FREQUENCY DOMAIN APPROACHES TO FAULT DETECTION IN CLOSED-LOOP SYSTEMS

by

JOSEPH HURK-NING CHENG

A Thesis presented for the Degree of
Master of Science in Engineering

Department of Electronics and
Electrical Engineering,
(Control Systems Engineering),
University of Glasgow.

January, 1986.

ACKNOWLEDGEMENTS

The author wishes to thank Professor J. Lamb, head of the department of Electronics and Electrical Engineering at Glasgow University, and Professor D. J. Murray-Smith for their continued interest and encouragement during the duration of the research and in particular Professor D. J. Murray-Smith for his supervision of the project and the valuable suggestions which he offered.

Most of the work described in this thesis was carried out at the Computer facility of the Faculty of Engineering at Glasgow University. The author would like to thank the computing staff for their help and in particular to Miss A. I. MacKinnon for making the facilities available.

Thanks is also extended to the BRITOIL company for the financial support of this project made available through a research studentship and for the guidelines which Mr. H. M. Khatib, engineering director of the company, laid out as a skeleton on which to base this project.

SUMMARY

This thesis describes fault detection techniques which can be applied to closed-loop automatic control systems. Particular emphasis is placed upon frequency domain methods.

Digital simulation is used for the evaluation of on-line techniques for fault detection using pseudo-random-binary test signals. This simulation work involves a simple process model based on a two tank liquid flow control system.

Frequency-domain identification using pseudo-random-binary test signals is performed by means of a mixed radix-2 fast Fourier transformation technique. This technique avoids the synchronisation problems which arise when the more conventional radix-2 transformation is used with pseudo-random binary test signals.

A novel method of parameter sensitivity analysis is investigated both in terms of the identification of faults and for system optimisation following fault occurrence.

CONTENTS

LIST OF FIGURES

CHAPTER ONE

1. INTRODUCTION TO FAULT DETECTION TECHNIQUES

1.1 General

Safety and reliability are two of the prime requirements in the design of control systems. They are particularly important when designing full-time flight control systems, or systems for large sophisticated process plants or nuclear power stations[1] etc. where a system failure may lead to some fatal consequences or loss of profit. To secure the reliability of a system, it is necessary to employ an efficient fault detection technique to safeguard against any system fault which may occur. The fault detection method should be able to discover and locate the system fault at an early stage thus allowing appropriate action to be taken before any hazardous situation can occur.

A system fault can be defined as an unpermittable deviation of the normal system characteristic that will result in an undesired system response which is incapable of meeting the original requirements or specfications. A system fault may be caused by a change of a system parameter or an error in a sensor etc. It should be noted that system failure can sometimes be brought about by a system malfunction which may be referred to as a deterioration of the state of a system. System malfunction can cause some incipient instrument failures which may not have any apparent effect on the system performance. However they are important because they might lead to some serious consequences when a more complete system failure occurs. Lee[2] has given a detailed account of various instrument malfunctions and their classification. In the following, a brief discussion of fault detection methods and redundancy schemes will be made.

Various fault detection techniques[3] have been developed in the past. Their usual objective is to discover the faults in the system earlier and locate them more accurately, and if possible identify the nature of the fault. The conventional out-of-range method provides a useful test to check if the value of a state variable is outside a prespecified range within which the values are considered as normal or acceptable. This method is restricted to check only the measurable variables. Later development of process computers and microcomputers allows the construction of the mathematical process models and signal models. These models can be used to predict signals or to

estimate non-measurable process state variables and other quantities characteristic of the process.

When a system fault has been identified, it is necessary to estimate the effect of the fault on the whole system. Once the effect of the fault is known, the operator of the control system itself should take appropriate action to counteract the fault. If the fault is found to be tolerable then the system may be allowed to continue its process. When the fault is found to be conditionally tolerable, some adjustment must be made in order to obtain a desirable system behaviour. In the worst case, when the fault is intolerable, the operation may have to be stopped completely. The fault in the system must be recovered and eliminated before the process can be restarted.

In some circumstances where dangerous reactions or toxic material are present inside a control system, it will be impractical to shut down the system because there could be a risk of explosion or a release of toxic material. Sometimes, a system shut down could cause a great loss in profit with possible loss of customers. Installation of redundancy equipment[4] could be a solution to these problems. When a system fault occurs, the faulty component can be taken over by redundant equipment thus allowing the system process to continue normally. The use of redundancy scheme can protect the system against system faults and breakdowns. The reliability of the system is then improved.

In a redundancy scheme, the system is duplicated into a number of identical components. The signal from each component is compared and checked with one another. The result of the checking can indicate if any of the components is faulty. The most common redundancy schemes are duplex, triplex and quadruplex systems. The choice of using a particular kind of redundancy scheme depends upon the specification of the system in question.

The design of a redundant system is not a straight forward task because not all components need the same degree or the same kind of redundancy. One needs to consider a number of factors such as: (1)the probabilities of failure in the circuit, (2)how often the component is to be used, (3)the location of the circuit with regard to the propagation of the fault from it etc. A choice must be made of the components for which redundancy could be most rewarding and best applied. The application of redundancy for each component can be based upon the extent to which the component is critical to the overall system function.

Redundancy will be introduced to those critical parts first. If the weight, volume and budget have not been exceeded, then redundancy could be applied to the less critical parts. In some circumstances, the level of redundancy of the critical parts would be increased further, e.g. the introduction of an additional set of redundant components.

One important issue to be considered when establishing a redundant system is the common-mode-fault[5]. It can be considered as a system defect which has unwanted simultaneous effects on all redundant components within the system. One kind of common-mode fault is associated with the power supply of the system. Therefore some measures to counteract any power failure must be introduced. Duplicated power pack , as suggested in Johnson's paper[5] is one good way to counteract common-mode faults in the power supply. The duplicated power pack consists of two power supplies with an automatic changover facility. Another alternative is the use of battery as a back-up power supply.

A redundant system which consists of a computerised controller may have common-mode faults due to program errors implanted within the system software. In fact, it is agreed that no software can be completely debugged and fully verified for all possible conditions of operation. For this reason, there are some bugs present all the time. This sort of common-mode fault is of particular concern due to the computerisation of most modern control systems. This means that digital control systems always carry some software common-mode faults and these faults could affect all the redundant components simultaneously. To overcome this problem, one can make use of software packages written by different teams, or use processor types made by different manufacturer. This is to minimise the chance of having the same software error in dissimilar programs or processors.

Although the original purpose of using a redundancy system is to improve the reliability of the overall system, it must be remembered that the additional redundant equipment is subject to faults itself, thus affecting the reliability of the system. Various internal checking techniques can be carried out to improve the reliability of the overall system : the use of Watch-dog timer for checking the execution time, the use of Cyclic Redundancy Code for checking the correct data transmission etc.

The use of redundancy schemes is an important approach to secure the system process against break-down or malfunction of any kind. However, the idea of replicating system components in a redundant system is usually costly and will make a simple system become very complicated. In addition, the problem of common-mode-fault always exists in a redundancy scheme. To solve some of these problems, one may use an analytical redundancy scheme in which the system faults are monitored without duplicating hardware channels,i.e. using only one set of hardware components. This scheme analyses the mutual relationship of two dissimilar signals. One of the signals can be obtained from the sensor output or the component output. Another signal can be the output of a software redundancy program which represents the theoretical model of the estimated system behaviour. When a fault occurs in the system, the relationship between the actual system output and the estimated signal will differ in a detectable way. Each redundancy system must be accompanied by an efficient fault detection method which can discover a fault promptly at its early stage.

Several fault detection techniques will be reviewed in the next few sections. The methods can be classified into two categories as follows :

1) Techniques which require exact knowledge of the system's structure and parameters, and also the input signal. Examples of this kind are the observer-based technique[6] and the Kalman filter technique[7]. Their approach is to reconstruct the state variables from the input and output signals using a known process model. The establishment of this process model requires an exact knowledge of the process parameters.

2) Fault detection techniques which are based only on some measurable signal from the system, such as input and output variables U(t) and Y(t). Methods like the out-of-range check[3], the signal processing technique (which uses a band-limited filter[8]) and the system identification technique (using the system response model[9]) can all be used to monitor the system performance without requiring any knowledge of the system structure.

1.2 Model-Based Methods of Fault Detection

1.2.1 Observer-Based Technique

An observer system[6] is a device which can produce an estimate of the state vector of a system based upon its available input and output signals. The observer should have the same state equation as that of the original system being monitored. A simple example is illustrated below:

Consider a single-input-single-output with state equation:

$$\dot{\underline{X}} = \underline{A}\,\underline{X} + \underline{B}\,U \qquad (1.1)$$

$$Y = \underline{C}\,\underline{X} \qquad (1.2)$$

where $\underline{X}$ is the state vector, Y is the output vector,
U is the input vector, $\underline{A}$ is the system matrix,
$\underline{B}$ is the input matrix, $\underline{C}$ is the output matrix.

An simplified block diagram of a linear system with state observer is shown in the diagram below.

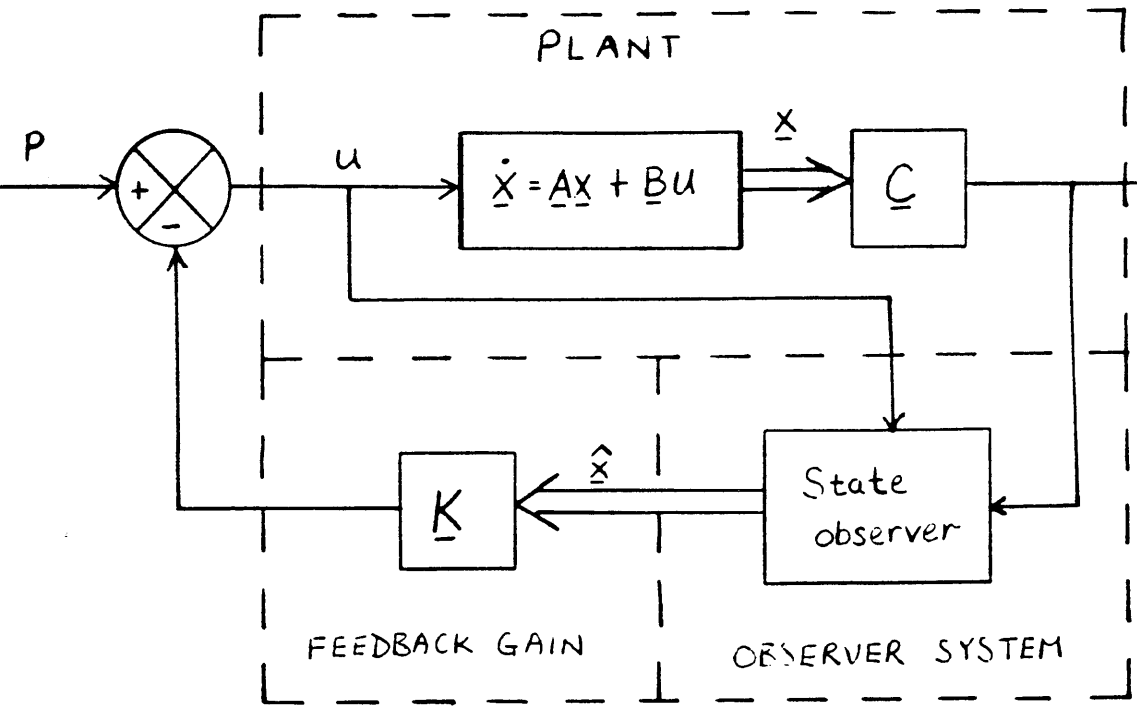


Figure 1 Block Diagram of a Linear System with State Observer

where P is the operator input, $\underline{K}$ is the feedback matrix.

From the observer's state vector $\hat{\underline{X}}$, the estimated output vector $\hat{Y}$ can be obtained by the equation:

$$\hat{Y} = \underline{C}\,\underline{X} \qquad (1.3)$$

The idea is to minimise the difference between $\hat{Y}$ and Y. This is equivalent to minimising $\hat{\underline{X}}$ and $\underline{X}$. However, in practice, $\underline{X}$ is not always accessable. Therefore the minimisation can only be made between $\hat{Y}$ and Y. The difference $(\hat{Y} - Y)$ is multiplied by a factor $\underline{G}$. The signal is then fed back into the input of the integrators of the observer. The block diagram of the observer system is shown below.

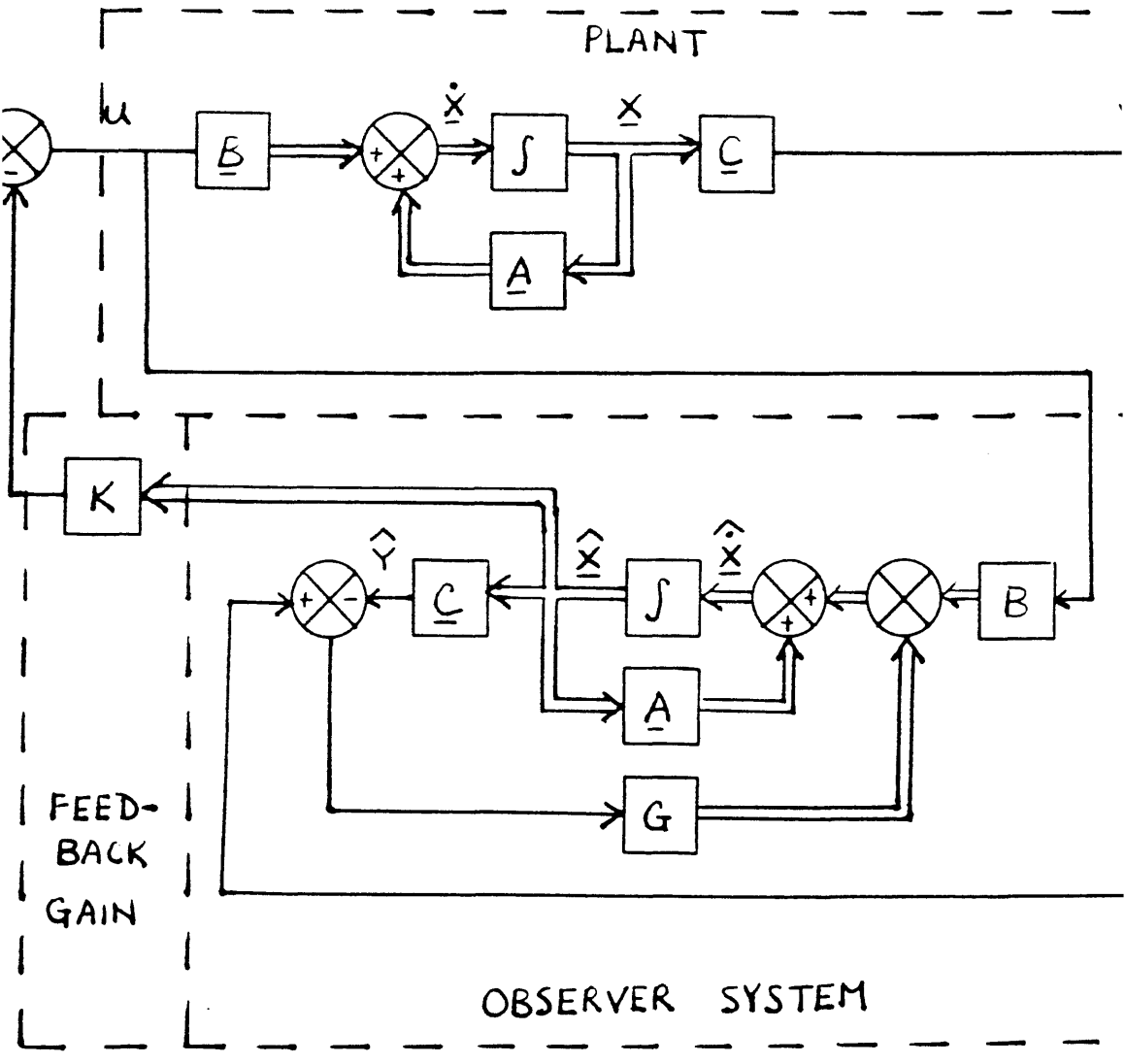


Figure 2 Block Diagram of the Observer System

It is required to find a suitable value of G such that ($\hat{Y}$ - Y) is a minimum. From the above diagram, the following equations can be derived as:

$$\dot{\hat{\underline{X}}} = \underline{A}\,\hat{\underline{X}} + \underline{G}\,(\underline{C}\,\underline{X} + \underline{C}\,\hat{\underline{X}}) + \underline{B}\,U \qquad (1.4)$$

$$= (\underline{A} - \underline{G}\,\underline{C})\,\hat{\underline{X}} + \underline{G}\,\underline{C}\,\underline{X} + \underline{B}\,U \qquad (1.5)$$

$$\dot{\underline{X}} - \dot{\hat{\underline{X}}} = (\underline{A} - \underline{G}\,\underline{C})\,(\underline{X} - \hat{\underline{X}}) \qquad (1.6)$$

The error vector $\underline{e}$ is defined as :

$$\underline{e} \backsimeq \underline{X} - \hat{\underline{X}} \qquad (1.7)$$

This gives : $$\dot{\underline{e}} \backsimeq (\underline{A} - \underline{G}\,\underline{C})\,\hat{\underline{X}} \qquad (1.8)$$

The matrix $\underline{G}$ must be chosen so that $\underline{e}$ will decay to zero. The rate of decay of $\underline{e}$ can be increased by choosing the suitable eigenvalues of ($\underline{A}$ - $\underline{G}\,\underline{C}$). A fault detection scheme can be established by comparing the measurement vector $\underline{X}$ with the estimated state vector $\hat{\underline{X}}$. The difference between the two vectors can be checked against a threshold level. When the threshold level is exceeded, a fault is said to have been detected and appropriate action can be taken.

Mclean's paper[10] describes the use of a observer system accompanied by a predictive technique. This technique could generate a predicted state vector $\underline{X}_p$. Under the circumstance when three state vectors : measured vector $\underline{X}$, estimated vector $\hat{\underline{X}}$ and predicted vector $\underline{X}_p$, are involved in the fault detection scheme, the majority voting scheme is used for comparing the state vectors to check if the original system is faulty.

The use of an observer requires that the measurement of the state variables be noise free. In the other words the system must be deterministic. If any system noise is present, it must have negligible impact on the system. However, in the situation when considerable amount of system noise is present, a state variable filter (e.g. Kalman filter) would be more appropriate than an observer system. A description of the Kalman filter will be given in the next section. One last point to note about the observer system is that the system parameters $\underline{A}$, $\underline{B}$ and $\underline{C}$ must be precisely given in order to construct an efficient observer system.

1.2.2 Kalman Filter Technique

As mentioned in the previous section, an observer system can be applied efficiently only on a deterministic system. Under noisy circumstances, the use of Kalman filter[7] would be more appropriate. It estimates the system states using the previous system estimates and system external measurements. The former are likely to contain error caused by sensor errors while the latter may contain random measurment noise. The optimal use of the system estimates and external measurements together can yield a better and more accurate system estimate than that obtained from either the system estimate or the extermal measurement alone. A properly designed Kalman filter should be able to produce an unbaised estimate of the original system based upon system previous estimates and system external measurements.

The state equations of a stochastic system whose input and output are contaminated with noise can be written as :

$$\dot{\underline{X}} = \underline{A}\,\underline{X} + \underline{B}\,(\underline{U} + \underline{V}) \qquad (1.9)$$

$$\hat{\underline{Y}} = \underline{C}\,\underline{X} + \underline{W} \qquad (1.10)$$

where $\underline{X}$ is the state vector
$\hat{\underline{Y}}$ is the output measurement vector
$\underline{U}$ is the input vector
$\underline{V}$ is the input noise vector
$\underline{W}$ is the output measurement noise vector

$\underline{V}(t)$ and $\underline{W}(t)$ both are white Gaussian stationary processes with zero mean and they are independent of each other. Their autocorrelation can be written as :

$$\phi_{VV}(\tau) = E\,[\,\underline{V}\,(t)\,\underline{V}^T\,(t+\tau)\,] \qquad (1.11)$$

$$= \underline{Q}\;\delta(\tau) \qquad (1.12)$$

and

$$\phi_{WW}(\tau) = E\,[\,\underline{W}\,(t)\,\underline{W}^T\,(t+\tau) \qquad (1.13)$$

$$= \underline{R}\;\delta(\tau) \qquad (1.14)$$

where $\underline{Q}$ and $\underline{R}$ are the covariances of $\underline{V}$ and $\underline{W}$ respectively

$\underset{\sim}{\underline{Y}}$ refers to the measurement output which is not necessarily the same as the actual control output. The actual controlled outputs are retained by the original notation :

$$\underline{Y} = \underline{C}\,\underline{X} \tag{1.15}$$

The block diagram of the stochastic system is shown below.

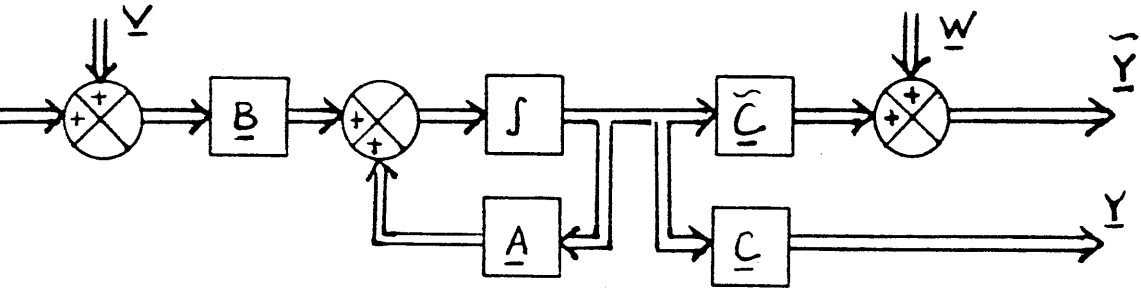


Figure 3 Block Diagram of a Stochastic System

A Kalman filter is used to produce an estimated state vector $\hat{\underline{X}}(t)$ from the noisy measurement $\underset{\sim}{\underline{Y}}(t)$. An optimal input vector is then obtained from the estimated state vector $\hat{\underline{X}}(t)$ by the optimal control law :

$$\underline{U}_o = -\underline{K}\,\hat{\underline{X}} \tag{1.16}$$

where $\underline{U}_o$ is the optimum of $\underline{U}$
and $\underline{K}$ is the feedback matrix

The dynamic equation of the Kalman filter can be expressed as :

$$\dot{\hat{\underline{X}}} = \underline{A}\,\hat{\underline{X}} + \bar{\underline{K}}\,(\underset{\sim}{\underline{Y}} - \underset{\sim}{\underline{C}}\,\hat{\underline{X}}) + \underline{B}\,\underline{U} \tag{1.17}$$

where $\bar{\underline{K}} = \underline{R}^{-1}\,\underset{\sim}{\underline{C}}\,\underline{M}$ (1.18)

and $\bar{M}$ (t) is the steady state solution of a matrix Riccati equation :

$$-\dot{\bar{M}} + \bar{M}\,A^T - \bar{M}\,\hat{C}^T\,R^{-1}\,\hat{C}\,\bar{M} + B\,\bar{Q}\,B^T + A\,M = 0 \qquad (1.19)$$

The block diagram of the plant-estimation-control system is shown in the diagram below :

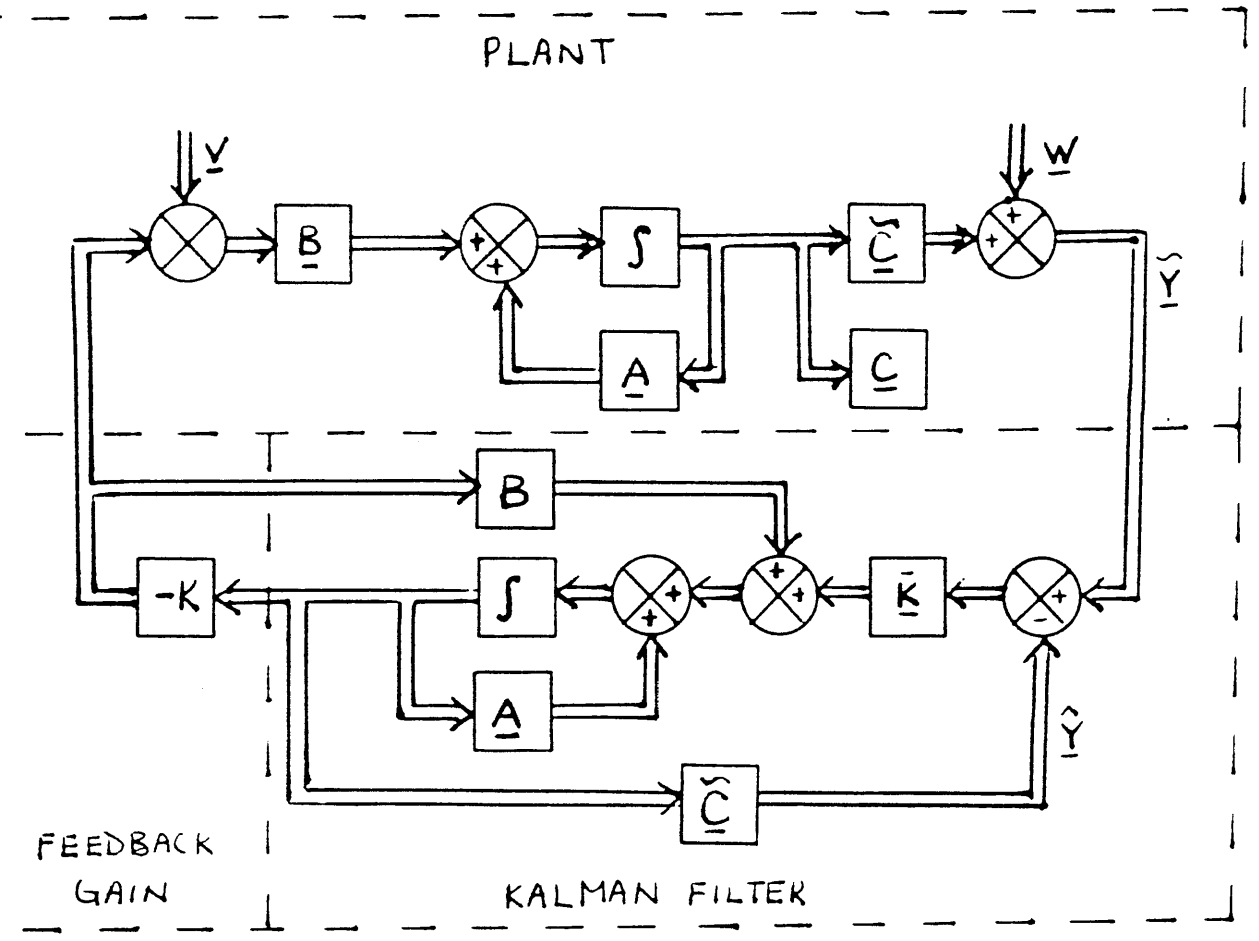


Figure 4 Block Diagram of the Plant-Estimation-Control System

The estimated value of $\hat{Y}$ can be obtained by the equation :

$$\hat{Y} = \hat{X}\,\tilde{C} \qquad (1.20)$$

As shown in the diagram, the difference ($\tilde{Y}$ - $\hat{Y}$) between the measured output value and the estimated output value is fedback to the original system through the matrix $\bar{K}$.

