
% parallel connection of the two N.C.D. systems described by
% the vectors NUM1, DEN1, NUM2, DEN2 and have NIO(1) inputs and
% NIO(2) outputs. That is both systems are fed with the same
% inputs and their outputs are summed.
% If the function is invoked as :
% [MVNUM,Comden] = ncdpar(NUM1,DEN1,NUM2,DEN2,NIO)
% the parallel system is defined by an MFD format transfer
% function matrix (see the MFD Toolbox manual).
% NUM1, NUM2 : Are the vectors that stores the numerators of
% the two parallel N.C.D. (non-common
% denominator) transfer function matrices.
% DEN1,DEN2 : Are the vectors that stores the denominators of
% the parallel N.C.D. Transfer function matrices.
% NIO : Is the two element row vector giving the number
% of system inputs { NIO(1) } and the number of
% outputs { NIO(2) }.

[a1,b1,c1,d1] = ncdtf2ss(NUM1,DEN1,NIO);
[a2,b2,c2,d2] = ncdtf2ss(NUM2,DEN2,NIO);
if nargout == 4
    [ A,B,C,D ] = mvpar(a1,b1,c1,d1,a2,b2,c2,d2);
else
    [ a,b,c,d ] = mvpar(a1,b1,c1,d1,a2,b2,c2,d2);
    [A,B] = mvss2tf(a,b,c,d);
end % if nargout == 4
end % ncdpar
function [A,B,C,D] = ncdser(NUM1,DEN1,NI01,NUM2,DEN2,NI02)

% [A,B,C,D] = ncdser(NUM1,DEN1,NUM2,DEN2,NI01,NI02) connects
% two N.C.D.systems in series. If either of the two systems is
% a pure gain block it is reduced. The two systems to be
% cascaded are given in a 'block diagram' order, so for
% example if { NUM1, DEN1, NI01 } describe a pre-compensator
% K, and { NUM2, DEN2, NI02 } describe the plant T.F.M. G, the
% state-space model {A,B,C,D} gives the forward path T.F.M.
% Q = G.K. If the function is invoked as :
% [MVNUM,Comden] = ncdser(NUM1,DEN1,NUM2,DEN2,NI01,NI02) the
% resulted system is given by an MFD format transfer function
% matrix (see the MFD Toolbox manual).
% NUM1, NUM2 : Are the vectors that stores the numerators of
% the two cascaded N.C.D. (non-common
% denominator) transfer function matrices.
% DEN1, DEN2 : Are the vectors that stores the denominators of
% the cascaded N.C.D. Transfer function matrices.
% NI01, NI02 : Are the two element row vectors giving the
% number of inputs NI0(1) and the number of
% outputs NI0(2) of systems 1 and 2 respectively.

[a1,b1,c1,d1] = ncdtf2ss(NUM1,DEN1,NI01);
a2,b2,c2,d2] = ncdtf2ss(NUM2,DEN2,NI02);
if nargout == 4
    [A,B,C,D] = mvser(a1,b1,c1,d1,a2,b2,c2,d2);
else
    [a,b,c,d] = mvser(a1,b1,c1,d1,a2,b2,c2,d2);
    [A,B] = mvss2tf[a,b,c,d];
end % if nargout == 4
end % ncdser

A2.9
function \([\text{NUM}, \text{DEN}, \text{NIO}] = \text{ncdser2}(\text{NUM1}, \text{DEN1}, \text{NIO1}, \text{NUM2}, \text{DEN2}, \text{NIO2})\)

\([\text{NUM}, \text{DEN}, \text{NIO}] = \text{ncdser2}(\text{NUM1}, \text{DEN1}, \text{NIO1}, \text{NUM2}, \text{DEN2}, \text{NIO2})\)

% connects two N.C.D systems in series. The system returned by
% ncdser2 is in N.C.D. transfer function format. If either of
% the two systems is a pure gain block it is reduced. The two
% systems to be cascaded are given in a 'block diagram' order.
% so for example if \(\{ \text{NUM1}, \text{DEN1}, \text{NIO1} \}\) describe a pre-
% compensator \(K\), and \(\{ \text{NUM2}, \text{DEN2}, \text{NIO2} \}\) describe the plant
% T.F.M. \(G\), the forward path T.F.M. \(O = G.K\), is given by
% \(\{ \text{NUM}, \text{DEN}, \text{NIO} \}\).