The Kalman filter is a time domain technique which performs optimal estimation of a system based upon the previous system estimates and the current system measurements. It produces an unbiased estimation with minimum variance and gives the best estimate of the state vector. Unlike an observer system, a Kalman filter is particular suitable for a noisy system i.e. in a stochastic case. On the other hand, a Kalman filter is similar to an observer system in the sense that the construction of both systems need a fairly exact knowledge of the original system being monitored.

1.3 Methods of Fault Detection based upon Signal Processing and System Identification

1.3.1 Out-of-Range Check

This is a method in which the output of the signal Y(t) is only allowed to vary within a prespecified range of values, i.e.

$$Y_{min} < Y(t) < Y_{max} \tag{1.21}$$

where Y_{max} and Y_{min} are the maximum and minimum values respectively within which the value of Y(t) must lie at all times. When considering the value Y_{max} and Y_{min}, one must make sure that the range of validity $(Y_{max} - Y_{min})$ is large enough to retain any appearance of fault and small enough so that the fault can be detected at an early stage.

This technique can also be extended to check the rate of change or the trend of the signal.

i.e. $$\dot{Y}_{min} < \dot{Y}(t) < \dot{Y}_{max} \tag{1.22}$$

where $\dot{Y}_{min}$ and $\dot{Y}_{max}$ are the minimun and maximum values of $\dot{Y}(t)$ respectively.

1.3.2 Signal Processing Technique using Band-Limited Filter

Corbin and Jones[8] have established a fault detection techniques using a band-limiting filter. This technique avoids the use of fully duplicated systems thus reducing the hardware needed for monitoring faults. The idea is to apply a series of ramp input signals. By passing the input signal through an increment detector or by passing the output signals of the systems being tested through a filter, a band-limited response can be obtained. This response could then be analysed by correlating it with another response signal. If faults occur in a system, its reponse changes, thus changing the correlation in a detectable way.

The input signal is a series of ramp inputs with different amplitudes and lengths. The amplitude is related to the length by a one third power law[8].

There are two types of fault monitoring schemes being considered : one approach is to relate the measurement of the input to the response of the system, another type is to compare two consequential signals which are the responses of the outputs from two duplicate systems. Both types of systems will be discussed in the following.

Type 1 system : A block diagram of a first type system is shown below :

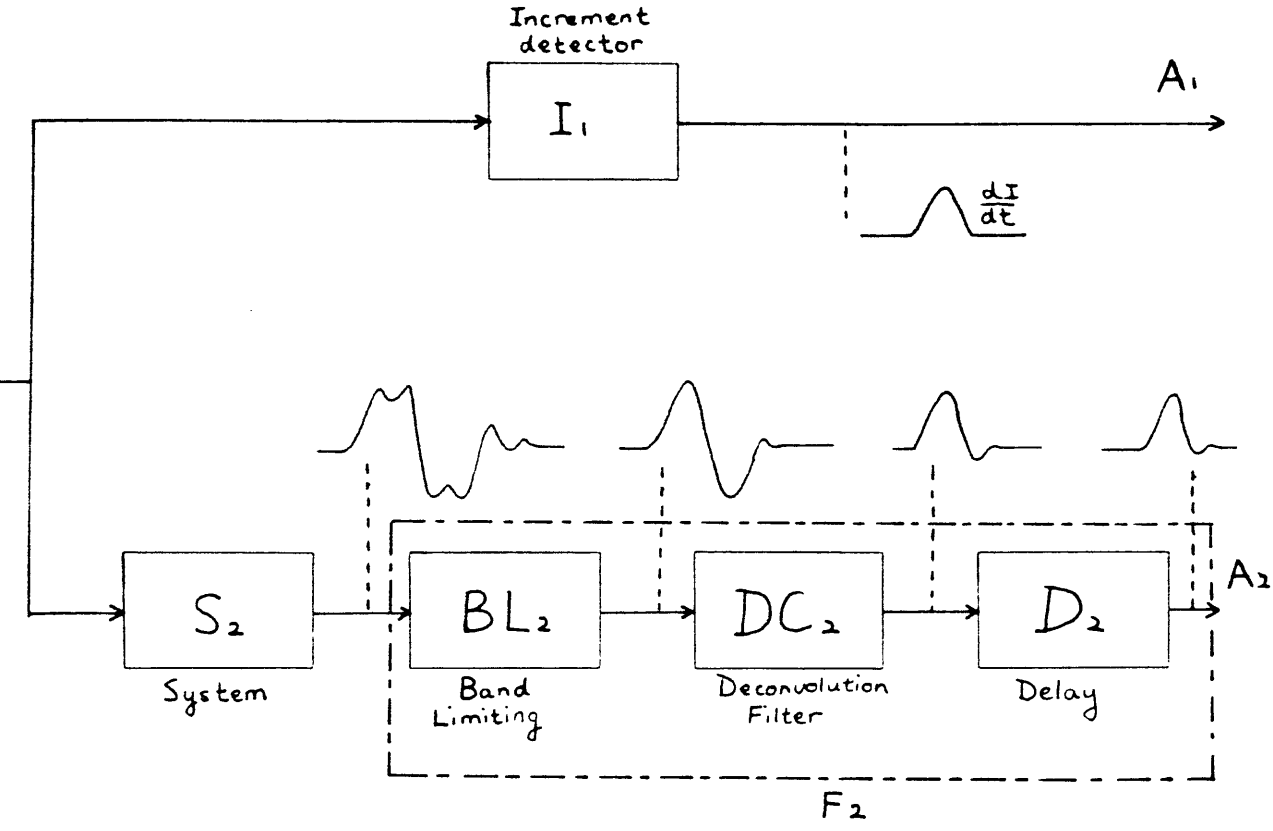


Figure 5 Block Diagram of a Fault Detection Scheme using Band-limited Filter

As shown in the diagram above, a ramp input signal is applied to a system (S_2) and also to a increment detector (I_1). The output of the system is passed through a filter (F_2). The output (A_1) from the increment detector is then compared with the output (A_2) from the filter. The result of the comparison can determine the health of the system (S_2). The system (S_2) and the filter (F_2) can be combined and considered as another increment detector (I_2).

The increment detector (I_1) is specially designed so that it will convert the ramp input signal into a smooth pulse which will resemble the output of the filter (A_2) as much as possible. Currently, the design of the increment detector (I_1) is based on practical experience only.

The output of the system is first passed through a band-limited filter to obtain a smoother waveform. It is then passed through a deconvolution filter which eliminates all the overswings of the output signal apart from its initial smooth pulse. A time delay is introduced to the signals A_1 and A_2 to minimise the phase difference between them.

The signal A_2 is then subtracted from signal A_1 to produce an error signal. This error signal can therefore give an indication of the condition of the system being monitored. For a fault free system, the error envelope should not exceed 15% of the peak amplitude of the response. By setting a threshold at say 20%, this gives good probability of detecting a fault without a high resultant transient.

Type 2 system : there are two systems being monitored. The signals from the system are passed through a different set of bandlimited filters and deconvolution filters so that another smooth single-pulse waveform can be obtained and compared. A block diagram of a system of this type is shown below:

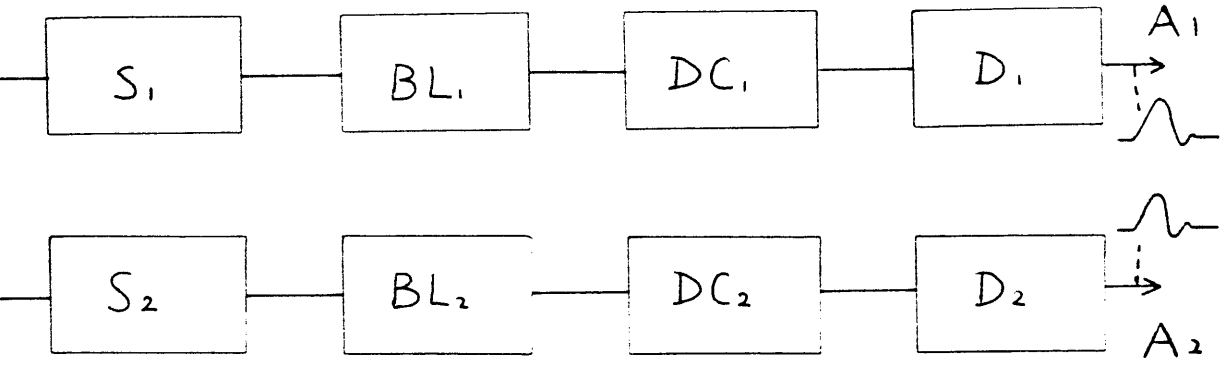


Figure 6 Block Diagram of a Fault Detection Scheme for two duplicated systems using Band-limited Filter

1.3.3 Fault Detection by System Identification

System identification[20] is a process by which a system can be characterized, thus providing information about the system performance. The system characteristic can be defined by its impulse response in the time domain or by its transfer function in the frequency domain. System identification can be applied to system fault detection because any system fault will alter the system response. This alteration of system characteristic can provide a useful source of information to identify and locate the system fault.

In the later chapters, fault detection techniques using system identification which are based upon frequency domain approaches will be discussed. The approaches are emphasised on the situation when the structure of the system being monitored is not precisely known. In the other words, the system is presented as a 'black box' component. Under such circumstances, the fault detection technique can only be based upon the input and output signals of the system.

CHAPTER TWO

2. FREQUENCY RESPONSE APPROACH TO SYSTEM IDENTIFICATION

2.1 Introduction

Various fault detection techniques have been developed and widely used in the past. They are mainly confined to the time-domain analysis. The use of an observer system and Kalman filter are two prime examples of this sort. However, fault detection techniques based upon the frequency domain do not appear to have received similar attention.

Frequency domain analysis remains a powerful tool for system identification and design. It can provide clear visualisation of the effect of noise disturbance and parameter variation. This can be applied usefully to fault detection techniques in which a parameter change can be thought of as a system fault. By observing the difference of the system frequency responses before and after the parameter variation, the parameter which causes the response change can be classified and identified. System identification techniques can be then be used to indicate or possibly determine which parameters cause the system response change.

The performance of a system can be observed from its response characteristic which can be expressed either in the time domain or in the frequency domain. The time domain response can be easily obtained at the system output by applying a standard test signal at the system input. The test signal can be an impulse, a step, a constant velocity or a constant acceleration.

To get the frequency response of a system, an input signal is injected into the system and an ouput signal is obtained. The input and output signals are Fourier transformed. The relationship between the two transformed signals gives the frequency response. From the system frequency response, various system specifications can be identified, e.g. resonance peak, resonance frequency, bandwidth, cut-off frequency etc. These specifications will indicate the behaviour and the condition of the system. The nature of the input signal used for obtaining the frequency response will be discussed in the next section

where three input signals including an impulse signal, a sinusoidal signal and a pseudo-random-binary-signal are compared and judged with respect to their merit on the derivation of the system frequency response. In the subsequent chapter, a fault detection technique based upon the frequency domain approach will be discussed.

2.2 Discussion of Three Input Signals for Frequency Response Analysis

To carry out frequency response analysis, the system is excited by an input signal and the output signal is obtained. The signals are then Fourier tranformed and studied. In order to obtained a good frequency spectrum, the input signal should have the following properties :

1) a wide bandwidth which covers at least the frequency range of interest. The signal should have sufficient energy density to ensure an acceptable signal to noise ratio over the relevant frequency range.

2) a low peak to average power ratio. This is to avoid any nonlinearities when operating the system.

3) a zero mean value. This is of particular importance for a system with a free integrator, e.g. type 1 system. Here a small bias in the test input would cause gross error in the estimate.

There exist a number of signals with such properties. The comparison of a sine wave, an impulse and a pseudo random binary sequence will be discussed.

2.2.1 Sinusoidal Signal

A sine wave satisfies properties (2) and (3). However, a sine-wave of frequency F Hertz can give a good estimation of the system response at F Hz only. To cover a whole range of frequencies, the estimation procedure needs to be carried out repeatedly using a sine-wave of different frequencies. The whole process could be very time consuming.

2.2.2 An Impulse Function

An impulse signal satisfies property(1) only. Its high peak to average power ratio may cause the system to run into non-linearity, and its non-zero mean value may cause error in a system which has one or more free integrators. It is important to note that an impulse signal has a flat magnitude spectrum and a zero phase spectrum which are two desirable features of an input signal.

The impulse signal can be expressed in the frequency domain as its complex transform, with real and imaginary components, $Re_i(f)$ and $Im_i(f)$ respectively. Alternatively, the frequency response can be expressed by a magnitude $Mag_i(f)$ and phase $Arg_i(f)$ components as shown below :

$$Mag_i(f) = \sqrt{Re_i(f)^2 + Im_i(f)^2} \qquad (2.1)$$

$$Arg_i(f) = \tan^{-1} Im_i(f) \,/\, Re_i(f) \qquad (2.2)$$

Similarly, the output response can be expressed as real and imaginary components, $Re_o(f)$ and $Im_o(f)$ respectively and

$$Mag_o(f) = \sqrt{Re_o(f)^2 + Im_o(f)^2} \qquad (2.3)$$

$$Arg_o(f) = \tan^{-1} Im_o(f) \,/\, Re_o(f) \qquad (2.4)$$

Then the system response G(jf) can be expressed as :

$$Mag_G(f) = Mag_o(f) \,/\, Mag_i(f) \qquad (2.5)$$

$$Arg_G(f) = Arg_o(f) - Arg_i(f) \qquad (2.6)$$

It should be noted that if the system input is a unit impulse function which has $Mag_i(f)$ equal to one and $Arg_i(f)$ equal to zero, then the system response can be derived simply from the output signal. However due to the disadvantage mentioned above, it is abandoned.

2.2.3 Pseudo Random Binary Sequence

Pseudo binary sequences[11] can be generated by means of a cascade of m electronic shift register stages connected in a feedback loop via an exclusive-OR gate.

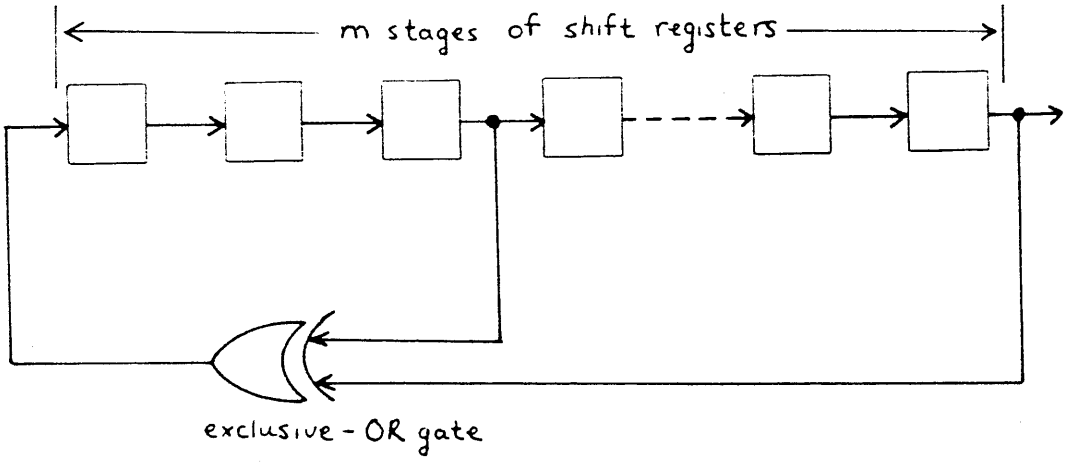


Figure 7 A Pseudo Random Binary Sequence Generator

An m stages shift register can produce a pseudo random binary sequence of 2^m-1 bits. As the signal always contains an odd number of bits, the average value of the sequence is always non zero. A pseudo random binary signal can be considered as a kind of white noise as it has a flat magnitude spectrum. The signal has a peak to average ratio of one. The pseudo random binary sequence therefore satisfies properties (1) and (2). However it has the disadvantage of non zero mean level[12] and non zero phase characteristics. A non-zero mean level can be overcome by adding one bit to the pseudo random binary sequence[13]. Moreover, a non-zero level would not matter for type-0 system. The phase spectrum of a pseudo random binary sequence takes up a widely scattered range of values. A pseudo random binary sequence is considered as the best of the three signals. The incompatibility of applying the radix-2 Fast Fourier Transformation, which needs 2^m data points, on a pseudo-random-binary-sequence, which has only $(2^m - 1)$ data points[14] will be discussed. A digital simulation of a PRBS generator is shown in appendix A.

2.3 Reduction of System Sensitivity to Parameter Variations in Closed Loop System

As mentioned in section 2.1, system may be monitored either in the time domain or in the frequency domain. When a change of system response occurs which exceeds a prespecified threshold level, a fault within the system is said to have been detected. This method is simple and straightforward. However, a problem will arise when dealing with a closed loop system.

The use of feedback in a closed loop system will reduce the sensitivity of the system to parameter variations. In fact the reduction of the system sensitivity to parameter variations is considered as an important advantage of a closed loop system. The feedback in a closed loop system will mitigate the effect of parameter variation so that the system output may not vary significantly with even for large parametric changes.

As far as fault detection is concerned, the reduction of system sensitivity in a closed loop system is not desirable. This is because this property will conceal any fault which occurs within the system. A closed loop system will only give a vague indication of parameter variation which makes the task of fault detection difficult. The following will show the reduction of system senstivity to parameter variation in a closed loop system compared with that in an open loop system.

Consider an open loop system :

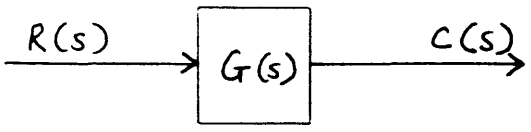


Figure 8 An Open Loop System

The Laplace transform of the system output can be written as :

$$C(s) = G(s)\,R(s) \qquad (2.7)$$

Supposing there is a change within the system G(s), the system now becomes

$$[\ G(s) + \Delta G(s)\ ]$$

The output of the system will subsequently change to

$$[\ C(s) + \Delta C(s)\ ]$$

therefore $C(s) + \Delta C(s) = R(s)\ [\ G(s) + \Delta G(s)\ ]$ (2.8)

or $\Delta C(s) = R(s)\,\Delta G(s)$ (2.9)

In all practical situations, the output will show a change which is proportional to the change of the element within the system.

Now consider a closed loop system as shown in the diagram below :

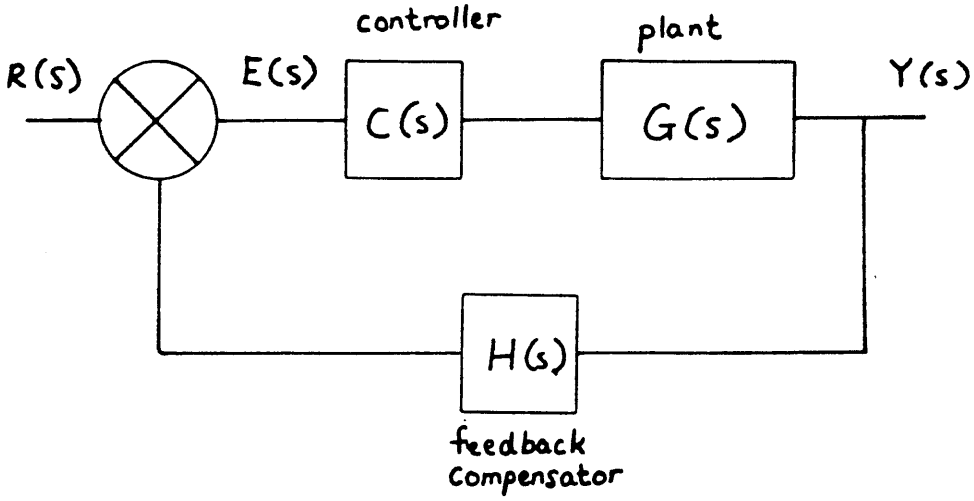


Figure 9 A Closed Loop System

so. $$C(s) = \frac{G(s)}{1 + G(s)H(s)}\,R(s)$$ (2.10)

If there is any change of parameters with G(s) then the system output will become :

$$C(s) + \Delta C(s) = \frac{G(s) + \Delta G(s)}{1 + [\, G(s) + \Delta G(s) \,]\, H(s)} R(s) \quad (2.11)$$

Then $|G(s)| >> |\Delta G(s)|$ then (2.11) can be approximated to

$$C(s) + \Delta C(s) = \frac{G(s) + \Delta G(s)}{1 + G(s)H(s)} R(s) \quad (2.12)$$

so

$$\Delta C(s) = \frac{\Delta G(s)}{1 + G(s)H(s)} R(s) \quad (2.13)$$

By comparing equations (2.13) and (2.9), it can been seen that the output of the closed-loop system due to variation in G(s) is reduced by [1 + G(s) H(s)] in comparison with the open loop case. In most practical situations [1 + G(s) H(s)] is much greater than unity thus reducing the system sensitivity to parameter changes considerably.

2.4 Application of Correlation to Frequency Response Analysis

Due to the reduction of system sensitivity to parameter variation in a closed loop system, fault detection techniques which are based upon the change of system response on parameter variation are difficult to apply. Therefore it is essential to increase the system sensitivity to parameter change. To do this, one can recover the open loop system response from a closed loop system. Observation can then be made on the open loop system which is independent to the feedback of the closed loop system, thus no reduction of system sensitivity to parameter change will occur. Wellstead[9] has applied the spectrum of auto-correlation and cross-correlation to obtain the open loop system response from a closed loop system which is corrupted by noise.

The auto-correlation of a function x(t) is designated by $R_{xx}(\tau)$ and is defined as :

$$R_{xx}(\tau) = \lim_{T\to\infty} 1/T \int_{-T/2}^{T/2} X^*(t)X(t+\tau)dt \qquad (2.14)$$

The auto-corrlation $\phi_{xx}(f)$ is defined as the Fourier transform of the auto-correlation function $R_{xx}(\tau)$. Therefore $\phi_{xx}(f)$ can be written as :

$$\phi_{xx}(f) = \int_{-\infty}^{\infty} \exp(-j2\pi f)R_{xx}(\tau)d\tau \qquad (2.15)$$

The correlation of two functions, x(t) and y(t) is designated by $R_{xy}(\tau)$ and is defined as :

$$R_{xy}(\tau) = \lim_{T\to\infty} 1/T \int_{-T/2}^{T/2} x^*(t)y(t+\tau)d\tau \qquad (2.16)$$

The cross-spectrum $\phi_{xy}(f)$ can be defined as the Fourier Transform of the cross-correlation $R_{xy}(\tau)$. $\phi_{xy}(f)$ can therefore be written as :

$$\phi_{xy}(f) = \int_{-\infty}^{+\infty} \exp(-2j\pi f)R_{xy}(\tau)d\tau \qquad (2.17)$$

Consider now a closed loop system whose output z(t) is corrupted by a noise signal n(t) as shown in the diagram below:

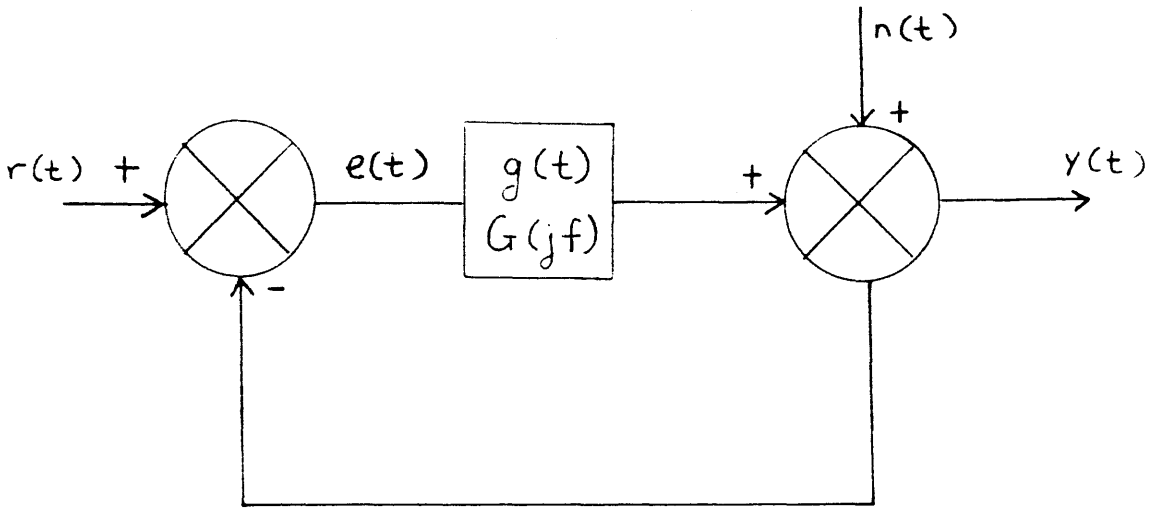


Figure 10 A Closed Loop System Corrupted by a Noise Signal

G(jf) is the transfer function of the system.
g(t) is the impulse response of the system.
r(t) is the input signal.
z(t) is the actual output of the system.
y(t) is the measurable output which has been corrupted by noise.
n(t) is the noise signal.

The direct approach of obtaining G(jf) from y(f) and r(f) can be disrupted by the noise n(f). However by using auto-correlaion and cross-correlation techniques, the open loop transfer function can be recovered from a closed loop system even though some noise is present within the system.

By using correlation techniques, the following equation is obtained :

$$\phi_{ry}(jf) = G(jf)\phi_{re}(jf) + \phi_{rn}(jf) \qquad (2.18)$$

When the input signal and the noise signal are orthogonal, $\phi_{rn}(jf)$ is equal to zero. G(jf) can then be written as :

$$G(jf) = \phi_{ry}(jf) / \phi_{re}(jf) \qquad (2.19)$$

From the above analysis, we can conclude that correlation techniques provide a means of obtaining the transfer function of an open loop system from a close loop system response which is corrupted by a noise signal. Thus open loop system information can be obtained from a closed loop system. Furthermore the sensitivity of the open loop system would not be reduced by the feedback of the closed loop system. Fault detection can then be carried out on the open loop system with better indication of open loop system response changes to parameter variations which represent system faults.

2.5 Evaluation of the Squared Coherency Spectrum

The use of the auto-spectrum and the cross-spectrum can be extended to the the evaluation of the squared coherency signal[15]. The squared coherency function $\gamma_{xy}(f)$, a measure of the interdependence of two functions, can be written as :

$$\gamma^2_{xy}(f) = \frac{|\phi xy(f)|^2}{\phi xx(f) \quad \phi yy(f)} \qquad (2.20)$$

A block diagram of system G(jf) is shown in the diagram below. The system output z(t) is corrupted by an additive noise signal n(t).

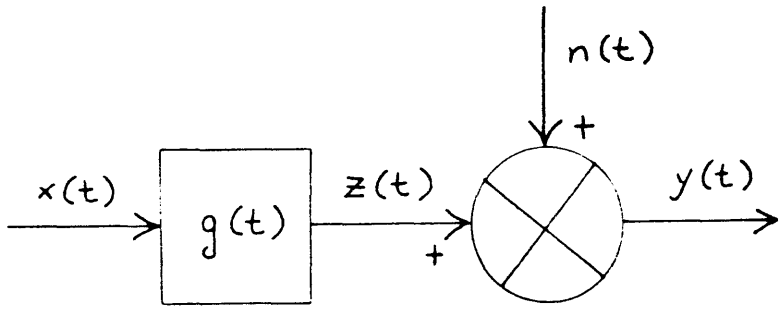


Figure 11 Block Diagram of a System Corrupted by Noise Signal

The coherency spectrum $\gamma^2_{xy}(f)$ can be expressed in terms of the autospectrum of the noise signal and the noise-free output signal. Therefore :

$$\gamma^2_{xy}(f) = \left[1 + \frac{\phi_{nn}(f)}{\phi_{zz}(f)} \right]^{-1} \qquad (2.21)$$

It can be seen from the above expression that when the noise spectrum $\phi_{nn}(f)$ is much greater than the estimated output spectrum $\phi_{zz}(f)$, the squared coherency spectrum $\gamma^2_{xy}(f)$ will approach zero. Under such circumstances, the measured output spectrum resembles the noise spectrum. When the squared coherency is zero, the measured output y(t) consists entirely of the noise signal. On the other hand, when the estimated output spectrum is greater than the noise spectrum then the squared coherency will approach one. When the squared coherency is one, the measured output spectrum is simply the input spectrum multiplied by the square of the gain of the system.

The squared coherency gives a measure on confidence on the measured system output z(t). The squared coherency shows the reliability of the measured system reponse. Without such ground for confidence, one could be totally misled by the measured system response.

2.6 The Radix-2 Fast Fourier Transform (FFT) and the Pseudo Random Binary Sequence (PRBS)

The radix-2 fast Fourier transform is the fastest and the most efficient Fourier transform. It can be employed to obtain a discrete estimate of the input and output signal spectrum, so that obtain the frequency response of the system can be obtained form the ratio of these two quantities directly. However problems arise when the radix-2 FFT is used in conjunction with a PRBS signal.

A PRBS, which is generated from a circuit of m stages shift registers, has 2^m-1 samples. This conflicts immediately with the use of simple radix-2 FFT which requires 2^m samples. Lamb and Rees[14] attempted to overcome this incompatibility by adding a further data point to the PRBS but they only got poor results. Golten[12] has shown since that an augumented PRBS can have a zero mean value which is of importance when testing a type-1 system.

Singleton[16] has introduced a less efficient mixed radix fast Fourier transformation algorithm which does not required 2^m samples. As a result this mixed radix algorithm is chosen to carry out the Fourier transformation of the PRBS of 2^m-1 samples and the frequency spectrum of an unaugmented PRBS is obtained.

Later Poussart and Ganguly[13] solved the problem of incompatibility between the radix-2 FFT and PRBS by simply changing the sampling clock to synchronise the PRBS for efficient radix-2 FFT transformation. Their method will be discussed in the next section.

2.7 Synchronisation of Pseudo Random Binary Sequence (PRBS)

As already discussed, pseudo random binary sequences, which consist of 2^m-1 bits, are not compatable with the computation of the radix-2 fast Fourier transform that requires 2^m data points. Poussart and Ganguly[13] solved this problem by introducing a time expansion and compression technique. Their idea was to generate 2^m data points (covering a period of T seconds) from a 2^m-1 bit PRBS signal (also covering a period of T seconds). This is done by asynchronising the clock rate which generates the PRBS and the sampling clock rate. The following diagram shows how the two different clock rates are generated in the case when m = 9.

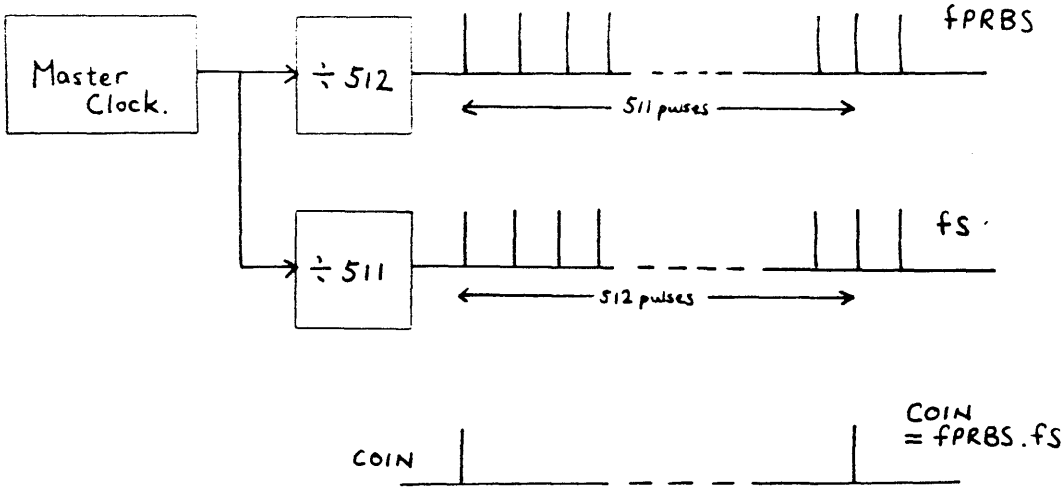


Figure 12 Asynchronous Sampling of PRBS Signal and Data Signal

The PRBS clock fPRBS and the sampling clock fs are derived from the master through two division chains, ÷ 512 and ÷ 511 respectively as shown in the diagram above. By using these two different clock rates, the period of 512 data samples would be matched with the period of 511 PRBS data. Having obtained 512 samples which can be expressed in the form of 2^m, the radix-2 fast Fourier transform can then be applied on those data. It is important to maintain the phase relation between the sampled data and the PRBS data because a small phase error may cause an unacceptable error at the output. A coincidence flag, as shown in the diagram above, is used to ensure that a sequence of 511 PRBS data matchesexactly with 512 sampled data points. It should be noted that every 511th PRBS value must coincide in time with every 512th data sample.

The discrete data can easily be obtained from an analogue signal by sampling at a frequency different from that of the PRBS signal. Unfortunately, in this project, due to limitations of the continuous system simulation program used for most of the work, it is not practical to generate a PRBS signal and a discrete signal with different sampling periods. For this reason, this method of introducing different clock rates is not applicable thus radix-2 FFT has not been used. Instead the mixed radix-2 FFT by Singleton is used although it is less efficient than the radix-2 FFT.

2.8 Aliasing Effect

The digitisation of continuous signals introduces aliasing error which may cause unacceptable deformation of the signal. An illustration will be shown below.

Consider a train of impulse functions which can be represented in the form :

$$v(t) = \sum_{n=-\infty}^{\infty} \delta(t - nT) \qquad (2.22)$$

where T is the sampling period.

The sampled data of the continuous signal x(t) can be written as :

$$x^*(t) = \sum_{n=-\infty} \delta(t - nT)\ x(t) \qquad (2.23)$$

Equation 2.22 has an exponential Fourier series representation :

$$v(t) = \sum_{n=-\infty}^{\infty} F_n e^{j2\pi nf_s t} \qquad (2.24)$$

where $f_s = 1/T$ is the sampling frequency in Hertz.

and $$F_n = 1/T \int_0^T v(t)\ e^{-j2\pi nf_s t}\, dt \qquad (2.25)$$

It is known that the area of an unit impulse function is one,

therefore $$\int_0^T v(t)\ e^{-j2\pi nf_s t}\, dt = 1 \qquad (2.26)$$

and $$F_n = 1/T \qquad (2.27)$$

Substituting (2.27) into (2.25) gives

$$v(t) = 1/T \sum_{n=-\infty}^{\infty} F_n \, e^{j2\pi n f_s t} \tag{2.28}$$

The discrete signal of the continuous signal x(t) is,

$$x^*(t) = \sum_{n=-\infty}^{\infty} v(t)\, x(t) \tag{2.29}$$

$$= \sum_{n=-}^{\infty} \delta\,(t - nT)\, x(t) \tag{2.30}$$

Now performing a Laplace transformation on (2.30),

$$X^*(s) = 1/T \sum_{n=-\infty}^{\infty} X(\, s - j2\pi n f_s \,) \tag{2.31}$$

The sinusoidal representation can be written as

$$X^*(jf) = 1/T \sum_{n=-\infty}^{\infty} X\,[\, j\,(f - nf_s)\,] \tag{2.32}$$

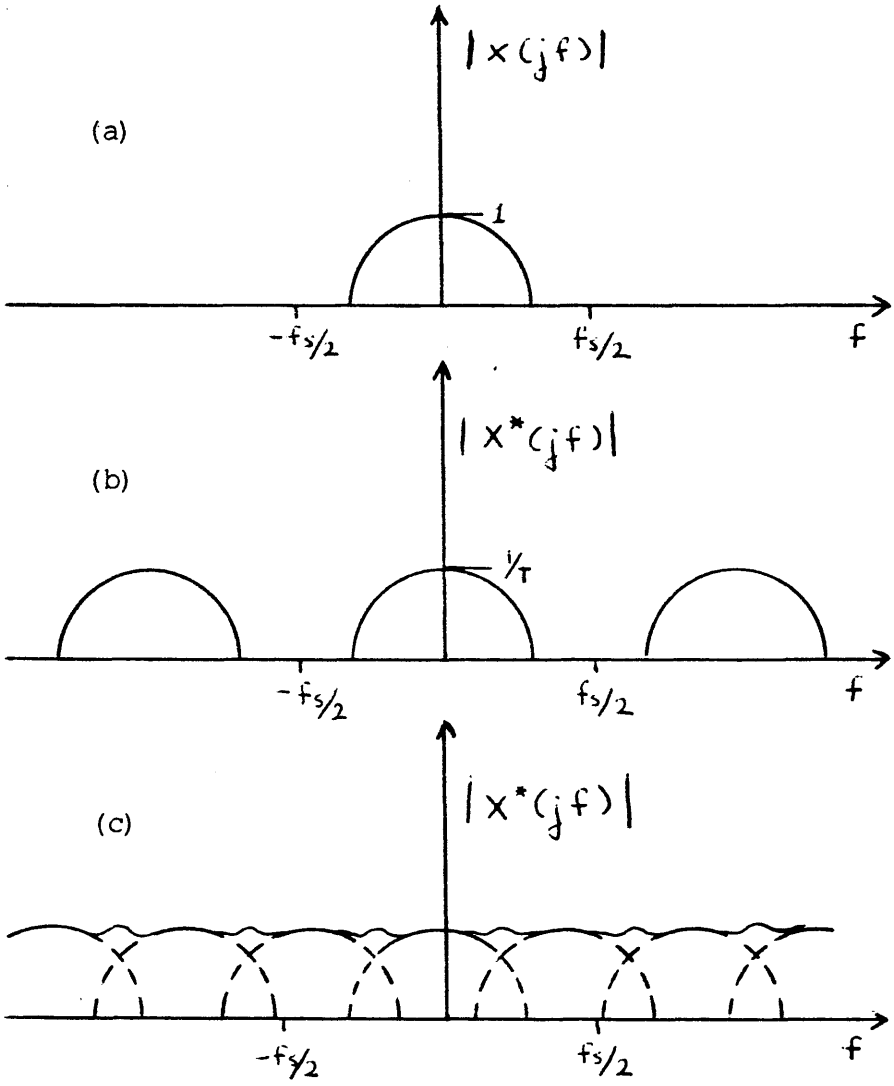


Figure 13(a) Spectrum of a Continuous Signal
13(b) Spectrum of sampled data signal when $f_s > 2B$
13(c) Spectrum of sampled data signal when $f_s < 2B$

It can be seen in figure 13 that spectral density of the sampled data is weighted by an amplitude of 1/T compared with the spectrum of the continuous signal within the original bandwidth. In addition the spectrun repeats itself periodically every w_s radians per second. It can be seen that if there is an increase in sampling rate, i.e. T decreases or f_s increases, then the replicas of the spectrum will move further apart from one another. However if the sampling rate decreases, i.e. f_s decreases, then the replicas will move closer together.

Inevitably a point is reached when the two adjacent replicas overlap due to the decrease in sampling rate. This point occurs when:

$$f_s / 2 = B \qquad (2.33)$$

where B is the bandwidth of the spectral (in Hz)

$$\text{or} \quad T = 1 / 2B \qquad (2.34)$$

At this point, all the replicas are just next to one another. Any slight decrease of sampling rate will lead to folding of the frequency spectra. This means that the original signal cannot be retrieved by filtering the sampling data. For this reason, it is generally a rule that :

$$T < 1 / 2B \qquad (2.35)$$

$$\text{or} \quad f_s > 2B \qquad (2.36)$$

i.e. the sampling frequency must be greater than twice the highest frequency component of the continuous signal.

The condition, T = 1/2B, gives the maximum sampling interval which can be used before any overlapping of spectral replicas occurs. In this situation T is known as the Nyquist interval, and the sampling frequency (1/T) is known as the Nyquist frequency.

The error caused by the overlapping of the spectral replicas is called the aliasing effect which can be eliminated by simply increasing the sampling rate. Methods of reducing the aliasing error will be discussed in the next section.

2.9 Aliasing Effect in Sampling

As shown in the previous section, if the original spectrum has any component which is above $f_s/2$, it will overlap with the adjacent replicas. Moreover, if it contains frequency components beyond $3f_s/2$, $5f_s/2$, ... , then overlapping with its second and third replicas will occur and so on.

In practice, this problem always arises because a time-limited signal is never completely bandlimited. There are always frequency components located above $f_s/2$ and overlapping of replicas is inevitable. The frequency component above $f_s/2$ will appear below this frequency. This may result in distortion of the original spectrum. One way to tackle this problem is to apply some sort of filtering before sampling. The filters used for this purpose are called pre-alias filters. One type of pre-alias filter is the Butterworth filter. It has a variable -3dB frequency and a variable attenuation rate. This filter can attenuate the frequency components beyond a desired frequency. The higher the order of the Butterworth filter used, the better the attenuation of the high frequency component obtained. However, it must be remember that it is practically impossible to construct an ideal low pass filter. In real cases, each filter has a finite slope at the band edge, thus transmitting the spectral replicas through the filter and distorting the original spectrum.

One other way to reduce aliasing error is to simply increase the sampling rate. When a continuous signal is sampled at a frequency f_s, overlapping will occur in the frequency range beyond $f_s/2$. However the spectral density decreases rapidly for frequencies higher than $f_s/2$. Therefore, by increasing the sampling rate, this has the effect of moving the replicas further apart. Although spectral overlapping still occurs it only involves the spectrum of very low amplitude and thus has a neglible distortion effect on the original spectrum.

Throughout the experiments which have been carried out in this research, the sampling rate has been selected to reduce any aliasing effect to a negligible amount. No filtering has been employed within the simulation studies carried out.

CHAPTER THREE

3 Parameter Dependence of Frequency Response of Closed Loop System

3.1 Introduction

A system fault may be simulated in some cases by a change of a system parameter. By identifying the effect of parameter changes on the system frequency response, the possible faults which could occur in the system can be classified. Each fault will have its unique frequency signature by which it can be identified. By using a model system each kind of fault and its corresponding frequency signature can be tabulated. With this information, future occurences of similar faults may be detected in a real system.

In the following, a two tank liquid flow control system is used as a test model to show the dependence of the system frequency response upon parameter changes. To carry out the test, different system parameters are changed. The system response before and after each parameter change will be recorded. Both the time domain approach and the frequency domain approach are applied out. The input signal used in the time domain approach is an impulse train. The input signal used for the frequency response test is a pseudo random binary sequence.

3.2 Test Model

A simple two tank liquid flow control is used as a test model. A simplified diagram of the system is shown below.

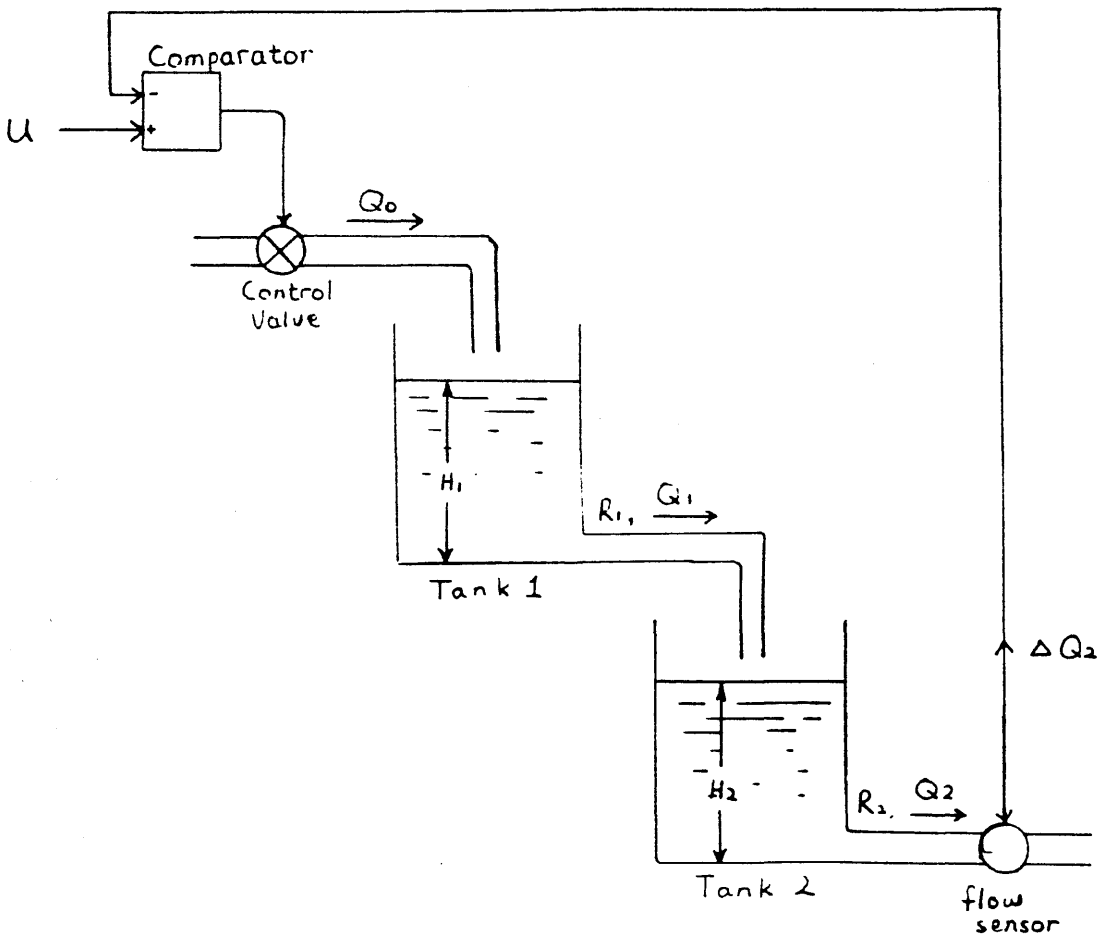


Figure 14 A Two Tank Liquid Flow Control System

where H_1 is the fluid head of tank 1,
H_2 is the fluid head of tank 2,
Q_0 is the liquid inflow rate into tank 1,
Q_1 is the liquid outflow rate of tank 1,
Q_2 is the liquid outflow rate of tank 2,
R_1 is the hydraulic resistance of the pipe from tank 1,
R_2 is the hydraulic resistance of the pipe from tank 2,
U is an input signal.

The system consists of two tanks which are arranged so that the outlet flow from the tank 1 is the inlet flow to tank 2. The system has an input liquid supply of flow rate Q_0. The flow rate of the input liquid supply (Q_0) is controlled by a valve. The opening of this valve is determined by an output signal from a comparator. This comparator compares an input signal (U) with the change of the outflow rate (ΔQ_2) from tank 2. A digital simulation of the two tanks system is shown in appendix B.

Assuming linearisation of the system, the change ΔQ_0 is related to U and ΔQ_2 by the equation :

$$\Delta Q_0 = K (U - \Delta Q_2) \qquad (3.1)$$

where K is a constant.

For reasons of simplicity, K is taken as one in the subsequent analysis. Therefore ,

$$\Delta Q_0 = U - \Delta Q_2 \qquad (3.2)$$

By applying the liquid flow rate balance equation :

Rate of accumulation of liquid in tank = inflow rate - outflow rate (3.3)

The system dynamic of tank 1 can be written as :

$$\Delta Q_0 = A_1 \frac{d \Delta H_1}{dt} + \Delta Q_1 \qquad (3.4)$$

where A_1 is the cross-sectional area of tank 1

and $\Delta Q_1 = \Delta H_1 / R_1$ (3.5)

The system dynamic of tank 2 can be written as :

$$\Delta Q_1 = A_2 \frac{d \Delta H_2}{dt} + \Delta Q_2 \qquad (3.6)$$

where A_2 is the cross-sectional area of tank 2

and $\Delta Q_2 = \Delta H_2 / R_2$ (3.7)

Taking the Laplace tranformation of equations (3.2), (3.4), (3.5), (3.6) and (3.7) the block diagram can be constructed as shown below.

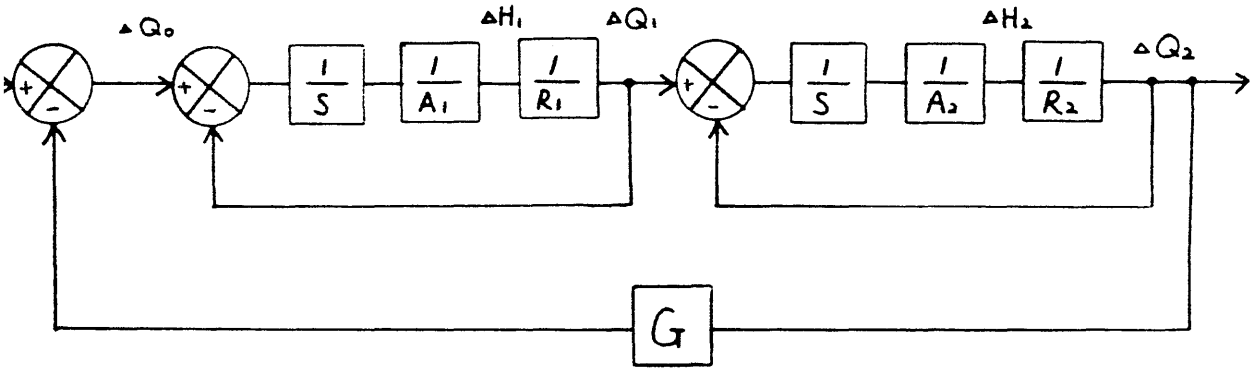


Figure 15 Block Diagram of a Two Tank Liquid Flow Control System

The values of the system parameters are specified as shown below:

$A_1 = 0.93\ m^2$
$A_2 = 0.93\ m^2$
$R_1 = 64.6\ s/m^2$
$R_2 = 226.0\ s/m^2$
$G = 1.0$

3.3 Results of PRBS Testing of System Simulation

The mixed radix-2 Fast Fourier Tranformation is used to obtained the system frequency response which is plotted in below.