% NUM1, NUM2 : Are the vectors that stores the numerators of
% the two cascaded N.C.D. (non-common
% denominator) transfer function matrices.
% DEN1, DEN2 : Are the vectors that stores the denominators of
% the cascaded N.C.D. Transfer function matrices.
% NIO1, NIO2 : Are the two element row vectors giving the
% number of inputs NIO(1), and the number of
% outputs NIO(2) of systems 1 and 2 respectively.

l = NIO2(2);
k = NIO2(1);
m = NIO1(1);
for i = 1:l
    for j = 1:m
        nsum = [];
dsum = [];
        for p = 1:k
            nsum = polysum(nsum, conv(NUM2(k*(i-1)+p,:), NUM1(m*(p-1)+j,:)));  
dsum = polysum(dsum, conv(DEN2(k*(i-1)+p,:), DEN1(m*(p-1)+j,:)));  
        end
        for p = 1:k
            NUM(m*(i-1)+j,:) = nsum;
            DEN(m*(i-1)+j,:) = dsum;
        end
    end
end
if nargout == 3
    NIO(1) = m;
    NIO(2) = 1;
end  % if nargout == 3
end  % nodser2
function getrgna(w,G,x)

% getrgna(w,G,x)
% function to plot the Direct Nyquist Array of the MVFR G,
% with row Gershgorin bands superimposed.
% G : The frequency response in MFD format
% w : The frequency vector
% x : When x is supplied, specifies the axis setting.
% getrgna works with 2x2 systems.

if nargin == 3, axis(x), end
subplot(221)
plotnyq(frgersh(w,G,1,2)), hold on
plotnyq(fget(w,G,[1,1])), title('Element [1,1]'), hold off
subplot(222)
if nargin == 3, axis(x), end
plotnyq(fget(w,G,[1,2])), title('Element [1,2]')
subplot(223)
if nargin == 3, axis(x), end
plotnyq(fget(w,G,[2,1])), title('Element [2,1]')
subplot(224)
if nargin == 3, axis(x), end
plotnyq(frgersh(w,G,2,2)), hold on
plotnyq(fget(w,G,[2,2])), title('Element [2,2]')
subplot(111)
end
function getogna(w,G,x)

% getogna(w,G,x)
% function to plot the Direct Nyquist Array of the MVFR G,
% with column Gershgorin bands superimposed.
% G : The frequency response in MFD format
% w : The frequency vector
% x : When supplied, x specifies the axis setting.
% getogna works with 2x2 systems

subplot(221)
if nargin == 3, axis(x), end
plotnyq(fogersh(w,G,1,2)), hold on
plotnyq(fget(w,G,[1,1])), title('Element [1,1]'), hold off
subplot(222)
if nargin == 3, axis(x), end
plotnyq(fget(w,G,[1,2])), title('Element [1,2]')
subplot(223)
if nargin == 3, axis(x), end
plotnyq(fget(w,G,[2,1])), title('Element [2,1]')
subplot(224)
if nargin == 3, axis(x), end
plotnyq(fogersh(w,G,2,2)), hold on
plotnyq(fget(w,G,[2,2])), title('Element [2,2]')
subplot(111)
end
function getrona(w,G,fs,x)

% getrona(w,G,fs,x)
% function to plot the Direct Nyquist Array of the MVFR G,
% with row Ostrowski bands superimposed.
%  G  : The frequency response in MFD format
%  w  : The frequency vector
%  fs  : is the vector that contains the feedback gains.
%  x  : When x is supplied, specifies the axis setting.
% getrona works with 2x2 systems.