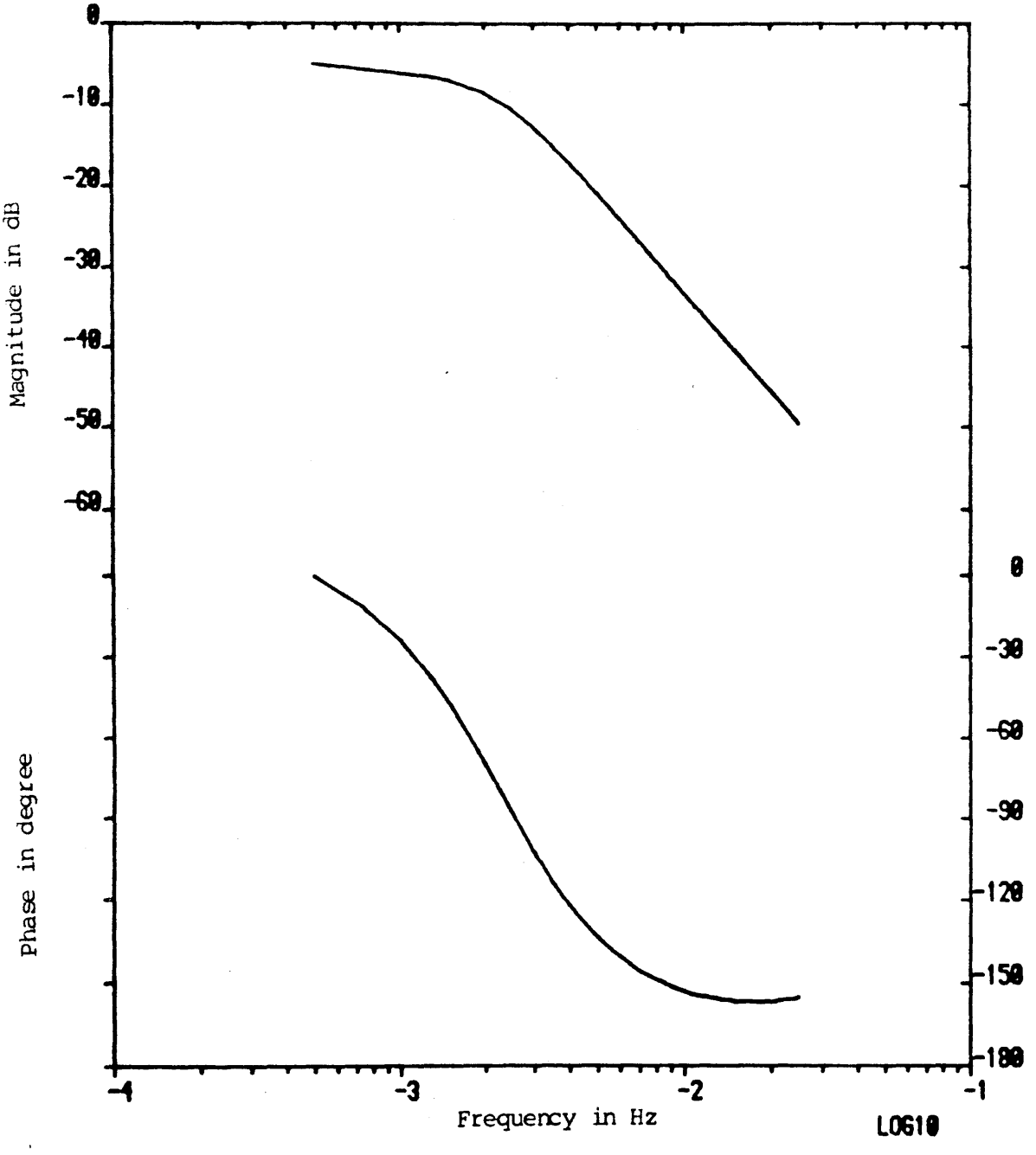


Figure 16 Spectra of the Frequency Response of the Two Tank Liquid Flow Control System

Each of the system parameters R_1, R_2 and G is increased by 100% at t = 3000s. The responses of the system are obtained for each case and are shown in figures 17-19. Then magnitude spectra before and after each parameter change are obtained and shown in figures 20-22. Likewise, the phase spectra are obtained and shown in figures 23-25.

Subsequently, the frequency responses before, during and after a parameter change are shown by a series of spectra. The series of magnitude spectra for each parameter change are shown in figures 26-28 while the phase spectra are shown in figures 29-31. The description of the figures 17-34 (on pages 47-61) are tabulated below.

Parameter change / Response	R1 increased by 100%	R2 increased by 100%	G increased by 100%
Impulse Response	Fig. 17	Fig. 18	Fig. 19
Magnitude spectra before and after a parameter change	20	21	22
Phase spectra before and after a parameter change	23	24	25
A series of magnitude spectra when a parameter changes	26	27	28
A series of phase spectra when a parameter changes	29	30	31

Finally, a number of system magnitude spectra are obtained. Each spectrum respresents the system response corresponding to a particular value of the system parameter. These spectra can be drawn as three dimensional surfaces, where the x-axis represents the frequency domain, the y-axis represents the value of a system parameter, and the z-axis represents the magnitude of the system reponse. These surfaces are drawn in figures 32-34 (on pages 62-64). Each surface corresponds to the variation of one parameter. In figure 32, the parameter R_1 varies from 4.0 to 120.0. In figure 33, the parameter R_2 varies from 20.0 to 600.0. In figure 34, the parameter G varies from 1.0 to 30.0.

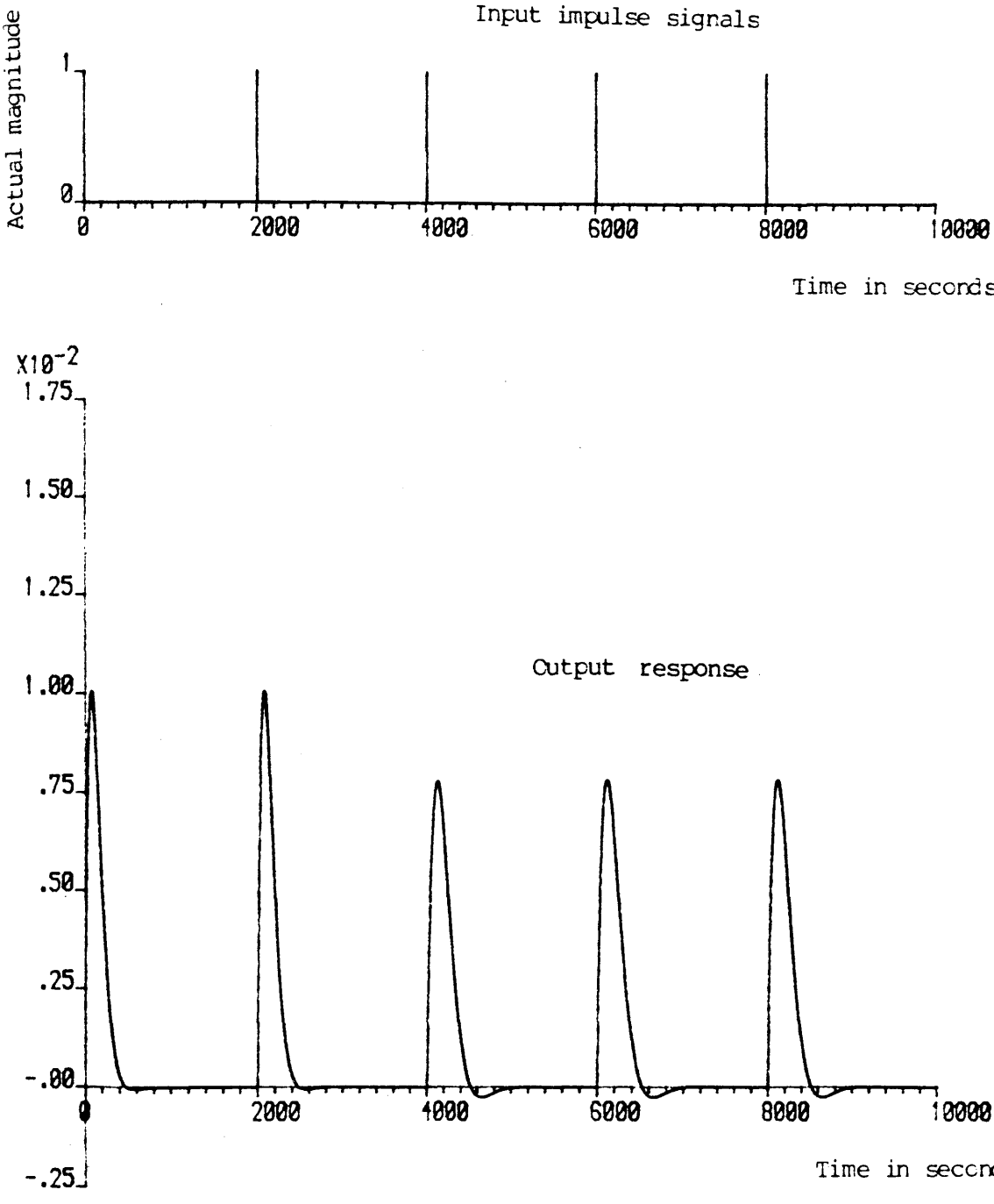


Figure 17 Impulse responses of the two tank liquid flow control system. The parameter R_1 is increased by 100% at t = 3000 sec.

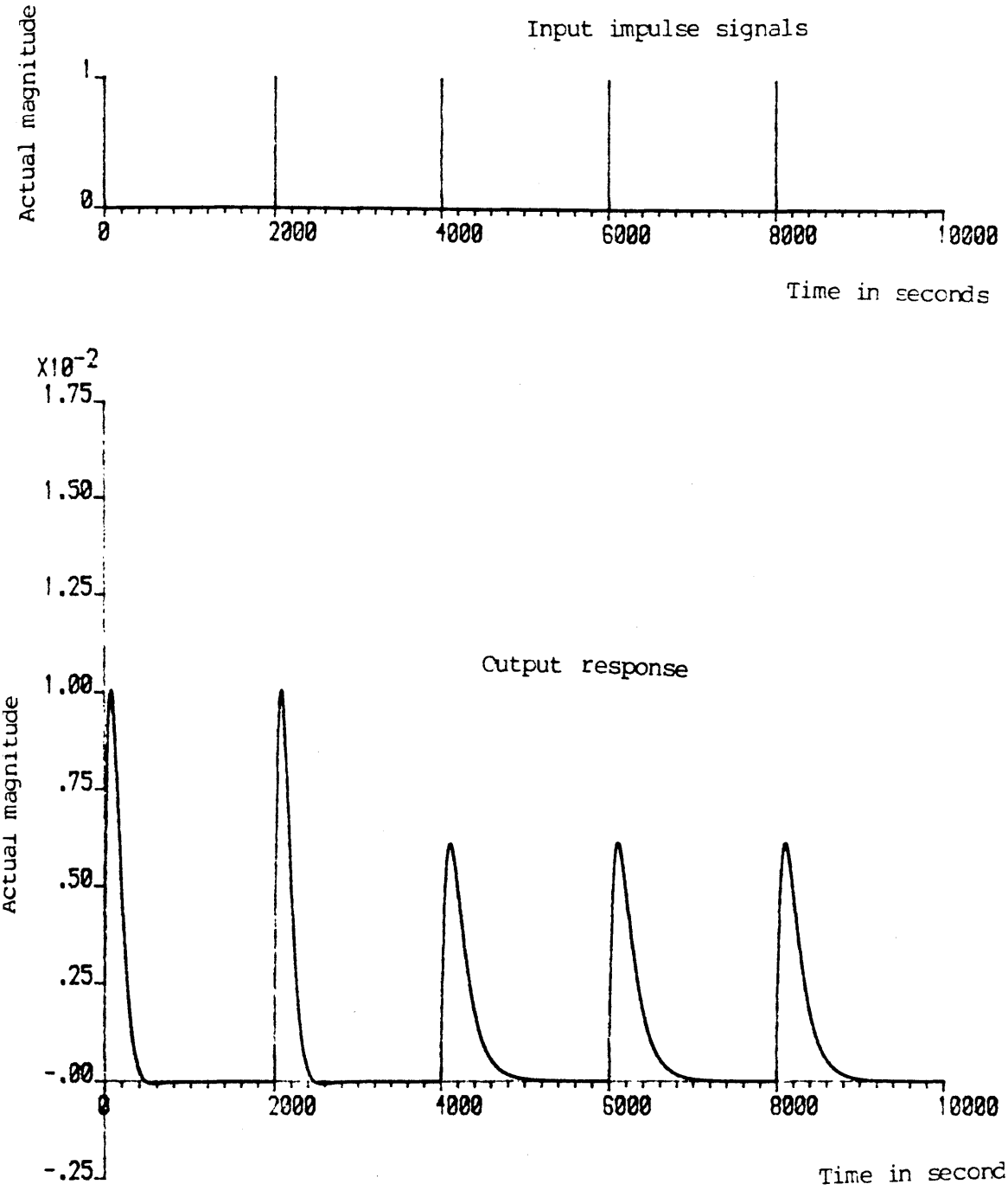
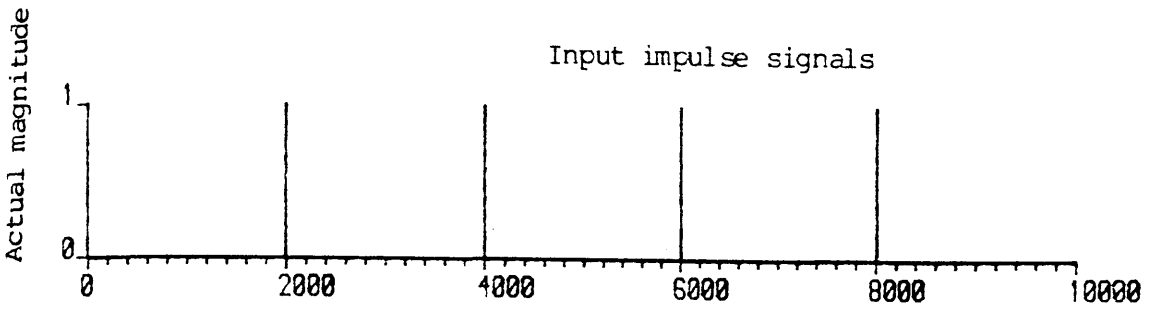


Figure 18 Impulse responses of the two tank liquid flow control system. The parameter R_2 is increased by 100% at t = 3000 sec.



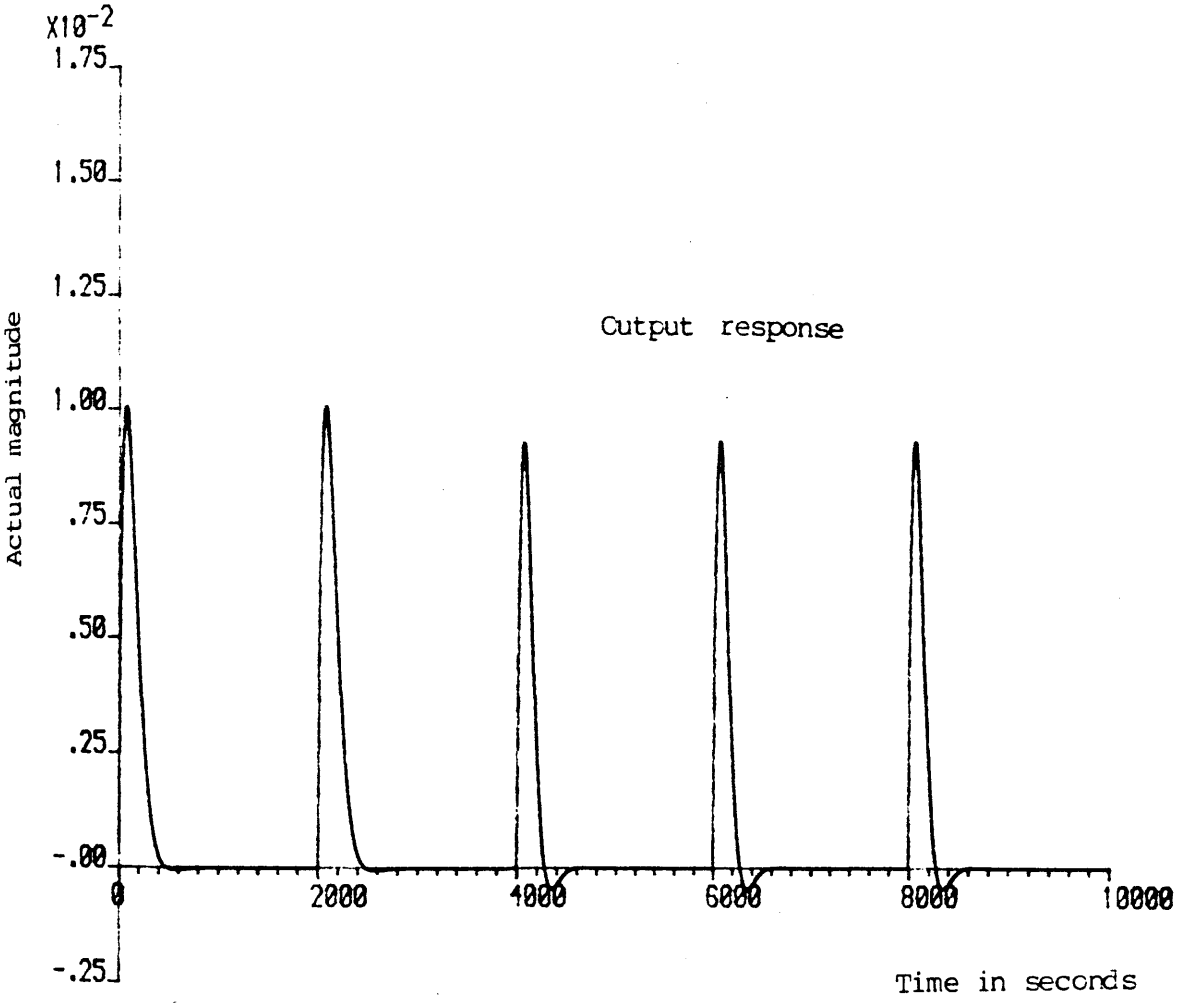


Figure 19 Impulse response of the two tank liquid flow control system. The parameter G is increased by 100% at t = 3000 sec.

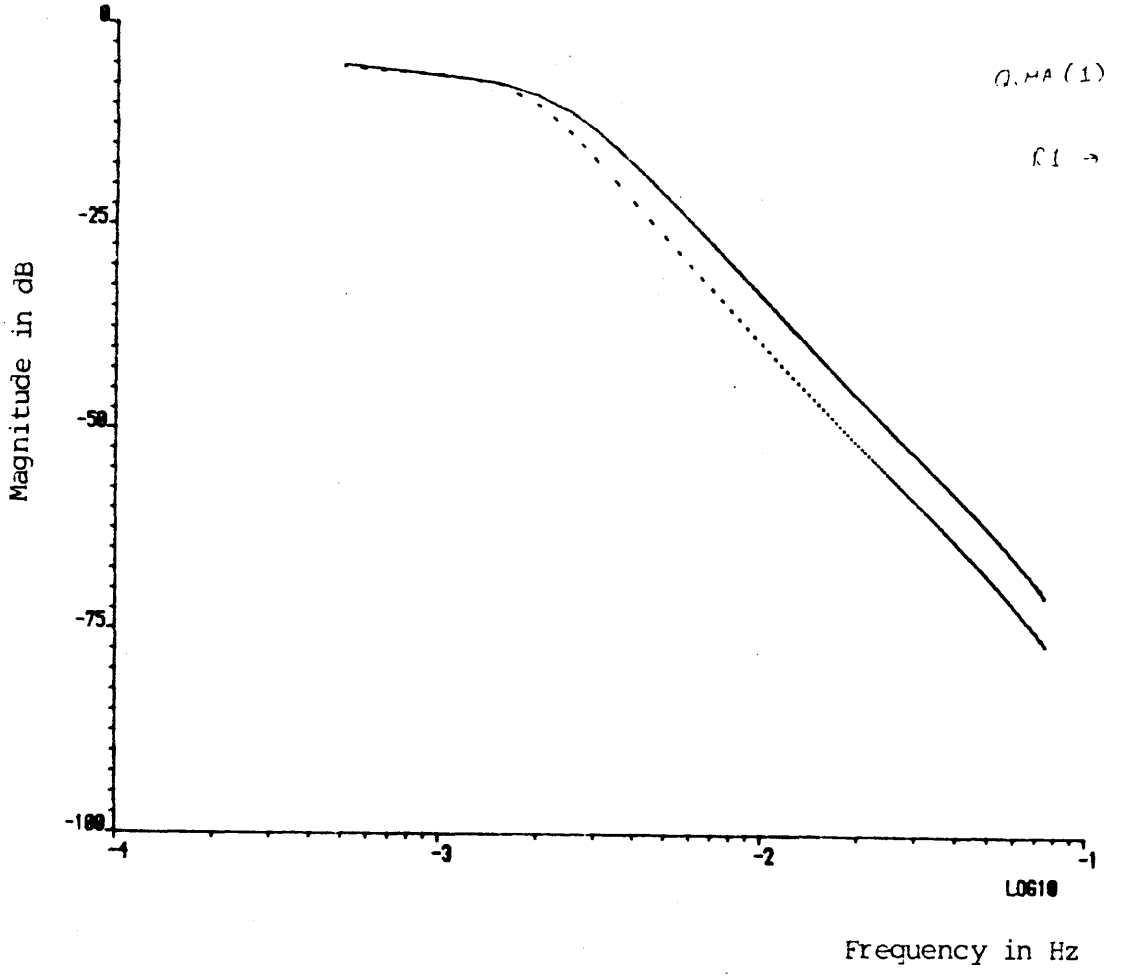


Figure 20 Magnitude spectra of the two tank system before and after the parameter R_1 is increased by 100%.

——— response before the parameter changes.
- - - - - response after the parameter has changed.

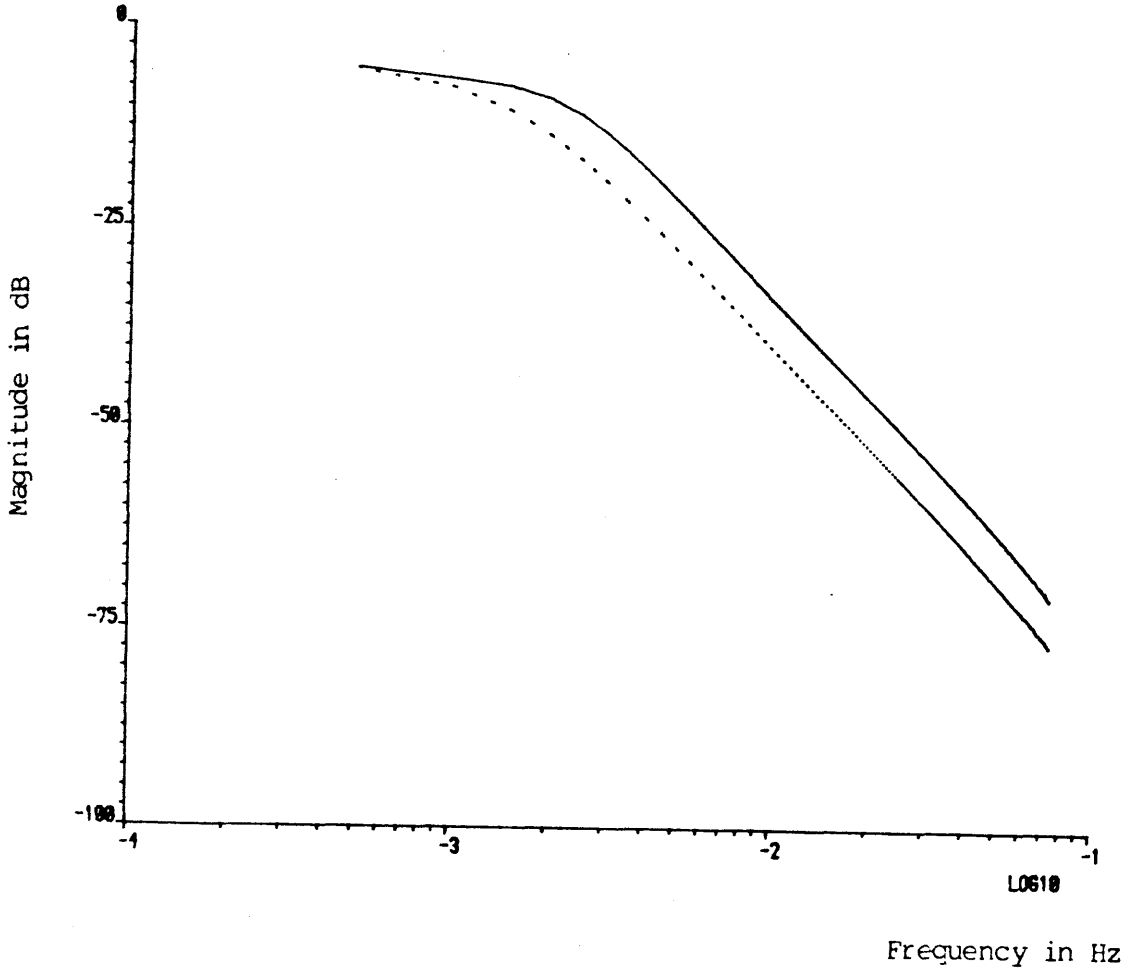


Figure 21 Magnitude spectra of the two tank system before and after the parameter R_2 is increased by 100%.

——— response before the parameter changes.
- - - - - response after the parameter has changed.

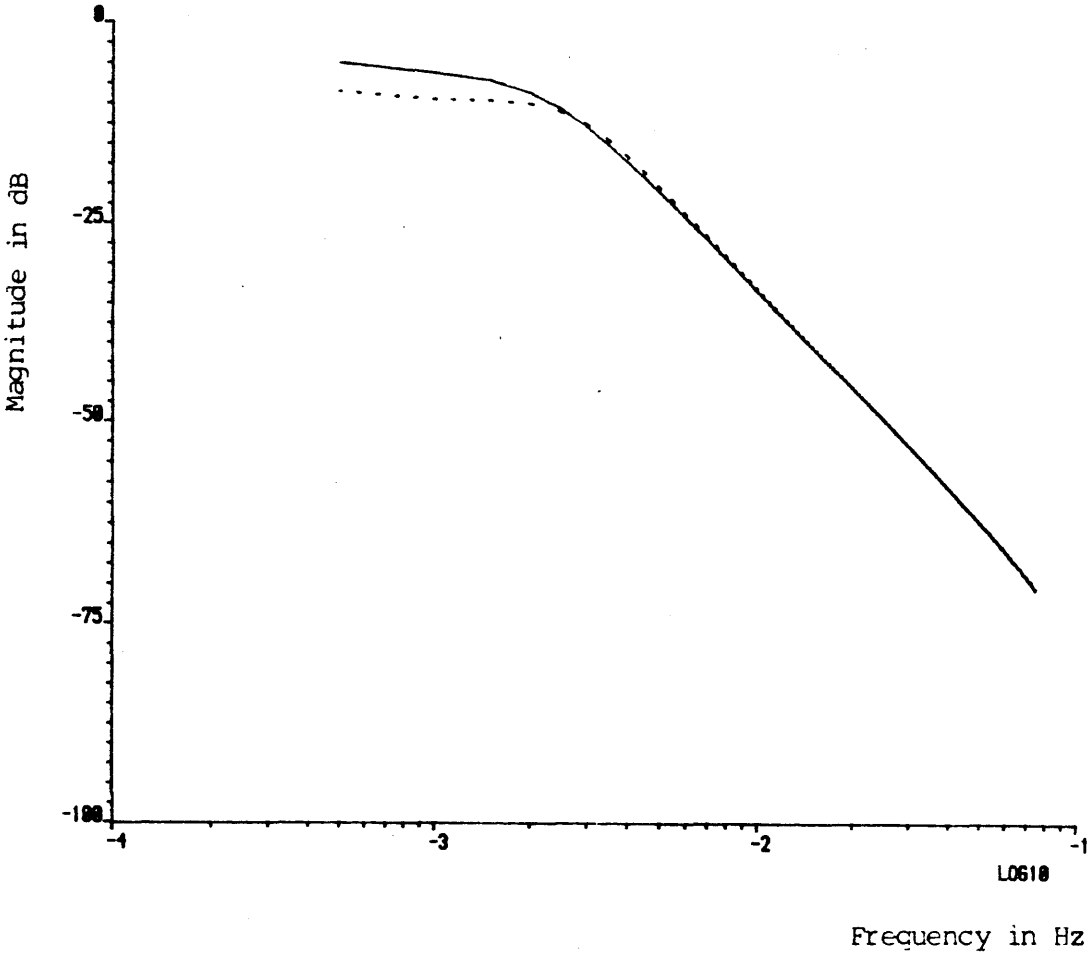


Figure 22 Magnitude spectra of the two tank system before and after the parameter G is increased by 100%.