if nargin == 4, axis(x), end
subplot(221)
plotnyq(frost(w,G,1,fs,2)), hold on
plotnyq(fget(w,G,[1,1])), title('Element [1,1]'), hold off
subplot(222)
if nargin == 4, axis(x), end
plotnyq(fget(w,G,[1,2])),title('Element [1,2]')
subplot(223)
if nargin == 4, axis(x), end
plotnyq(fget(w,G,[2,1])), title('Element [2,1]')
subplot(224)
if nargin == 4, axis(x), end
plotnyq(frost(w,G,2,fs,2)), hold on
plotnyq(fget(w,G,[2,2])), title('Element [2,2]'), hold off
subplot(111)
end
function getoona(w,G,fs,x)

% getoona(w,G,fs,x)
% function to plot the Direct Nyquist Array of the MVFR G,
% with column Ostrowski bands superimposed.
% G : The frequency response in MFD format
% w : The frequency vector
% fs : is the vector that contains the feedback gains.
% x : When x is supplied, specifies the axis setting.
% getoona works with 2x2 systems.

if nargin == 4, axis(x), end
subplot(221)
plotnyq(fcost(w,G,1,fs,2)), hold on
plotnyq(fget(w,G,[1,1])), title('Element [1,1]'), hold off
subplot(222)
if nargin == 4, axis(x), end
plotnyq(fget(w,G,[1,2])), title('Element [1,2]')
subplot(223)
if nargin == 4, axis(x), end
plotnyq(fget(w,G,[2,1])), title('Element [2,1]')
subplot(224)
if nargin == 4, axis(x), end
plotnyq(fcost(w,G,2,fs,2)), hold on
plotnyq(fget(w,G,[2,2])), title('Element [2,2]'), hold off
subplot(111)
end
function getrgbod(w,G,el)

% getrgbod(w,G,el)
% function to plot the Bode Array of the MVFR G,
% with row Gershgorin bands superimposed.
% G : The frequency response in MFD format
% w : The frequency vector
% el : When el is supplied, specifies which element of G is
% to be plotted.
% getrgbod works with 2x2 systems, if the input el is not
% given. Note also that the gain axis is not logarithmic. If
% the gain must be plotted in db use the function getrgbdb
if (nargin == 3)
    g' = abs(fget(w,G,[el,el]));
    d = frsod(w,G); d = d(:,el);
    semilogx(w,[g+d,g-d],'-'), hold on, semilogx(w,g), grid, hold .
    off
    sel = [' [',num2str(el),',',num2str(el),']'];
    title(['Element ', sel, ' of TFM with row Gershgorin band'])
    xlabel('Frequency'), ylabel('gain V/V')
else
    subplot(211)
    g = abs(fget(w,G,[1,1]));
    d = frsod(w,G); d1 = d(:,1);
    semilogx(w,[g+d1,g-d1],'-'), hold on, semilogx(w,g)
    grid, hold off
    title('Bode array with row Gershgorin band')
    xlabel('Frequency'), ylabel('gain V/V')
    subplot(212)
    g = abs(fget(w,G,[2,2]));
    d = d(:,2);
    semilogx(w,[g+d,g-d],'-'), hold on, semilogx(w,g)
    grid, hold off
    xlabel('Frequency'), ylabel('gain V/V')
    subplot(111)
end
function getogbod(w,G,el)
% getogbod(w,G,el)
% function to plot the Bode Array of the MVFR G,
% with column Gershgorin bands superimposed.
% G : The frequency response in MFD format
% w : The frequency vector
% el : When el is supplied, specifies which element of G is
% to be plotted.
% getogbod works with 2x2 systems, if the input el is not
% given. Note also that the gain axis is not logarithmic. To
% have the gain plotted in db use the function getcgdb
% if (nargin == 3)
g = abs(fget(w,G,[el,el]));
d = fcsod(w,G); d = d(:,el);
semilogx(w, [g+d,g-d], '--'),hold on,semilogx(w,g),grid,hold off
sel = ['[',num2str(el),',',num2str(el),']'];
title(['Element ',sel,' of TFM with column Gershgorin band'])
xlabel('Frequency'),ylabel('gain V/V')
else
subplot(211)
g = abs(fget(w,G,[1,1]));
d = fcsod(w,G); d1 = d(:,1);
semilogx(w, [g+d1,g-d1], '--'),hold on,semilogx(w,g),grid
hold off
xlabel('Bode array with column Gershgorin band')
xlabel('Frequency'),ylabel('gain V/V')
subplot(212)
g = abs(fget(w,G,[2,2]));
d = d(:,2);
semilogx(w, [g+d,g-d], '--'),hold on,semilogx(w,g),grid
hold off
xlabel('Frequency'),ylabel('gain V/V')
end
function getrobod(w,G,F,el)