——— response before the parameter changes.
- - - - - response after the parameter has changed.

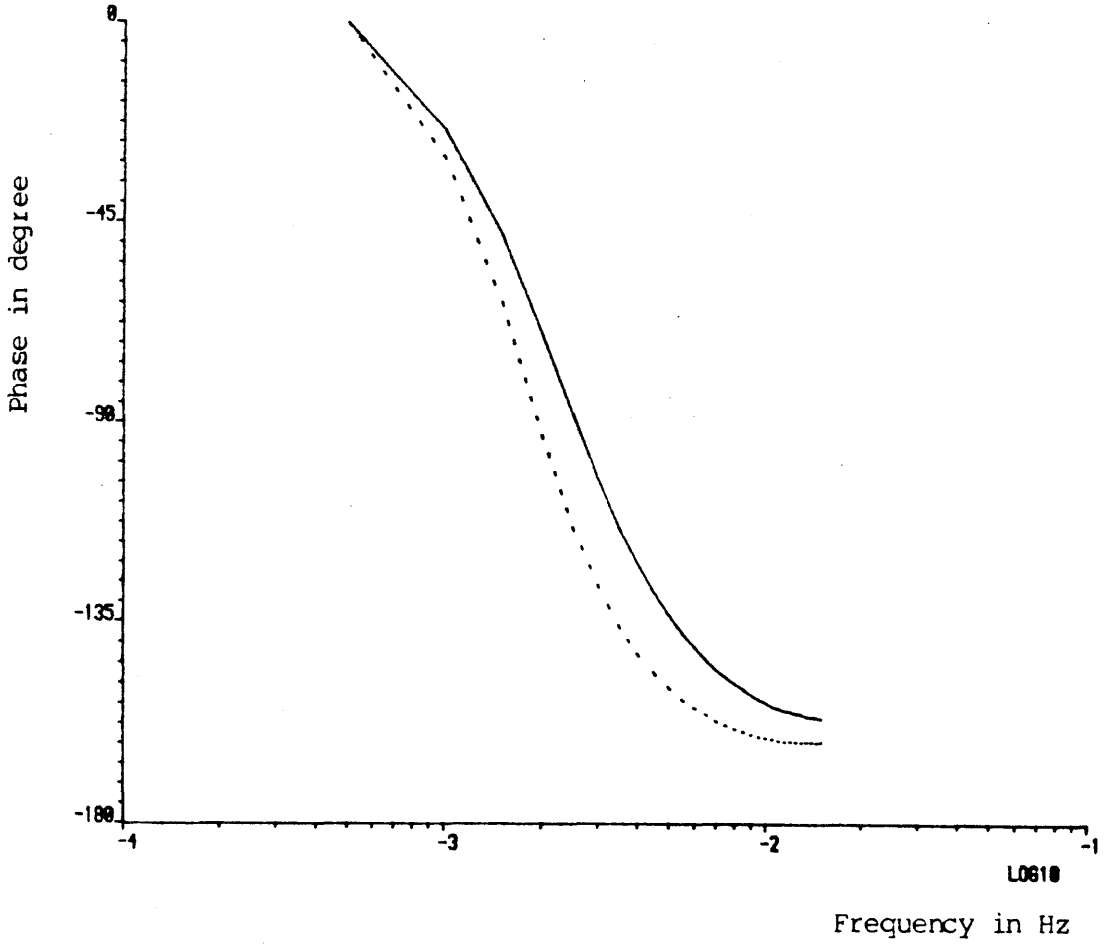


Figure 23 Phase spectra of the two tank system before and after the parameter R_1 is increased by 100%.

——— response before the parameter changes.
- - - - - response after the parameter has changed.

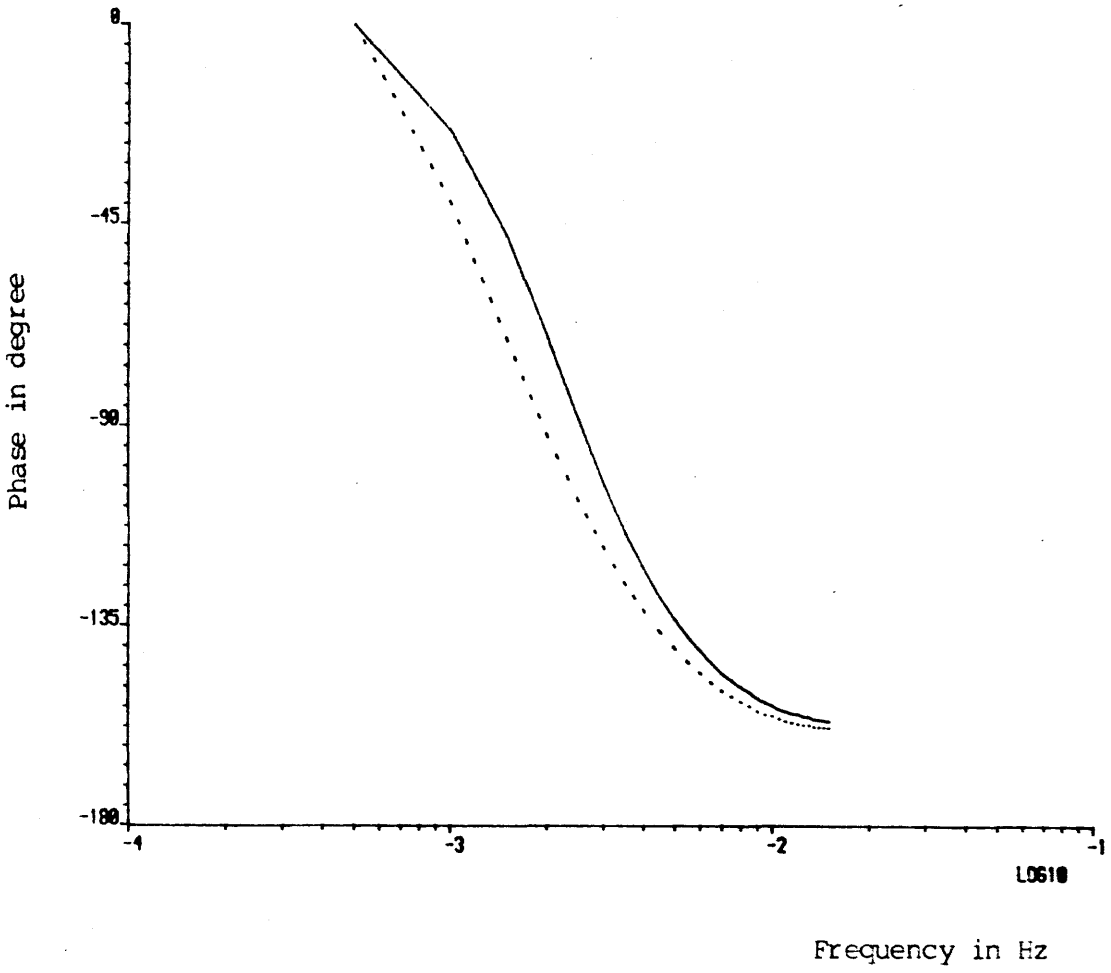


Figure 24 Phase spectra of the two tank sytem before and after the parameter R_2 is increased by 100%.

——— response before the parameter changes.
- - - - - response after the parameter has changed.

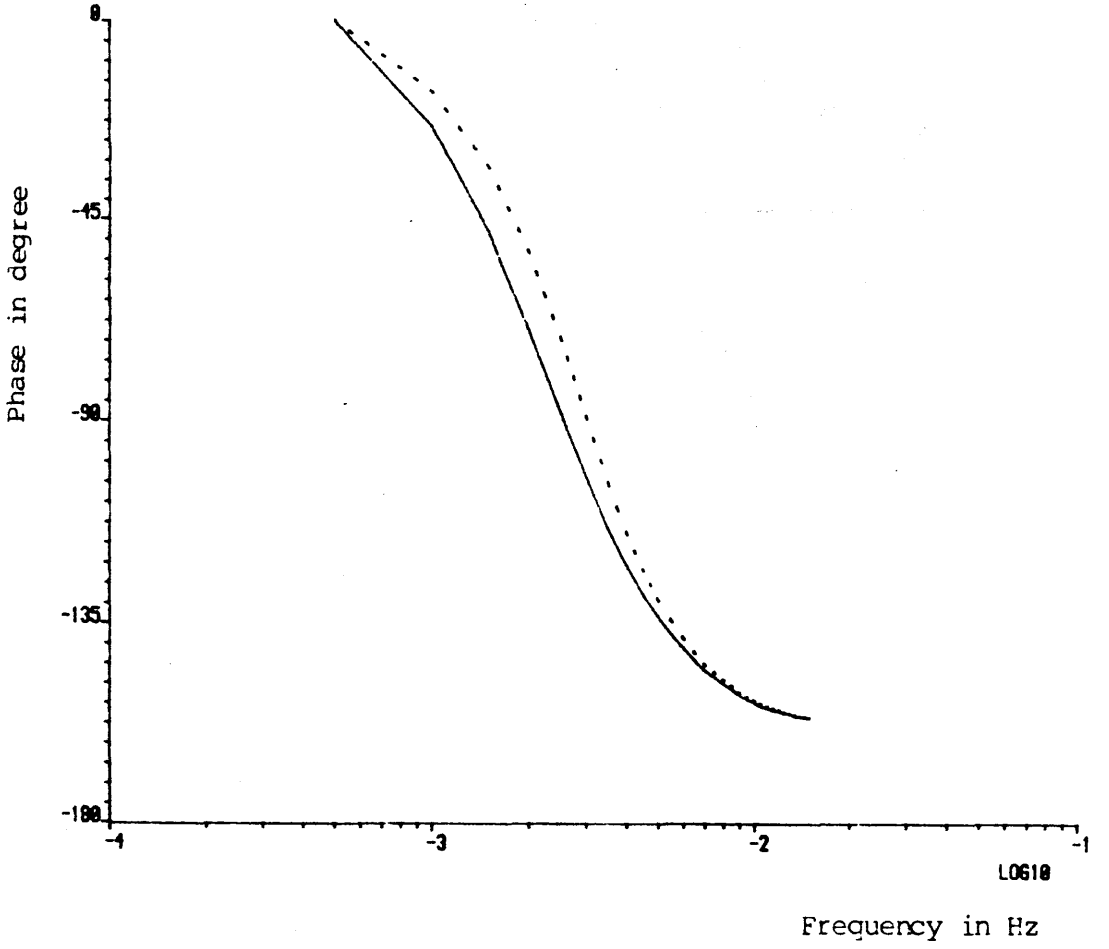


Figure 25 Phase spectra of the two tank system before and after the parameter G is increased by 100%.

——— response before the parameter changes.
- - - - - response after the parameter has changed.

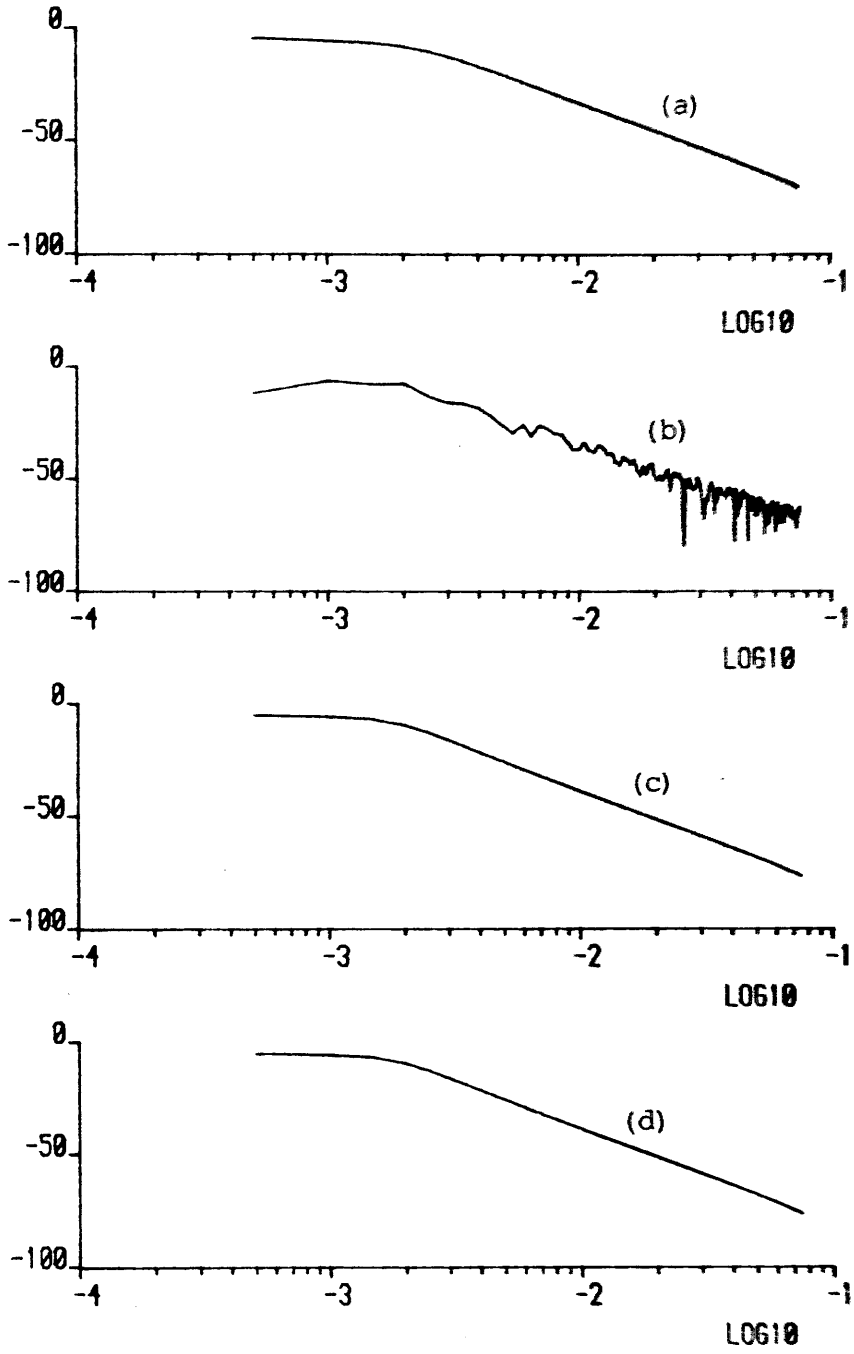


Figure 26 Magnitude spectra of the two tank system.
The parameter R_1 is increased by 100%.

(a) the spectrum before the parameter changes.
(b) the spectrum when the change takes place.
(c) the spectrum 1000 seconds after the change took place.
(d) the spectrum 3000 seconds after the change took place.

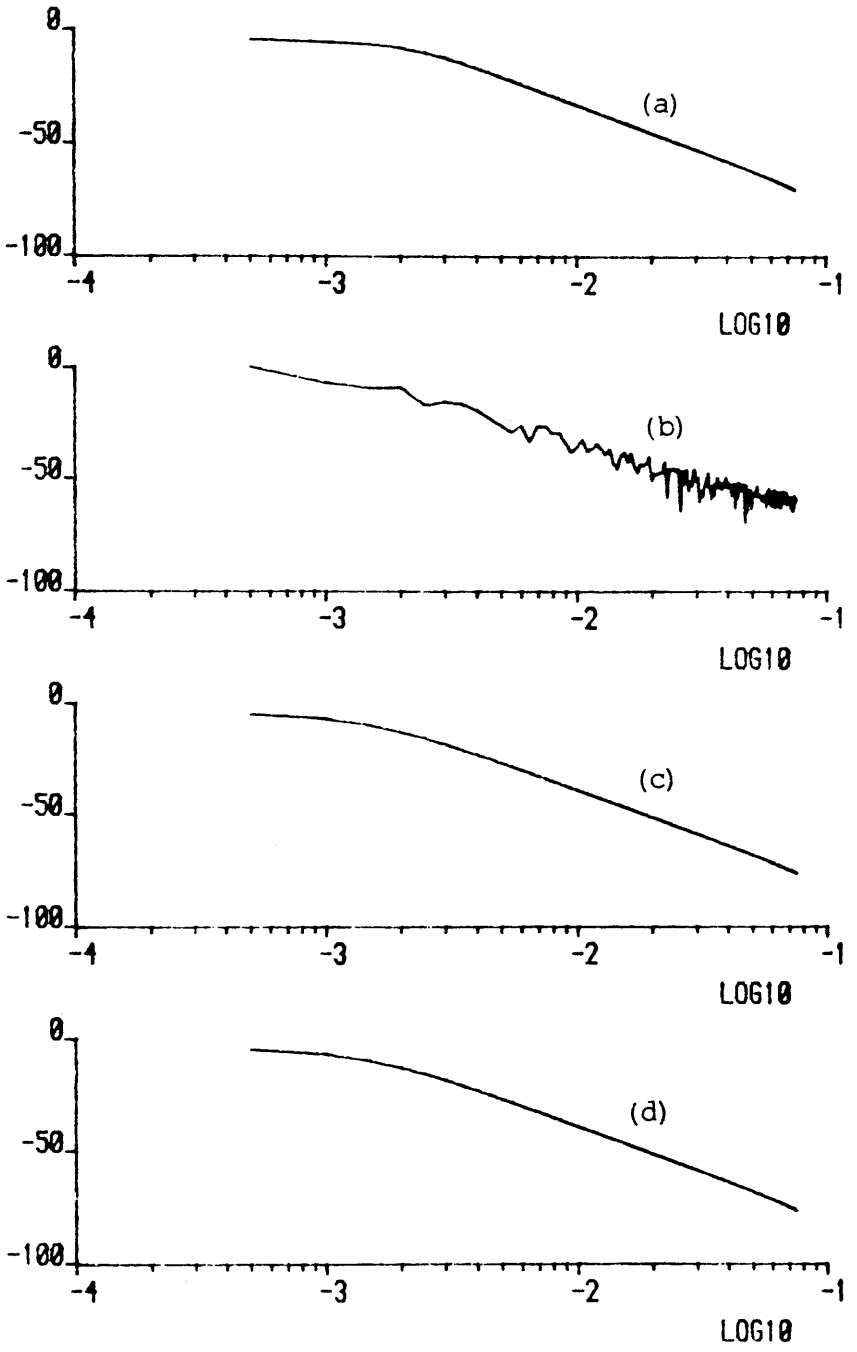


Figure 27 Magnitude spectra of the two tank system.
The parameter R_2 is increased by 100%.

(a) the spectrum before the parameter changes.
(b) the spectrum when the change takes place.
(c) the spectrum 1000 seconds after the change took place.
(d) the spectrum 3000 seconds after the change took place.

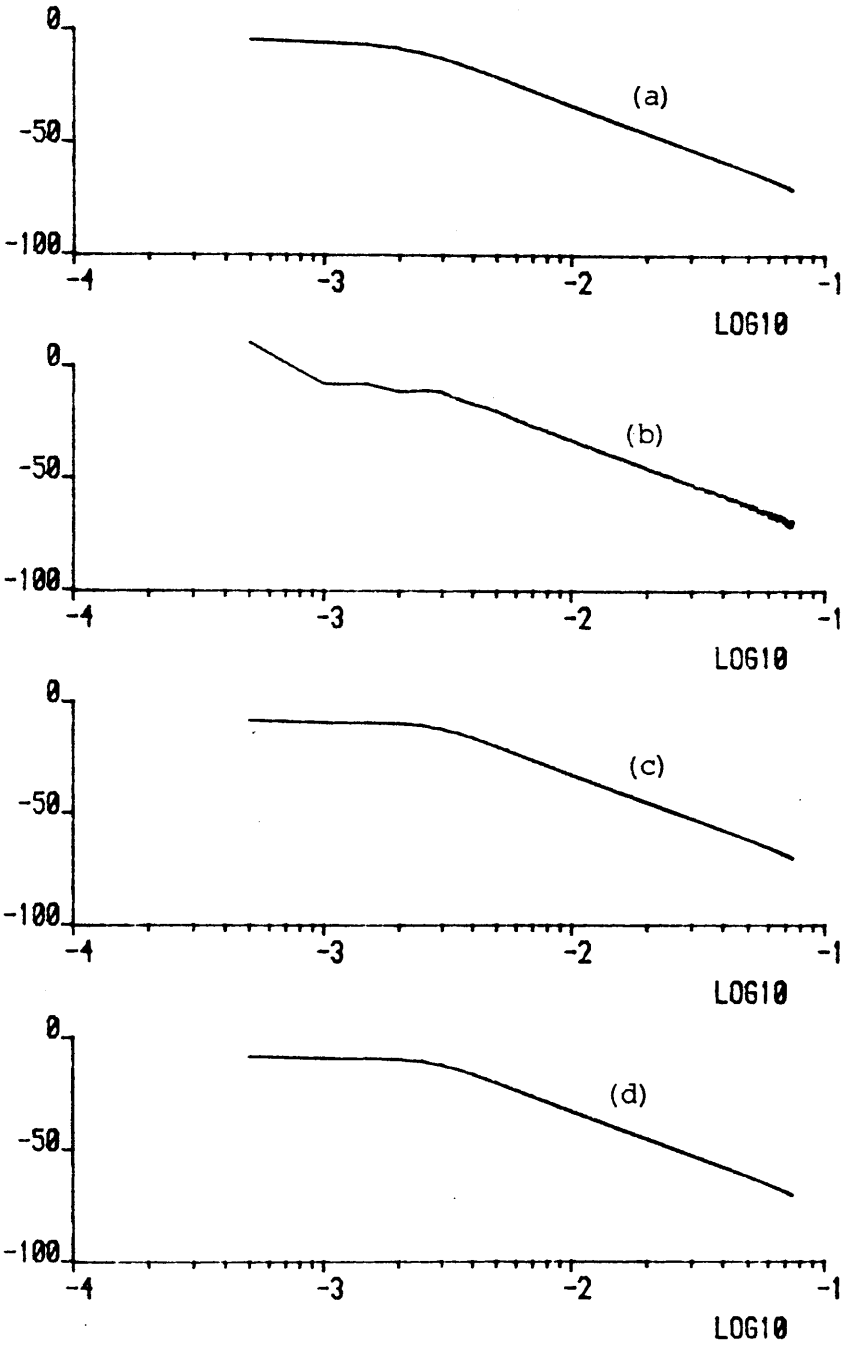


Figure 28 Magnitude spectra of the two tank system.
The parameter G is increased by 100%.

(a) the spectrum before the parameter changes.
(b) the spectrum when the change takes place.
(c) the spectrum 1000 seconds after the change took place.
(d) the spectrum 3000 seconds after the change took place.

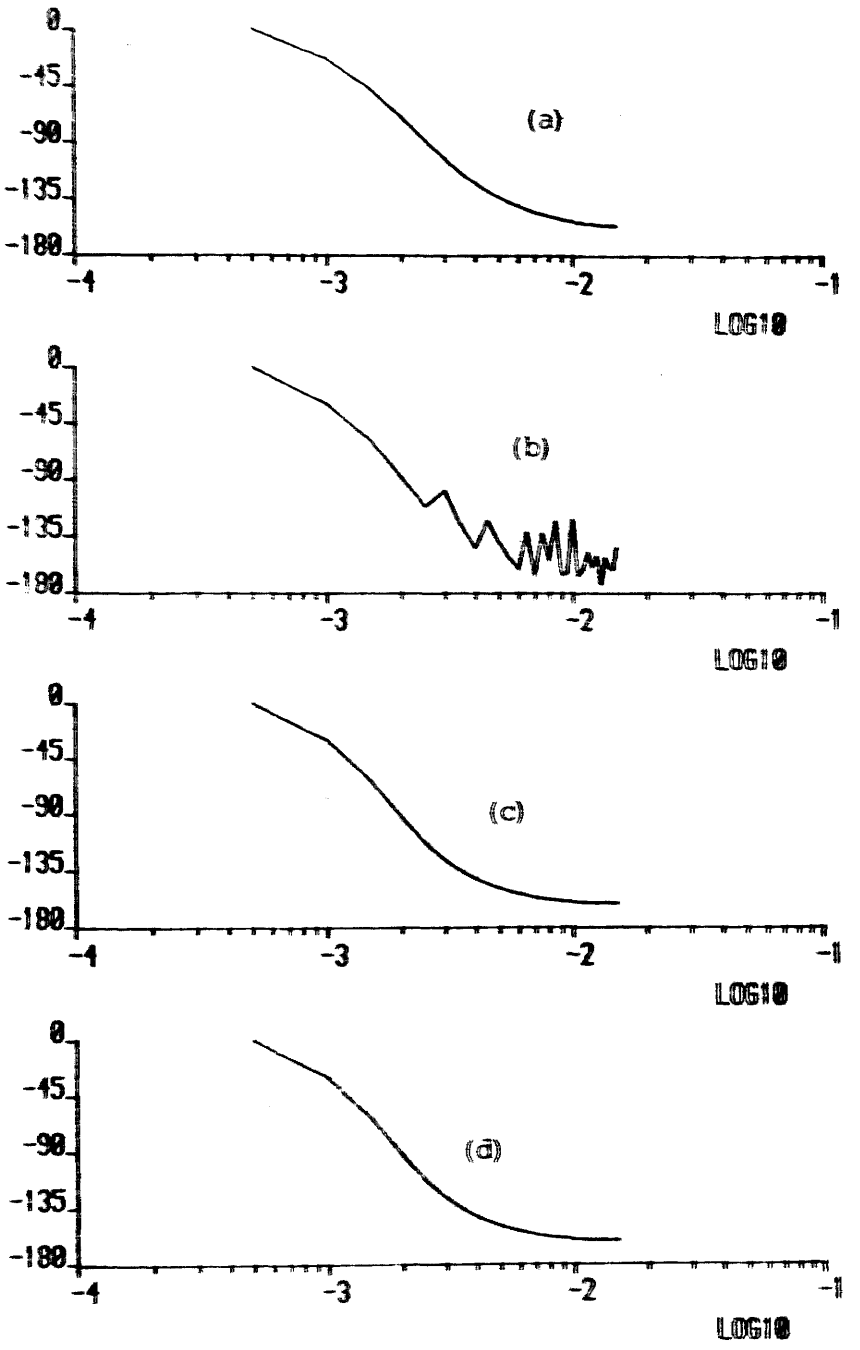


Figure 29 Phase spectra of the two tank system.
The parameter P_1 is increased by 100%.

(a) the spectrum before the parameter changes.
(b) the spectrum when the change takes place.
(c) the spectrum 1000 seconds after the change took place.
(d) the spectrum 3000 seconds after the change took place.

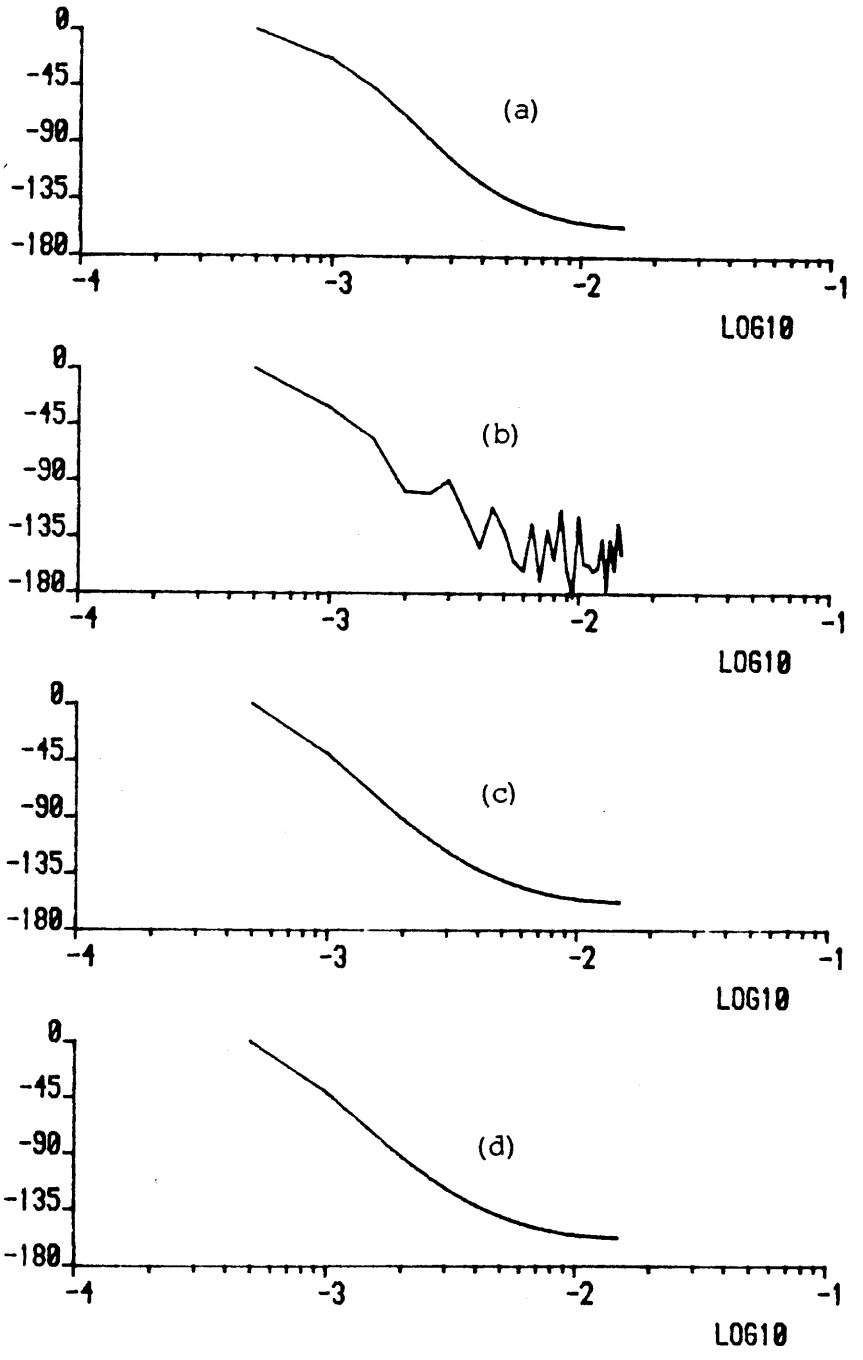


Figure 30 Phase spectra of the two tank system.
The parameter R_2 is increased by 100%.

(a) the spectrum before the parameter changes.
(b) the spectrum when the change takes place.
(c) the spectrum 1000 seconds after the change took place.
(d) the spectrum 3000 seconds after the change took place.

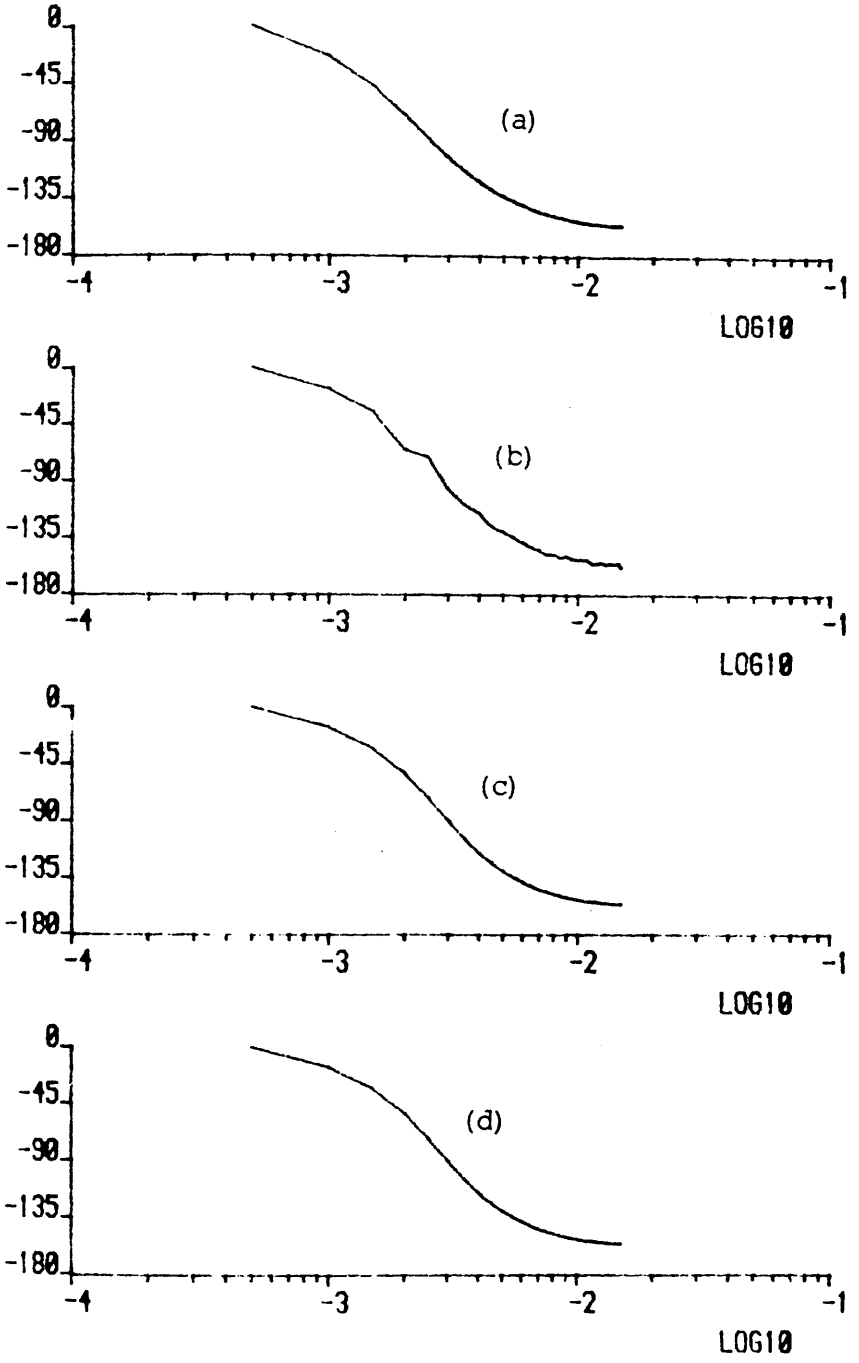


Figure 31 Phase spectra of the two tank system.
The parameter G is increased by 100%.

(a) the spectrum before the parameter changes.
(b) the spectrum when the change takes place.
(c) the spectrum 1000 seconds after the change took place.
(d) the spectrum 3000 seconde after the change took place.

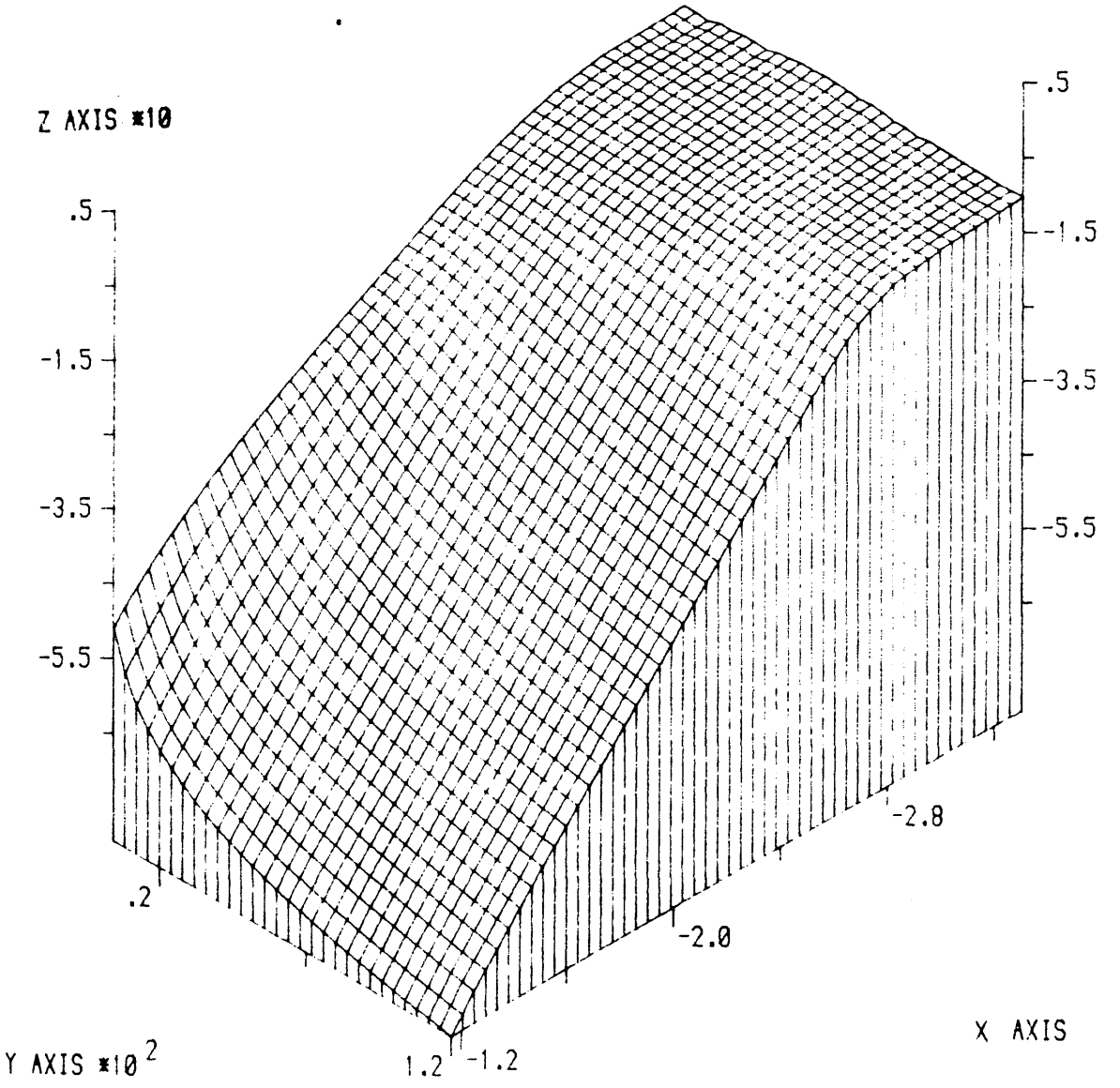


Figure 32 Three dimensional plotting of the frequency response of the two tank system.
X-axis represents the frequency (Hz) on a log scale.
Y-axis represents the value of the parameter R_1 which varies from 4.0 to 120.0.
Z-axis represents the magnitude of the response in decibels.

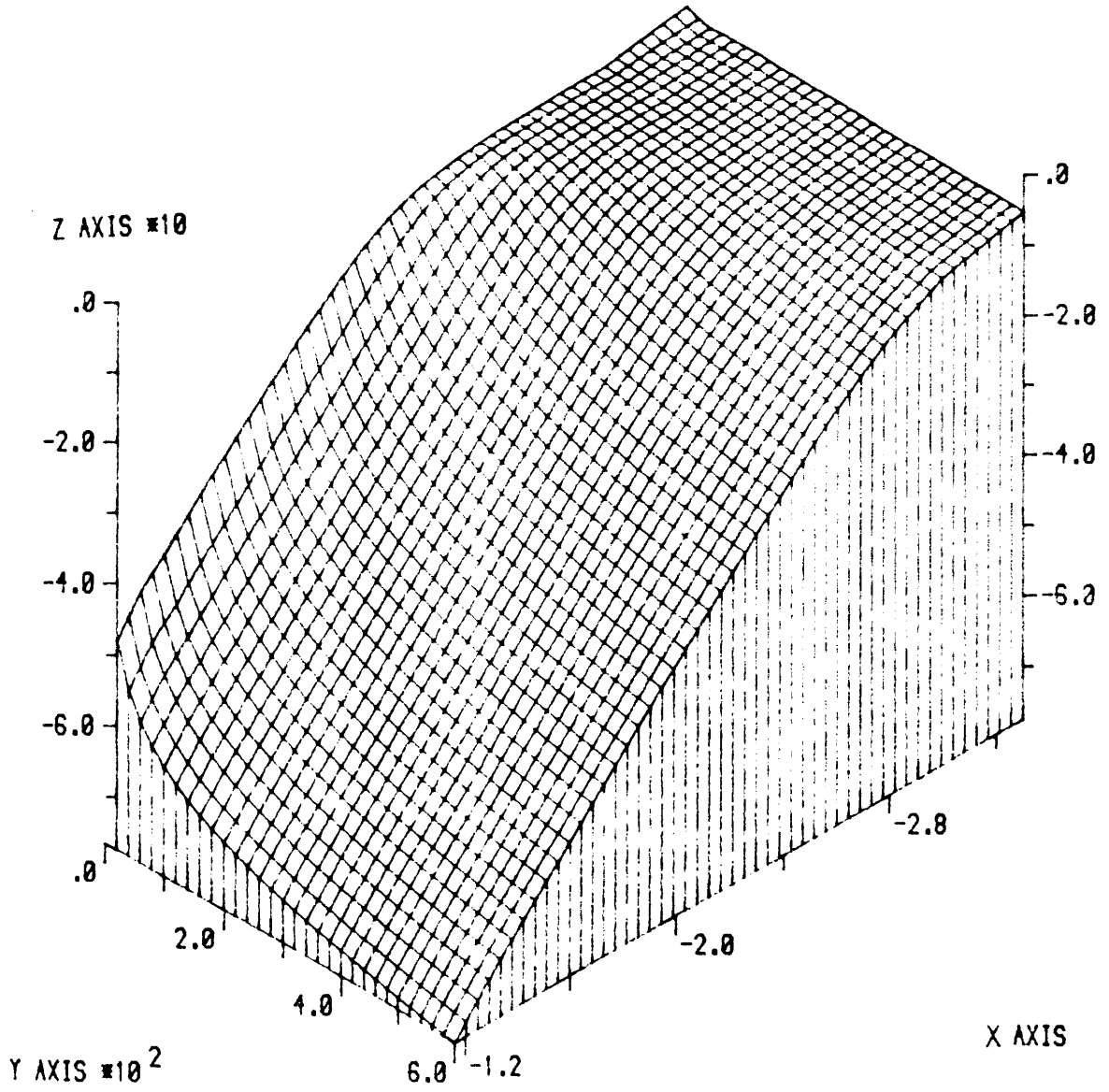


Figure 33 Three dimensional plotting of the frequency response of the two tank system.
X-axis represents the frequency (Hz) on a log scale.
Y-axis represents the value of the parameter R_2 which varies from 20.0 to 600.0.
Z-axis represents the magnitude of the response in decibels.

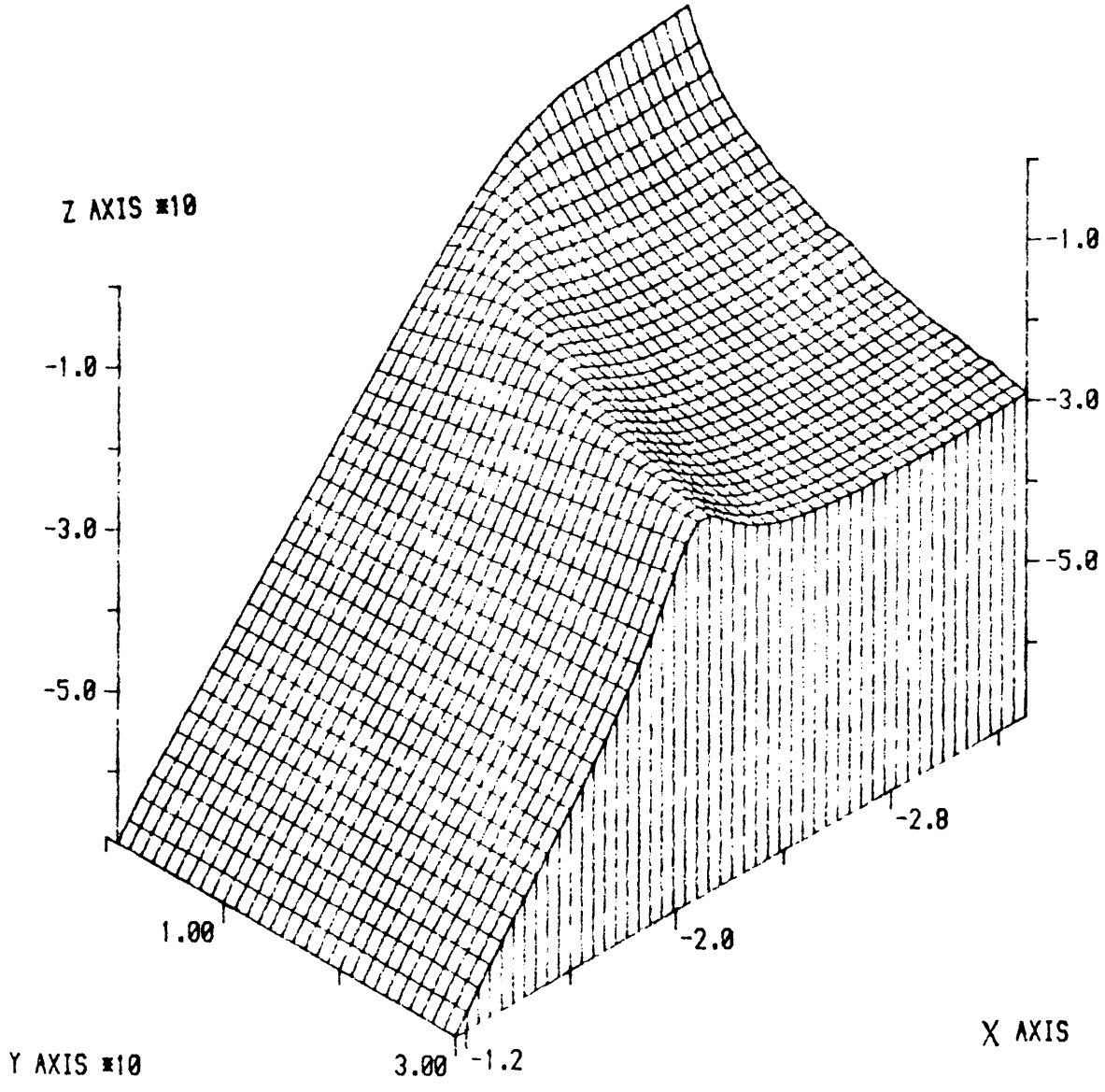


Figure 34 Three dimensional plotting of the frequency response of the two tank system.
X-axis represents the frequency (Hz) on a log scale.
Y-axis represents the value of the parameter G which varies from 1.0 to 30.0.
Z-axis represents the magnitude of the response in decibels.

As shown in figures 26, 27, 28, 29, 30 and 31 the spectra in figures 26b, 27b, 28b, 29b, 30b and 31b are far less smooth than the rest of the spectra. The (b) figures are the Fourier transformation of a time series data in which a change of a parameter has taken place. The duration of the time series data in which the spiky spectra are obtained give an approximate indication of the time when the parameter changes occur. When comparing the frequency domain notation with the impulse response notation as shown in figures 17, 18 and 19, it can be seen that the impulse response does not show exactly when the parameter changes. This is because the impulse response has already reached a zero steady state when the parameter change takes place at t=300 sec. The effect of the parameter change does not show up until the next impulse response.

Both the impulse responses and the frequency responses show the effect of parameter change on the system characteristics but in different ways. The changes of the impulse response can be expressed in terms of peak value, number of overshoots, time constant etc. while the changes of the frequency response are expressed in terms of bandwidth, corner frequency, cut-off rate etc.

Comparing figures 20, 21 and 22 shows that the change of parameters R_1 and R_2 have similar effects on the system response. They both reduce the magnitude of the high frequency components by about 6dB while little change can be seen in the low frequency components. R_1 increases the rate of change of magnitude spectrum around the corner frequency more severely than R_2. The change of the parameter G has little effect on the high frequency components but it reduces the magnitude of the low frequency components. As the value of G increases, a resonant peak is formed. This forming of a resonant peak can be clearly seen in the three dimensional plotting as shown in figure 34.

From the phase diagrams as shown in figures 23, 24 and 25, the increase of either of the parameters, R_1 or R_2, has introduced a phase lag effect in the system phase response. The change of R_1 seems to have more effect on the high frequency components than that of R_2. The increase of parameter G introduces a phase lead on the system response.

As a result, it can be concluded that the change of each parameter has its unique effect on the system frequency response.

CHAPTER FOUR

4. ON-LINE METHOD FOR DETERMINATION OF PARAMETER SENSITIVITIES IN CLOSED LOOP SYSTEMS

4.1 Introduction to Sensitivity

A parameter sensitivity function of a closed-loop control system can be defined as the change of the system response with respect to the change of a system parameter. It shows the degree of influence of a parameter change upon the system response. A system sensitivity function can be generated by the parameter perturbation method or by using an appropriate filter. Both approaches will be shown later.

In the last chapter, a system fault was simulated by a change of a parameter. It was identified by observing the system frequency response before and after the change of the parameter occurs. In this chapter, observation will be made on the system sensitivity functions that can also be varied by a change of a system parameter. By identifying the effect of parameter changes on system sensitivity functions, the future system fault can be classified.

A parameter sensitivity function can be used to improve the system performance by making appropriate changes on the system parameters, thus allowing optimisation of the system response to be carried out. System optimisation can be used to maintain a normal system response when the system is faulty. The optimisation technique using system sensitivity functions will be shown in section 4.7.

4.2 Generation of Sensitivity Functions using Small Perturbation method

The sensitivity of the output signal Y(s) to an adjustable parameter (a_i say) is defined in the frequency domain[17] as :

$$S^{Y}_{a_i}(s) = \frac{\partial Y(s)}{\partial a_i}$$

$$= \lim_{\Delta a_i \to \phi} \frac{Y(s, a_{i0} + \Delta a_i) - Y(s, a_{i0})}{\Delta a_i} \qquad (4.1)$$

where a_{i0} is the initial change of the parameter a_i and Δa_i is the small change of the parameter.

As shown in equation 4.1, the sensitivity function can be expressed as the normalised difference between the output responses before and after a small change of a parameter. The equation is only valid when the change of parameter, i.e. a_i is very small, due to the linearization involved in the above definition. In the following, a number of sensitivity functions will be derived based upon the equation 4.1.

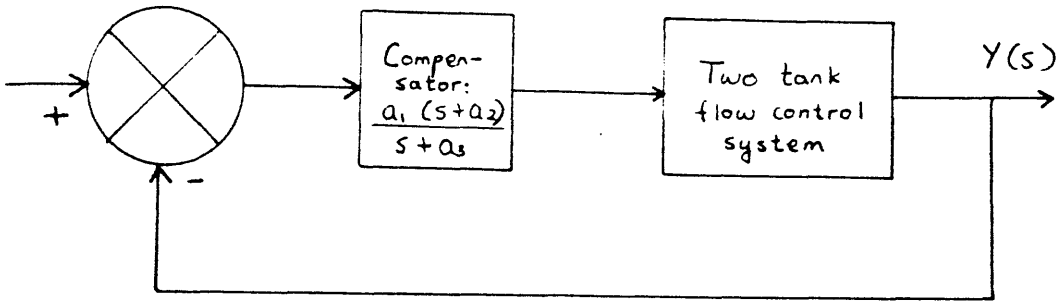


Figure 35 Block Diagram of the Two Tanks Liquid Flow Control System controlled by Cascade Compensator

Figure 35 shows the block diagram of the two-tank flow control system which is cascaded by a compensator of the form:

$$C(s) = \frac{a_1 (s + a_2)}{s + a_3} \qquad (4.2)$$

where $a_1 = 2.5$, $a_2 = 0.01$, $a_3 = 0.025$.

This compensator is intended to increase the bandwidth of the frequency response of the two-tank flow control system, thus an improvement in the speed of the system response results. Impulse responses of the overall system with and without the insertion of the compensator are shown in figures 36. The frequency spectra of the system responses are shown in figure 37.

Sensitivity functions are generated by injecting an impulse function at the system input. A small change is then made on the parameter a_i of the compensator. The normalised difference of the impulse responses before and after the parameter change gives the system sensitivity function with respect to the parameter a_i. By the varying each of the parameters : a_1, a_2, a_3 of the compensator, the sentitivity functions S^Y_{a1}, S^Y_{a2}, S^Y_{a3} are derived in turn. The magnitude spectra of the sensitivity functions are plotted as shown in figure 38- 40.