% getrobod(w,G,el)
% function to plot the Bode Array of the MVFR G,
% with row Ostrowski bands superimposed.
% G : The frequency response in MFD format
% w : The frequency vector
% F : The vector of feedback gains.
% el : When el is supplied, specifies which element of G is
% to be plotted.
% getrobod works with 2x2 systems, if the input el is not
% given. Note also that the gain axis is not logarithmic. If
% the gain must be plotted in db use the function getrodb
%

cent=fdiag(w,G);
rad=frsod(w,G);

[rows,cols] = size(cen);

for r = 1:rows
    for ci = 1:cols
        max = 0;
        for cj = 1:cols
            if cj ~= ci
                val = rad(r,cj)/abs(F(cj)+cent(r,cj));
                if val > max
                    max = val;
                end
            end
        end
        rad(r,ci) = rad(r,ci)*max;
    end
    end
if (nargin == 4)
    g = abs(fget(w,G,[el,el]));
end
d = rad(:, el);
semilogx(w, [g+d, g-d], '--').hold on, semilogx(w, g)
grid, hold off
sel = ['[', num2str(el), ',', num2str(el), ']'];
title(['Element ', sel, ' of TFM with row Ostrowski band']);
xlabel('Frequency'), ylabel('gain V/V')
else
subplot(211)
g = abs(fget(w, G, [1, 1]));
d1 = rad(:, 1);
semilogx(w, [g+d1, g-d1], '--').hold on, semilogx(w, g)
grid, hold off
title('Bode array with row Ostrowski band')
xlabel('Frequency'), ylabel('gain V/V')
subplot(212)
g = abs(fget(w, G, [2, 2]));
d = rad(:, 2);
semilogx(w, [g+d, g-d], '--').hold on, semilogx(w, g)
grid, hold off
xlabel('Frequency'), ylabel('gain V/V')
end
hold off

sel = ['[' ,num2str(el), ', ',num2str(el), ' ]'];
title(['Element ',sel,' of TFM with column Ostrowski band'])
xlabel('Frequency'), ylabel('gain V/V')
else

subplot(211)
  g = abs(fget(w,G,[1,1]));
  d1 = rad(:,1);
  semilogx(w,[g+d1,g-d1], '--'), hold on, semilogx(w,g), grid
hold off
  title('Bode array with column Ostrowski band')
  xlabel('Frequency'), ylabel('gain V/V')

subplot(212)
  g = abs(fget(w,G,[2,2]));
  d = rad(:,2);
  semilogx(w,[g+d,g-d], '--'), hold on, semilogx(w,g), grid
hold off
  xlabel('Frequency'), ylabel('gain V/V')

subplot(111)
end
function getcobod(w,G,F,el)

% getcobod(w,G,el)
% function to plot the Bode Array of the MVFR G,
% with column Ostrowski bands superimposed.
% G : The frequency response in MFD format
% w : The frequency vector
% F : The vector of feedback gains.
% el : When el is supplied, specifies which element of G is
% to be plotted.
% getcobod works with 2x2 systems, if the input el is not
% given. Not also that the gain axis is not logarithmic. If
% the gain must be plotted in db use the function getcodb

cent=fdiag(w,G);
rad=fosod(w,G);
[rows,cols] = size(cent);
for r = 1:rows
    for ci = 1:cols
        max = 0;
        for cj = 1:cols
            if cj ~= ci
                val = rad(r,cj)/abs(F(cj)+cent(r,cj));
                if val > max
                    max = val;
                end
            end
        end
        rad(r,ci) = rad(r,ci)*max;
    end
end
if (nargin == 4)
    g = abs(fget(w,G,[el,el]));
    d = rad(:,el);
    semilogx(w,[g+d,g-d],'--',hold on,semilogx(w,g),grid
function f=getmvshp(X,m,n)

% getmvshp reshapes matrix X into MVFR matrix. Modified SHPF, 
% getmvshp(X,m), reshapes the columns of X into an MVFR 
% matrix with square component matrices m-by-m. 
% X is a matrix with one column per element of the MVFR matrix 
% and one row per frequency. Each row is reshaped into a 
% component matrix using A(:) where A=eye(n). The 2nd column 
% of X contains the (1,2) elements of the MVFR, and the 3rd 
% column of X contains the (2,1) elements of the MVFR. The 
% reverse is true for the function SHPF, that this function 
% modifies. 
% 
% getmvshp(X,m,n) reshapes the rows of X into component 
% matrices with m rows and n columns. 
% 
% getmvshp (X,A) reshape the row of X into component matrices 
% the same size as A.

nargs=nargin;
error(nargchk(2,3,nargs));
[mx,nx]=size(x);
if nargs==2
    a=eye(m);
else
    a=eye(m,n);
end
[m,n]=size(a);
if m*n~=nx
    error('Size not consistent with number of columns of X')
end
k=1:m;
for i=0:mx-1
    a(:)=x(i+1,:);
    f(k+i*m,:)=a';
end
function [Rw,VL,UR] = getpfeig(w,G)

% [Rw,VL,UR] = getpfeig(w,G)
% % Function to calculate the Perron - Frobenius eigenvalue and left and right eigenvectors of the frequency response matrix G
% % w : The frequency vector
% G : The system frequency response in MFD format
% Rw : The P-F eigenvalues. Rw(i) corresponds to the frequency w(i)
% VL : The P-F left eigenvectors. The ith row contains the left P-F eigenvector corresponding to Rw(i).
% UR : The P-F right eigenvectors. The ith row contains the right P-F eigenvector corresponding to Rw(i).
% % The P-F dominance graph is plotted.
% % The eigenvetors are normalized so that V(i,1) = U(i,1) = 1.

M = abs(G);
Omega = fdiag(w,fdiag(w,M));
Gamma = fmulf(w,finv(w,Omega),M);
[Rw,VL,UR] = fperron(w,Gamma);
semilogx(w,Rw),grid
title('P-F dominance graph'),xlabel('Log(w)')
ylabel('P / F r')
for i = 1:length(w)
    VL(i,:) = VL(i,:) / VL(i,1);
    UR(i,:) = UR(i,:) / UR(i,1);
end
function [Rw,Vl,Ur] = getpfeig(w,G)

%M = abs(G);
Omega = fdiag(w,fdiag(w,M));
Gamma = fmulf(w,finv(w,Omega),M);
[Rw,Vl,Ur] = fperron(w,Gamma);
semilogx(w,Rw),grid
title('P-F dominance graph '),xlabel('Log(w)')
ylabel('P / F r')
for i = 1:length(w)
    Vl(i,:) = Vl(i,:) / Vl(i,1);
    Ur(i,:) = Ur(i,:) / Ur(i,1);
end
function getdbarr(w,G)