In the particular example shown above, since an impulse response of a system is the time domain equivalent of the system transfer function, the sensitivity functions of the system output signal are equal to the those with respect to the overall system transfer function. Therefore S^{y}_{a1}, S^{y}_{a2}, S^{y}_{a3} are equal to S^{Wc}_{a1}, S^{Wc}_{a2}, S^{Wc}_{a3} respectively.

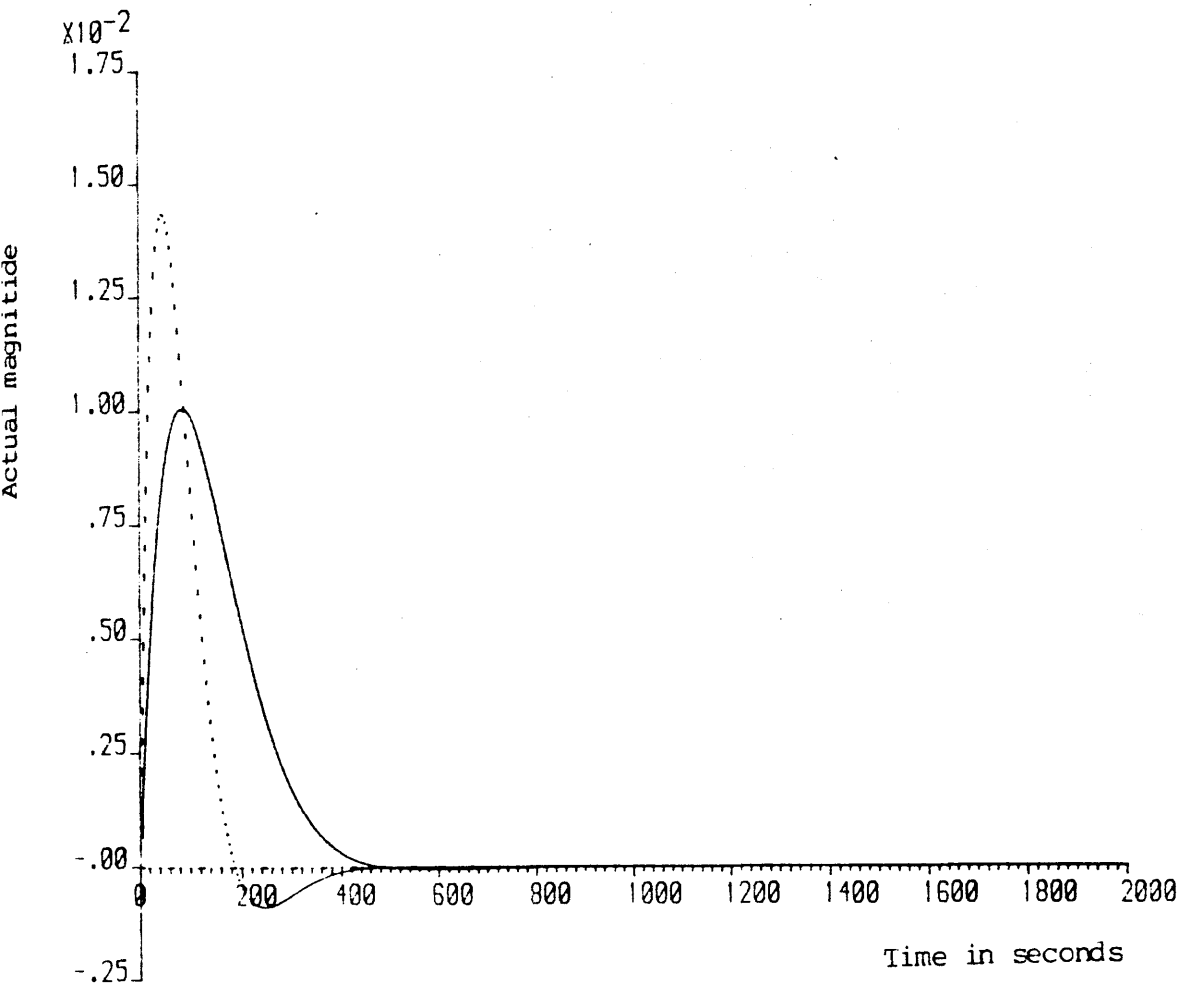


Figure 36 Impulse Responses of the Two Tank System with and without Compensation

——— impulse response of the two tank system.
---- impulse response of the two tank system with compensation.

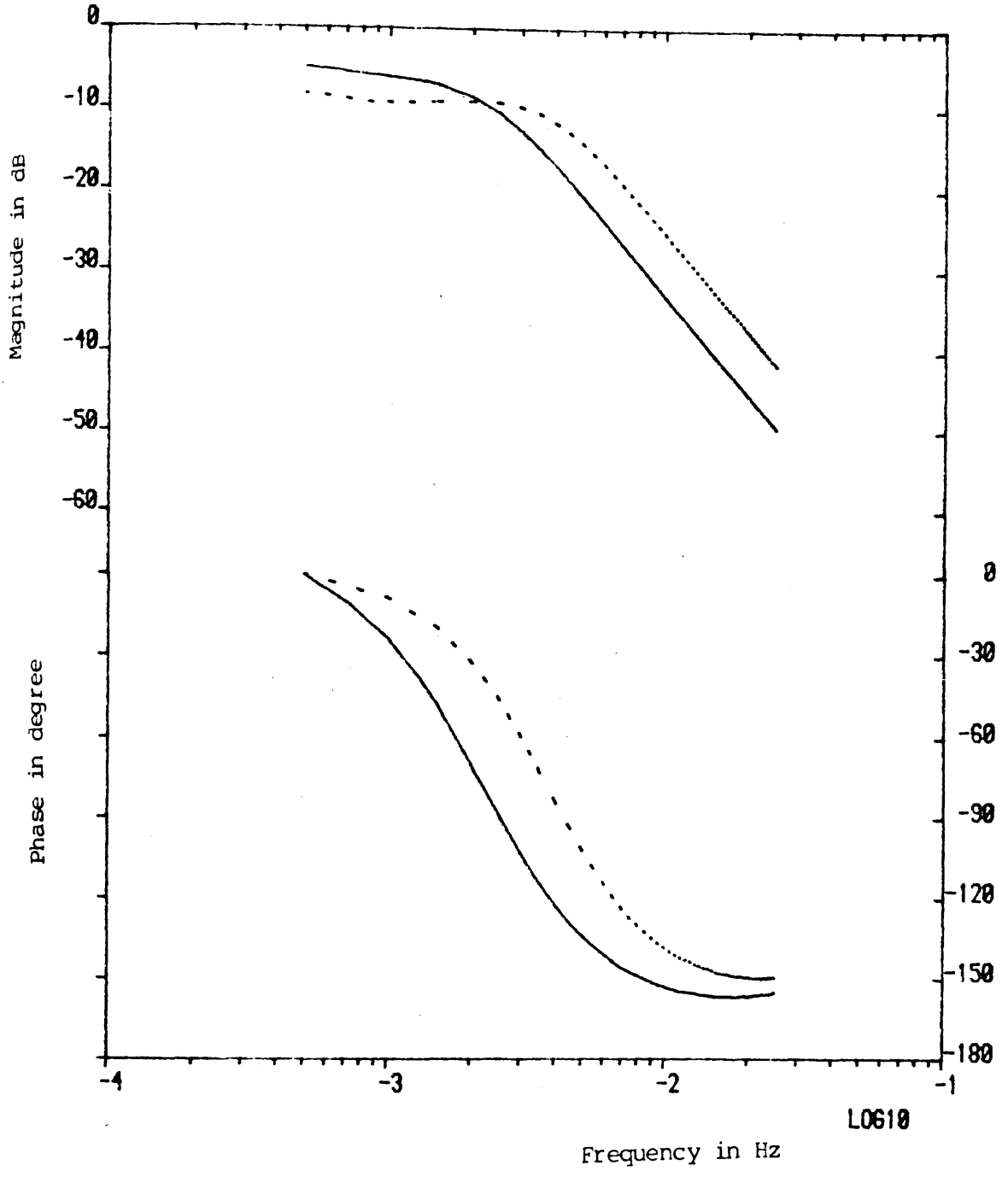


Figure 37 Frequency Spectra of the Two Tank System Response with and without Compensation

——— response of the two tank system.

- - - - response of the two tank system with compensation.

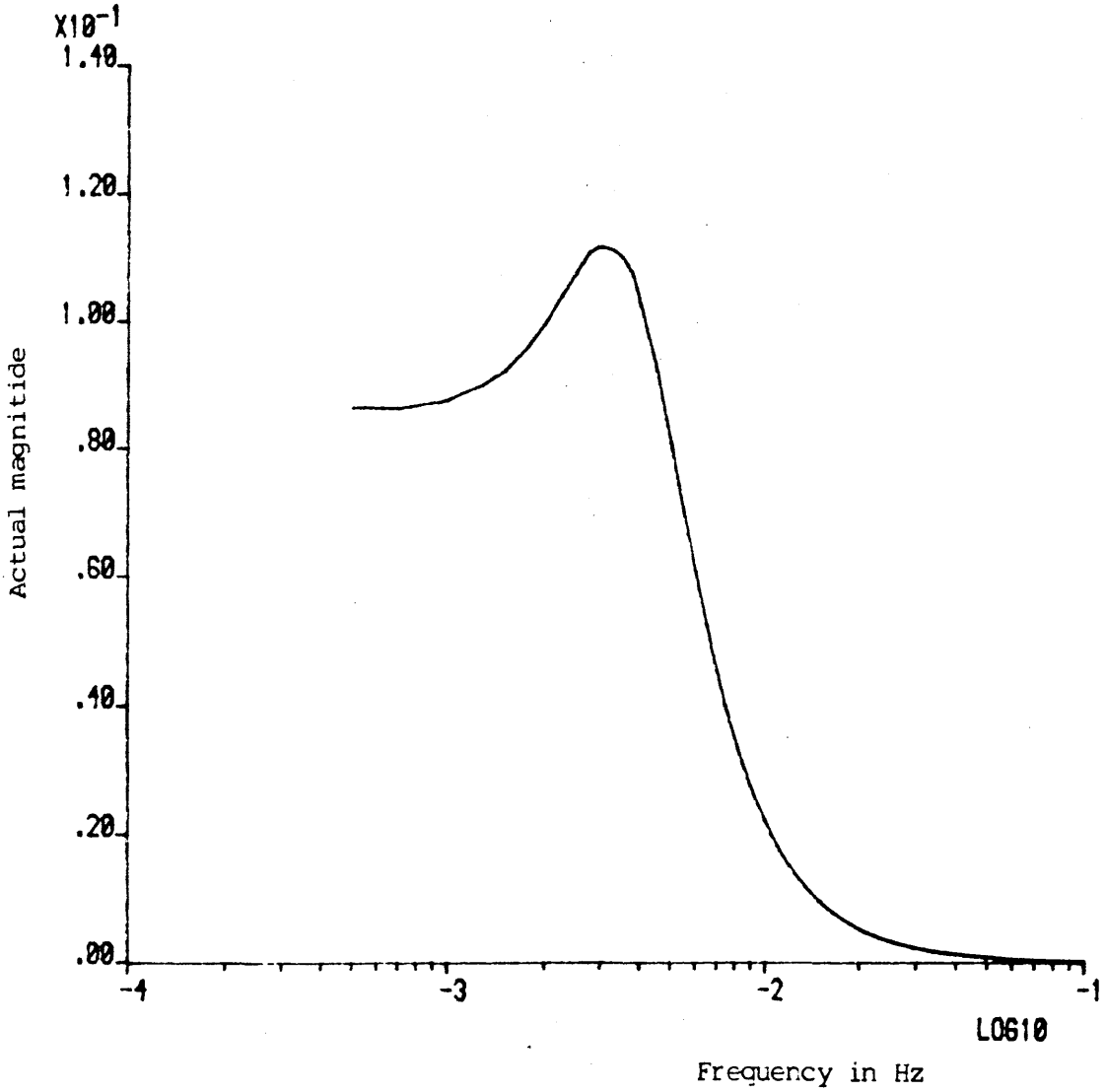


Figure 38 Plot of the Sensitivity Function $\left|S^{y}_{a_1}\right|$ vs Frequency

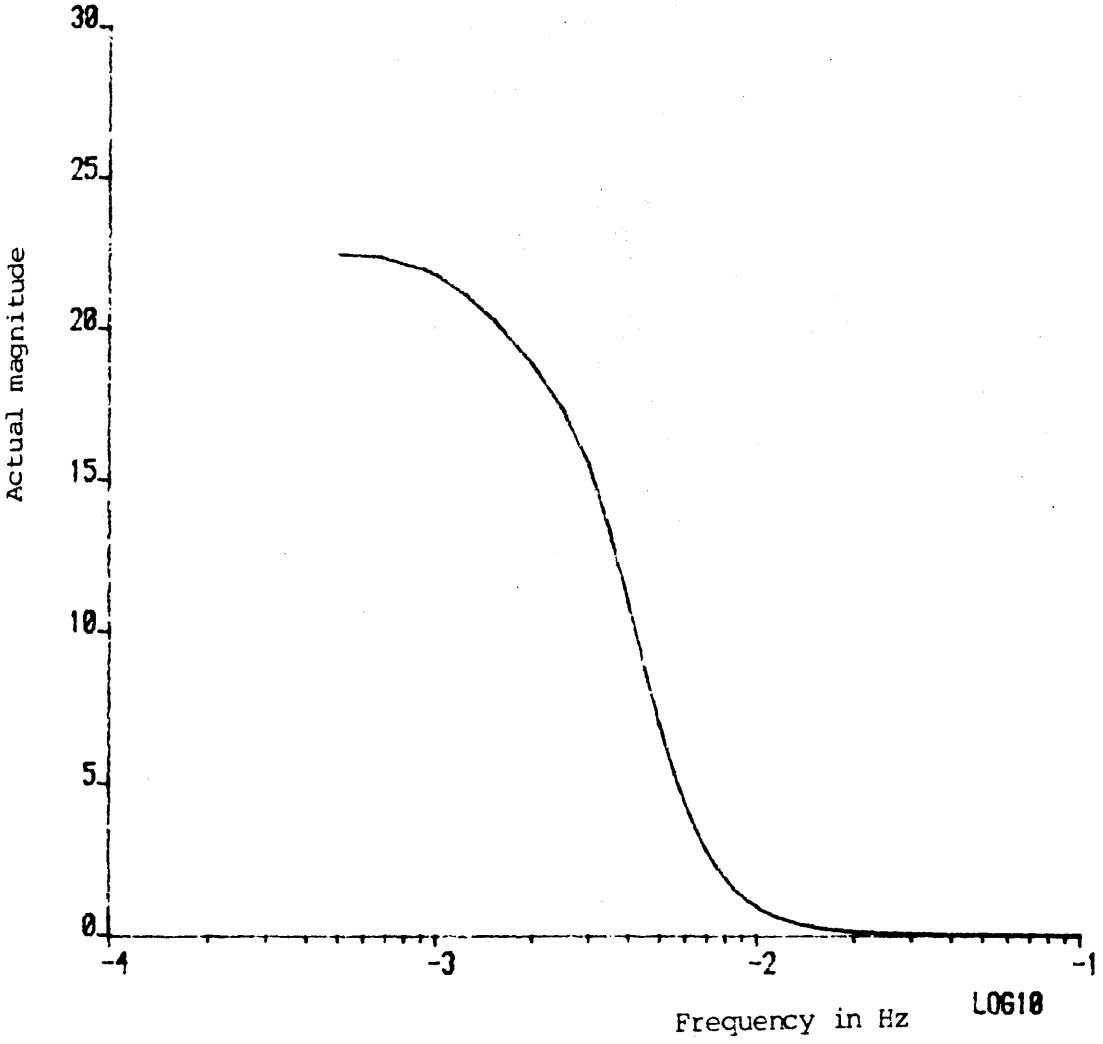


Figure 39 Plot of the Sensitivity Function $\left|S^{y}_{a_2}\right|$ vs Frequency

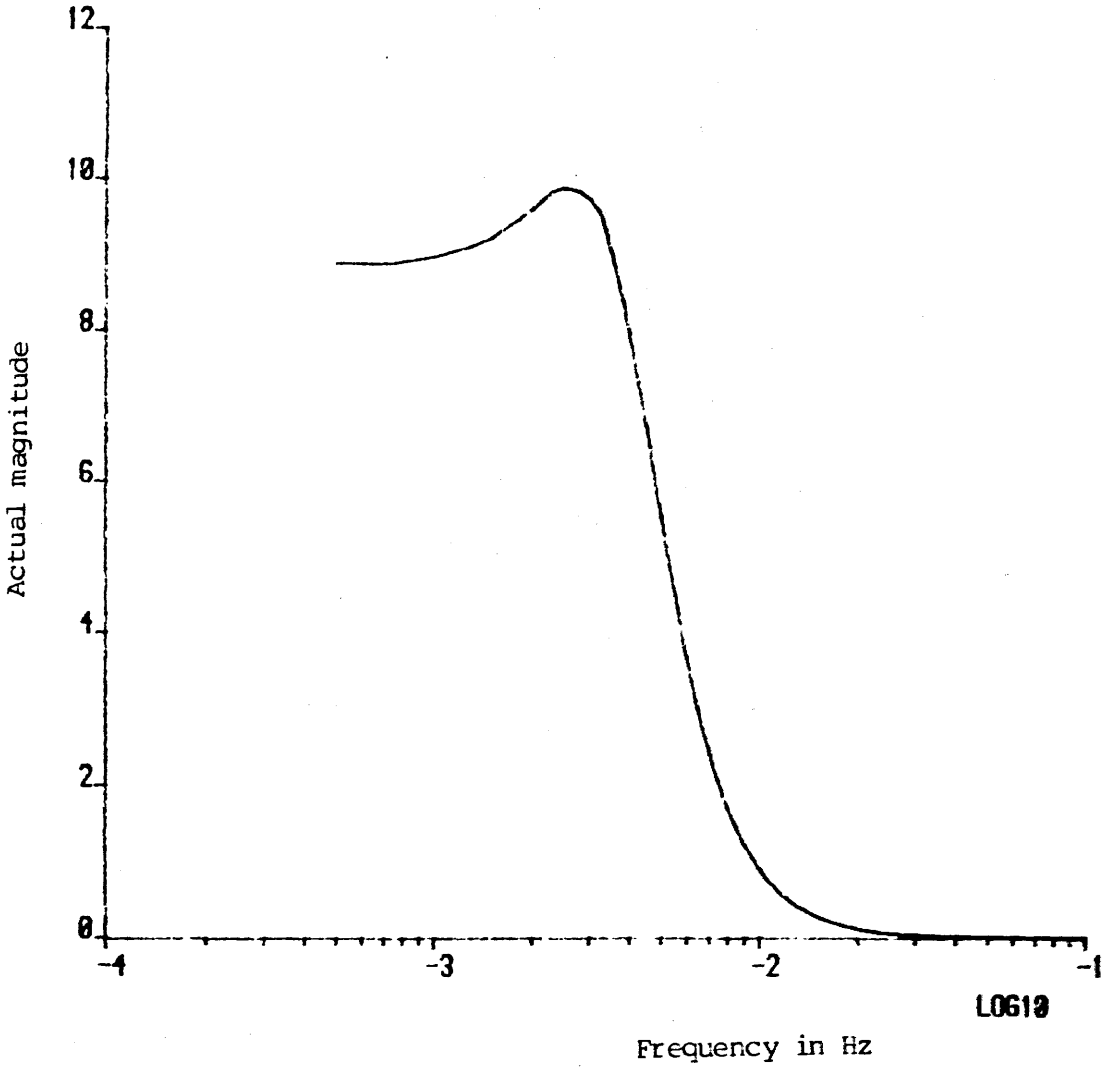


Figure 40 Plot of the Sensitivity Function $\left|S^{y}_{a_3}\right|$ vs Frequency

4.3 Generation of Sensitivity Functions using Filters

The diagram below shows a closed loop system :

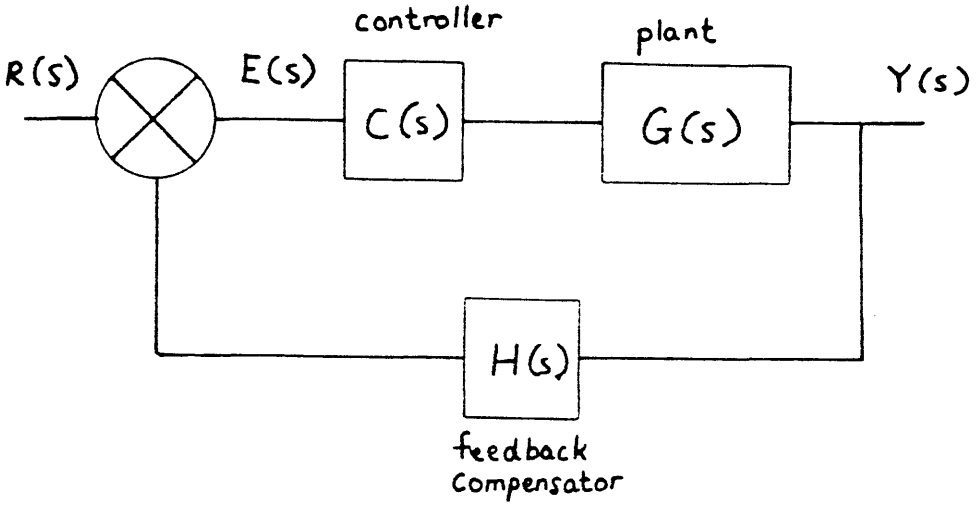


Figure 41 Block Diagram of a Closed Loop system

where C(s) is the controller which has m adjustable parameters $a_1, a_2, \ldots, a_m$. H(s) is the feedback compensator which contains n adjustable parameters $b_1, b_2, \ldots, b_n$. G(s) is the transfer function of the plant under monitored. R(s) is the input signal. E(s) is the actuating signal and Y(s) is the output signal.

The output Y(s) of the system can written as :

$$Y(s) = \frac{C(s)\,G(s)}{1 + C(s)\,G(s)\,H(s)}\,R(s) \qquad (4.3)$$

The sensitivity function of the system output Y(s) with respect to the parameter a_i can be expressed as follows :

$$\frac{\partial Y(s)}{\partial a_i} = \frac{\frac{\partial C(s)}{\partial a_i}\,G(s)}{[1 + C(s)H(s)G(s)]^2}\,R(s) \qquad (4.4)$$

$$\frac{\partial Y(s)}{\partial a_i} = \frac{G(s)C(s)}{[1 + C(s)H(s)G(s)]} \frac{1}{C(s)} \frac{\partial C(s)}{\partial a_i} \frac{R(s)}{[1 + C(s)H(s)G(s)]} \tag{4.5}$$

$$= \frac{Y(s)}{R(s)} \frac{1}{C(s)} \frac{\partial C(s)}{\partial a_i} E(s) \tag{4.6}$$

Equation (4.6) can be rewritten as

$$= \frac{Y(s)}{R(s)} F_{a_i}(s) E(s) \tag{4.7}$$

where $F_{a_i}(s) = \frac{1}{C(s)} \frac{\partial C(s)}{\partial a_i}$ (4.8)

Similarly the sensitivity of the system output Y(s) with respect to b_j can be derived as follows :

$$\frac{\partial Y(s)}{\partial b_j} = \frac{-C(s)G(s)R(s)}{[1 + C(s)G(s)R(s)]^2} C(s)G(s) \frac{\partial H(s)}{\partial b_j} \tag{4.9}$$

$$= \frac{-Y(s)}{R(s)} \frac{\partial H(s)}{\partial b_j} Y(s) \tag{4.10}$$

Equation (4.10) can be written as

$$\frac{\partial Y(s)}{\partial b_j} = \frac{Y(s)}{R(s)} F_{b_j}(s) Y(s) \tag{4.11}$$

where $F_{b_j}(s) = \frac{-\partial H(s)}{\partial b_j}$ (4.12)

From equation (4.6) and (4.10), we can say that the sensitivity function $\frac{\partial Y(s)}{\partial a_i}$ and $\frac{\partial Y(s)}{\partial b_j}$ are both dependent on the input signal R(s). In the other words, by injecting different input signals into the system, different sensitivity functions can be obtained.

As seen from equation (4.7) and (4.11) that the generation of the sensitivity functions $\frac{\partial Y(s)}{\partial a_i}$ and $\frac{\partial Y(s)}{\partial b_j}$ using the cosystem filters F_{a_i}, F_{b_j} does not require any knowledge of the plant G(s) under controlled. However precise knowledge of the input and output signals, the feedback controller and the compensation controller is needed.