% getdbarr(w,G)
%
% Procedure to plot the array of the magnitudes of the MVFR G.
% w is the frequency vector. The array consists of plots of db
% against frequency in logarithmic scale.
%
subplot(221)
mvdb(w,G,[1,1]), title('Element [1,1]')
mvdb(w,G,[1,2]), title('Element [1,2]')
mvdb(w,G,[2,1]), title('Element [2,1]')
mvdb(w,G,[2,2]), title('Element [2,2]')
subplot(111)
function blokprog(w,so,ts,tinc)
    % blokprog(w,so,ts,tinc)
    %
    % procedure to plot a fatigue test 'block program' waveform.
    % w : is the frequency of the sine wave.
    % so : the amplitude of the sine wave in each block.
    % ts : a vector containing the time durations of each
    % block in the program.
    % tinc: the time increment, for the plot
    %
    t0 = 0; t = []; s = []; x = [];
    for i = 1:length(ts)
        tl = t0+tinc:ts(i);
        t = [t,tl];
        s = [s,so(i)*sin(w*tl)];
        x = [x,so(i)*ones(1,length(tl))];
        t0 = ts(i)+tinc;
    end
    plot(t,s), hold on, plot(t,x), plot(t,-x), hold off
function [num, den] = anbutter(n, wc, ftype)

% [num, den] = anbutter(n, wc, ftype)
% function to return the transfer function of an analogue
% Butterworth filter of a given order and cut-off frequency.
% wc : The filter's cut-off frequency (-3db level)
% n : The order of the filter.
% ftype : If ftype is omitted a low pass filter is designed
%         when wc is a scalar and a bandpass filter is
%         designed when wc is a two element vector.
%         if ftype = 'high' a high pass filter is designed
%         if ftype = 'stop' and wc = [wl, w2] then a band
%         stop filter is designed of order 2n
% num : The numerator of the filter's transfer function.
% den : The denominator of the filter's transfer function.

btype = 1;
if nargin == 3
    btype = 3;
end
if max(size(wc)) == 2
    btype = btype + 1;
end

% step 1: get analog, pre-warped frequencies
u = wc;

% step 2: convert to low-pass prototype estimate
if btype == 1
    wo = u;
elseif btype == 2
    Bw = u(2) - u(1);
    wo = sqrt(u(1)'*u(2));
elseif btype == 3
    wo = u;
elseif btype == 4
    Bw = u(2) - u(1);
\[ \text{wc} = \sqrt{u(1)u(2)}; \]

% center frequency

end

% step 3: Get n-th order Butterworth analog lowpass prototype
\[ [z,p,k] = \text{buttap}(n); \]

% Transform to state-space
\[ [a,b,c,d] = \text{zp2ss}(z,p,k); \]

% step 4: Transform to lowpass, bandpass, highpass,
% or bandstop of desired wc
if btype == 1 % Lowpass
\[ [a,b,c,d] = \text{lp2lp}(a,b,c,d,\text{wc}); \]
elseif btype == 2 % Bandpass
\[ [a,b,c,d] = \text{lp2bp}(a,b,c,d,\text{wc},Bw); \]
elseif btype == 3 % Highpass
\[ [a,b,c,d] = \text{lp2hp}(a,b,c,d,\text{wc}); \]
elseif btype == 4 % Bandstop
\[ [a,b,c,d] = \text{lp2bs}(a,b,c,d,\text{wc},Bw); \]
end

if nargout == 4
num = a;
den = b;
z = c;
p = d;
else % nargout <= 3
% Transform to zero-pole-gain and polynomial forms:
if nargout == 3
\[ [z,p,k] = \text{ss2zp}(a,b,c,d,1); \]
num = z;
den = p;
z = k;
else % nargout <= 2
\[ \text{den} = \text{poly}(a); \]
\[ \text{num} = \text{poly}(a-b*c)+(d-1)*\text{den}; \]
end
end