4.4 Generation of the Sensitivity Functions $\frac{\partial Y(s)}{\partial a_i}$, $\frac{\partial Y(s)}{\partial b_j}$, from the Time Domain Analysis

It can be seen from the previous section that the sensitivity function of the system output signal Y(s) is dependent upon the input signal R(s). The expression of the sensitivity functions for various types of standard input signals are derived as shown below :

1. UNIT-IMPULSE INPUT

$$\text{Input} \quad r(t) = \delta(t)$$
$$R(s) = 1 \qquad (4.13)$$

Recalling equation (4.7) gives :

$$\frac{\partial Y(s)}{\partial a_i} = \frac{Y(s)}{R(s)} F_{a_i}(s) E(s) \qquad (4.14)$$

$$= Y(s) F_{a_i}(s) E(s) \qquad (4.15)$$

$$= Y(s) Z_{a_i}(s) \qquad (4.16)$$

where $$Z_{a_i}(s) = E(s) F_{a_i}(s) \qquad (4.17)$$

Applying convolution on equation (4.16), the sensitivity function in the time domain can be expressed as follows :

$$\frac{\partial Y(t)}{\partial a_i} = \int_0^T Y(t) Z_{a_i}(t - \tau) d\tau \qquad (4.18)$$

Similarly for $\frac{\partial Y(s)}{\partial b_j}$, recalling equation (4.11) gives :

$$\frac{\partial Y(s)}{\partial b_j} = \frac{Y(s)}{R(s)} F_{b_j}(s)\, Y(s) \qquad (4.19)$$

$$= Y(s)\, F_{b_j}(s)\, Y(s) \qquad (4.20)$$

$$= Y(s)\, Z_{b_j}(s) \qquad (4.21)$$

where $$Z_{b_j}(s) = F_{b_j}(s)\, Y(s) \qquad (4.22)$$

Applying convolution on equation (4.21) gives

$$\frac{\partial Y(t)}{\partial b_j} = \int_o^\tau Y(t)\, Z_{b_j}(t-\tau)\, d\tau \qquad (4.23)$$

2 UNIT-STEP INPUT

$$\text{Input} \quad r(t) = u(t)$$
$$R(s) = 1/s \qquad (4.24)$$

Using the same method as in unit-impulse input, equation (4.7) can be written as :

$$\frac{\partial Y(s)}{\partial a_i} = sY(s)\, Z_{a_i}(s) \qquad (4.25)$$

So $$\frac{\partial Y(t)}{\partial a_i} = \int_c^{\tau} \frac{d Y(t)}{d t} Z_{a_i}(t - \tau)\, d\tau \qquad (4.26)$$

Similarly, equation (4.11) can be written as :

$$\frac{\partial Y(s)}{\partial b_j} = sY(s)\, Z_{b_j}(s) \qquad (4.27)$$

$$\frac{\partial Y(t)}{\partial b_j} = \int_o^{\tau} \frac{d Y(t)}{d t} Z_{b_j}(t - \tau)\, d\tau \qquad (4.28)$$

3 UNIT-RAMP (Velocity) INPUT

Input $r(t) = t$ or $\dot{r}(t) = 1$

$$R(s) = 1/s^2 \qquad (4.29)$$

Using similar method as above gives :

$$\frac{\partial Y(s)}{\partial a_i} = s^2\, Y(s)\, Z_{a_i}(s) \qquad (4.30)$$

therefore $$\frac{\partial Y(t)}{\partial a_i} = \int_o^{\tau} \frac{d^2 Y(t)}{d t^2} Z_{a_i}(t - \tau)\, d\tau \qquad (4.31)$$

Substituting equations (4.29) and (4.22) into (4.11) gives,

$$\frac{\partial Y(s)}{\partial b_j} = s^2 Y(s)\, Z_{b_j}(s) \qquad (4.32)$$

therefore $$\frac{\partial Y(t)}{\partial b_j} = \int_o^T \frac{d^2 Y(t)}{dt^2} Z_{b_j}(t-\tau)d\tau \qquad (4.33)$$

4 UNIT-PARABOLIC (Acceleration) INPUT

$$\text{Input } r(t) = t^2/2 \quad \text{or} \quad \ddot{r}(t) = 1$$

$$R(s) = 1/s^3 \qquad (4.34)$$

Using the same method as above gives

$$\frac{\partial Y(s)}{\partial a_i} = s^3 Y(s) Z_{a_i}(s) \qquad (4.35)$$

thus $$\frac{\partial Y(t)}{\partial a_i} = \int_o^T \frac{d^3 Y(t)}{d(t)^3} Z_{a_i}(t-\tau)d\tau \qquad (4.36)$$

$$\frac{\partial Y(s)}{\partial b_i} = s^3 Y(s) Z_{b_i}(s) \qquad (4.37)$$

Similarly, $$\frac{\partial Y(t)}{\partial b_j} = \int_o^T \frac{d^3 Y(t)}{d(t)^3} Z_{b_i}(t-\tau)d\tau \qquad (4.38)$$

It can be seen from equations(4.18),(4.26),(4.31) and (4.36) that $\frac{\partial Y(t)}{\partial a_i}$ can be evaluated from the signal $Z_{a_i}(t)$ and the output signal Y(t) and its derivatives. $Z_{a_i}(t)$ can be easily obtained by passing the actuating signal E(s) through a filter of $F_{a_i}(s)$. Similarly, $\frac{\partial Y(t)}{\partial b_j}$ can be derived from the signal $Z_{b_j}(t)$ and the output signal Y(t) or its derivatives. $Z_{b_j}(t)$ can be obtained by passing the actuating signal E(s) through a filter $F_{b_j}(s)$. It can be seen from equations (4.8) and (4.12) that both filters $F_{a_i}(s)$ and $F_{b_j}(s)$ only involve either the transfer function C(s) or H(s) but not the transfer function G(s). Therefore C(s) and H(s) are required to have a well defined form with all the adjustable parameter $a_1, a_2, \ldots, a_m$ and $b_1, b_2, \ldots, b_n$ properly specified, while the transfer function G(s) of the plant under controlled can remains totally unknown or non-specified.

4.5 Sensitivity Functions of the Closed-loop Transfer Function with respect to a Change of Parameter

Recalling from equation (4.3) gives the expression of the output related to the input signal of a closed-loop system gives :

$$Y(s) = \frac{C(s)G(s)}{1 + C(s)H(s)G(s)} R(s) \tag{4.39}$$

Therefore the transfer function of the closed-loop system can be written as :

$$W_c(s) = \frac{Y(s)}{R(s)} = \frac{C(s)G(s)}{1 + C(s)H(s)G(s)} \tag{4.40}$$

The sensitivity of the system transfer function $W_c(s)$ with respect to a_i can be expressed as :

$$\frac{\partial W_c(s)}{\partial a_i} = \frac{C(s)G(s)}{1 + C(s)H(s)G(s)} \frac{1}{C(s)} \frac{\partial C(s)}{\partial a_i} \frac{1}{1 + C(s)H(s)} \tag{4.41}$$

$$= W_c(s) \frac{1}{C(s)} \frac{\partial C(s)}{\partial a_i} \frac{E(s)}{R(s)} \tag{4.42}$$

Equation (4.42) can be written as :

$$\frac{\partial W_c(s)}{\partial a_i} = W_c(s)\, P(s) \tag{4.43}$$

where $$P(s) = \frac{1}{C(s)} \frac{\partial C(s)}{\partial a_i} \frac{E(s)}{R(s)} \tag{4.44}$$

Similarly, the sensitivity function of the system output with respect to b_j can be expressed as follows:

$$\frac{\partial W_c(s)}{\partial b_j} = \frac{-C(s)G(s)}{1 + C(s)H(s)G(s)} \frac{\partial H(s)}{\partial b_j} \frac{C(s)G(s)}{1 + C(s)H(s)G(s)} \tag{4.45}$$

$$= -\frac{Y(s)}{R(s)} \frac{\partial H(s)}{\partial b_j} W_c(s) \tag{4.46}$$

Equation (4.46) can be written as :

$$\frac{\partial W_c(s)}{\partial b_j} = W_c(s)\, Q(s) \tag{4.47}$$

where $$Q(s) = -\frac{\partial H(s)}{\partial b_j} \frac{Y(s)}{R(s)} \tag{4.48}$$

It should be pointed out that the sensitivity functions $\frac{\partial W_c(s)}{\partial a_i}$, $\frac{\partial W_c(s)}{\partial b_j}$ shown in equations (4.41) and (4.45) are independent of the input signal R(s) while the sensitivity functions $\frac{\partial Y(s)}{\partial a_i}$ and $\frac{\partial Y(s)}{\partial b_j}$ as shown in equations (4.7) and (4.11) are dependent on R(s).

4.6 Generation of the Sensitivity Functions $\frac{\partial W_c(s)}{\partial a_i}$ and $\frac{\partial W_c(s)}{\partial b_j}$ from the Frequency Domain

To generate $\frac{\partial W_c(s)}{\partial a_i}$, equations (4.42), (4.43) and (4.44) are recalled as follows:

$$\frac{\partial W_c(s)}{\partial a_i} = W_c(s) \frac{1}{C(s)} \frac{\partial C(s)}{\partial a_i} \frac{E(s)}{R(s)} \tag{4.49}$$

$$= W_c(s)\, P(s) \tag{4.50}$$

where $$P(s) = \frac{1}{C(s)} \frac{\partial C(s)}{\partial a_i} \frac{E(s)}{P(s)} \tag{4.51}$$

The functions $W_c(s)$ and $P(s)$ can be expressed in polar form as shown below :

$$W_c(jw) = M(w)\, e^{j\alpha(w)} \tag{4.52}$$

and $$P(jw) = U(w)\, e^{j\beta(w)} \tag{4.53}$$

where $M(w) = |W_c(jw)|$ and $\alpha(w) = \text{Arg}[W_c(jw)]$
and $U(w) = |P(jw)|$ and $\beta(w) = \text{Arg}[P(jw)]$

Substituting (4.51) and (4.52) into (4.49) gives :

$$\frac{\partial W_c(jw)}{\partial a_i} = M(w)\, U(w)\, e^{j[\alpha(w) + \beta(w)]} \tag{4.54}$$

By using approximation,

$$W_c(jw, a + \Delta a_i) = W_c(jw, a) + \frac{\partial W_c(jw, a)}{\partial a_i} \Delta a_i \tag{4.55}$$

$$W_c(jw,\ a + \Delta a_i) = M(w)e^{j\alpha(w)} + U(w)e^{j\beta(w)}\Delta a_i \tag{4.56}$$

It may be shown that :

$$W_c(jw,\ a+\Delta a_i) \simeq [\ M(w)^2 + 2M(w).U(w).\cos(\alpha-\beta).\Delta a_i\ ]^{1/2} \tag{4.57}$$

$$= M(w)\ [\ 1 + U(w).\cos\beta\ .\ \Delta a_i\ ] \tag{4.58}$$

Therefore $$\frac{\Delta M(w)}{\Delta a_i} \simeq M(w)\ U(w)\ \cos\beta \tag{4.59}$$

or $$\frac{\Delta |W_c(jw)|}{\Delta a_i} \simeq M(w)\ U(w)\ \cos\beta \tag{4.60}$$

Next, the function $\frac{\partial W_c(s)}{\partial b_j}$ is generated by recalling equations (4.46), (4.47) and (4.48).

$$\frac{\partial W_c(w)}{\partial b_j} = -\frac{Y(s)}{R(s)}\frac{\partial H(s)}{\partial b_j}\ W_c(s) \tag{4.61}$$

$$= W_c(s)\ Q(s) \tag{4.62}$$

where $$Q(s) = -\frac{\partial H(s)}{\partial b_j}\frac{Y(s)}{R(s)} \tag{4.63}$$

Q(s) can be expressed in polar form as shown below :

$$Q(jw) = V(w)\ e^{j\gamma(w)} \tag{4.64}$$

where $V(w) = |Q(jw)|$ and $\gamma(w) = \text{Arg}[Q(jw)]$

Then applying the same method as shown above gives :

$$\frac{\Delta M(w)}{\Delta b_j} \simeq V(w)\ M(w)\ \cos\gamma(w) \qquad (4.65)$$

or $$\frac{\Delta|W_c(jw)|}{\Delta b_j} \simeq V(w)\ M(w)\ \cos\gamma(w) \qquad (4.66)$$

In the following, the system as shown in figure 35 is used as a test model. The sensitivity functions of the system are generated based upon the equation (4.59). A PRBS signal is injected into the system as input signal. By applying the equation (4.59), the sensitivity functions $|S_{a1}^{Wc}|$, $|S_{a2}^{Wc}|$ and $|S_{a3}^{Wc}|$ are obtained and plotted in figures 42, 43 and 44 respectively. Figure 45 is also a graph of S_{a3}^{Wc} versus frequency. It can be seen from figure 45 that S_{We}^{a3} takes negative values.

The sensitivity functions generated here are compared with those generated in section 4.2 using the small perturbation method. The sensitivity functions generated by both approaches are shown in figures 46-47. It can be seen from these graphs that the sensitivity functions obtained by the two methods differ from one another. However the general shapes of the spectra still remain. The difference between two sets of spectra is partly caused by the approximation which was used in the generation of the sensitivity functions by both methods.

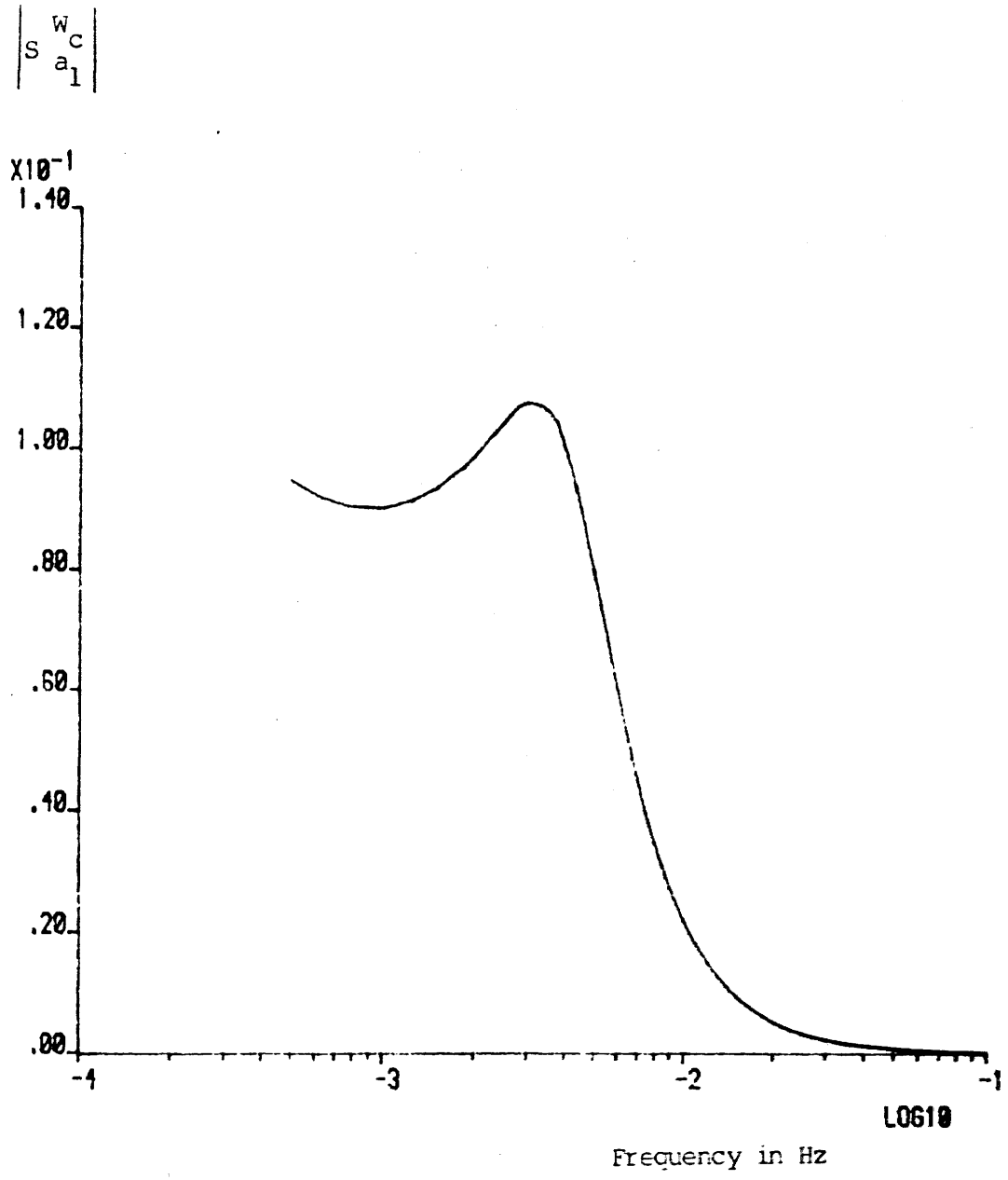


Figure 42 Frequency spectrum of the sensitivity function $\left| S_{a_1}^{w_c} \right|$ which is generated by the cosystem filter method

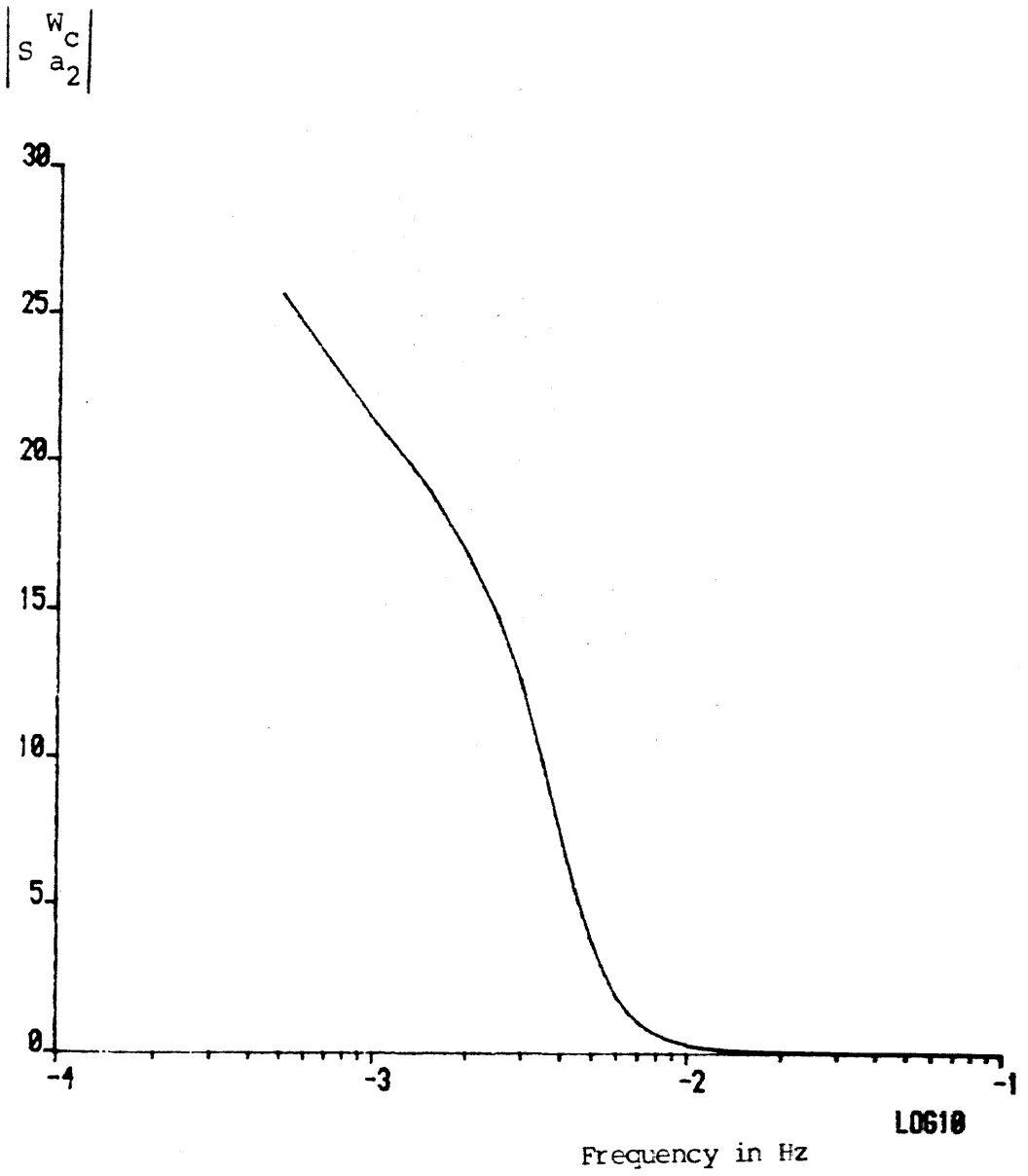


Figure 43 Frequency spectrum of the senstivity function $\left|S_{a_2}^{W_c}\right|$ which is generated by the cosystem filter method

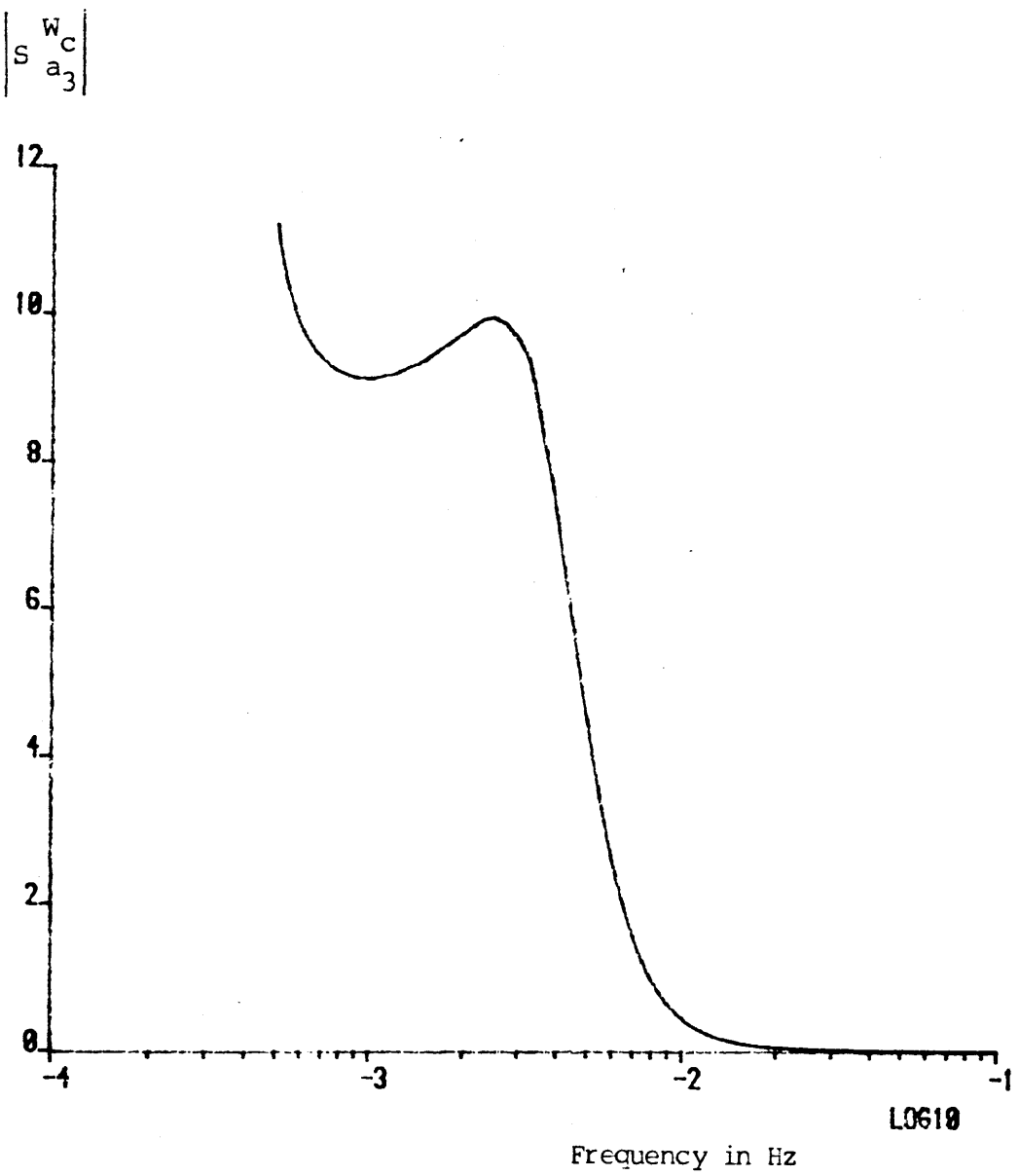


Figure 44 Frequency spectrum of the sensitivity function $\left|S_{a_3}^{w_c}\right|$ which is generated by the cosystem filter method

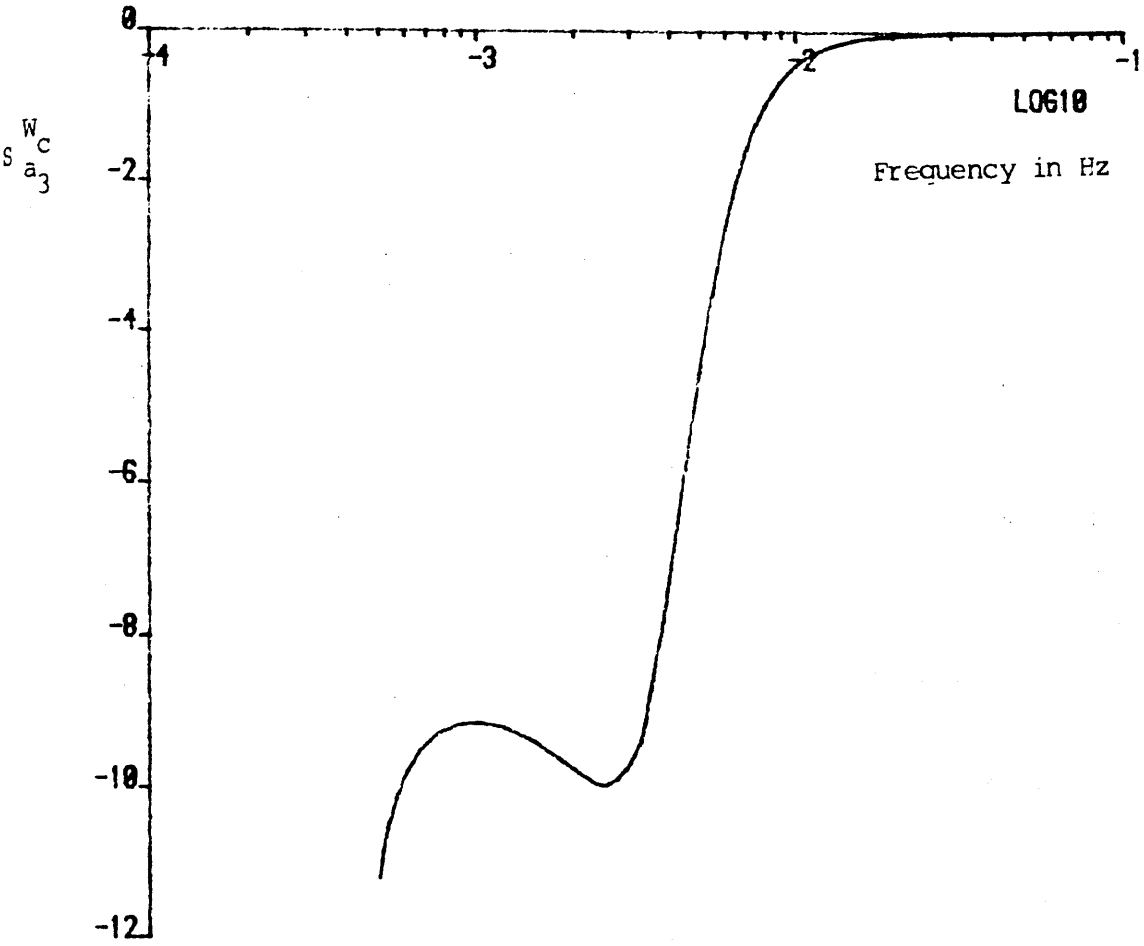


Figure 45 Frequency spectrum of the sensitivity function $S^{W_c}_{a_3}$ which is generated by the cosystem filter method

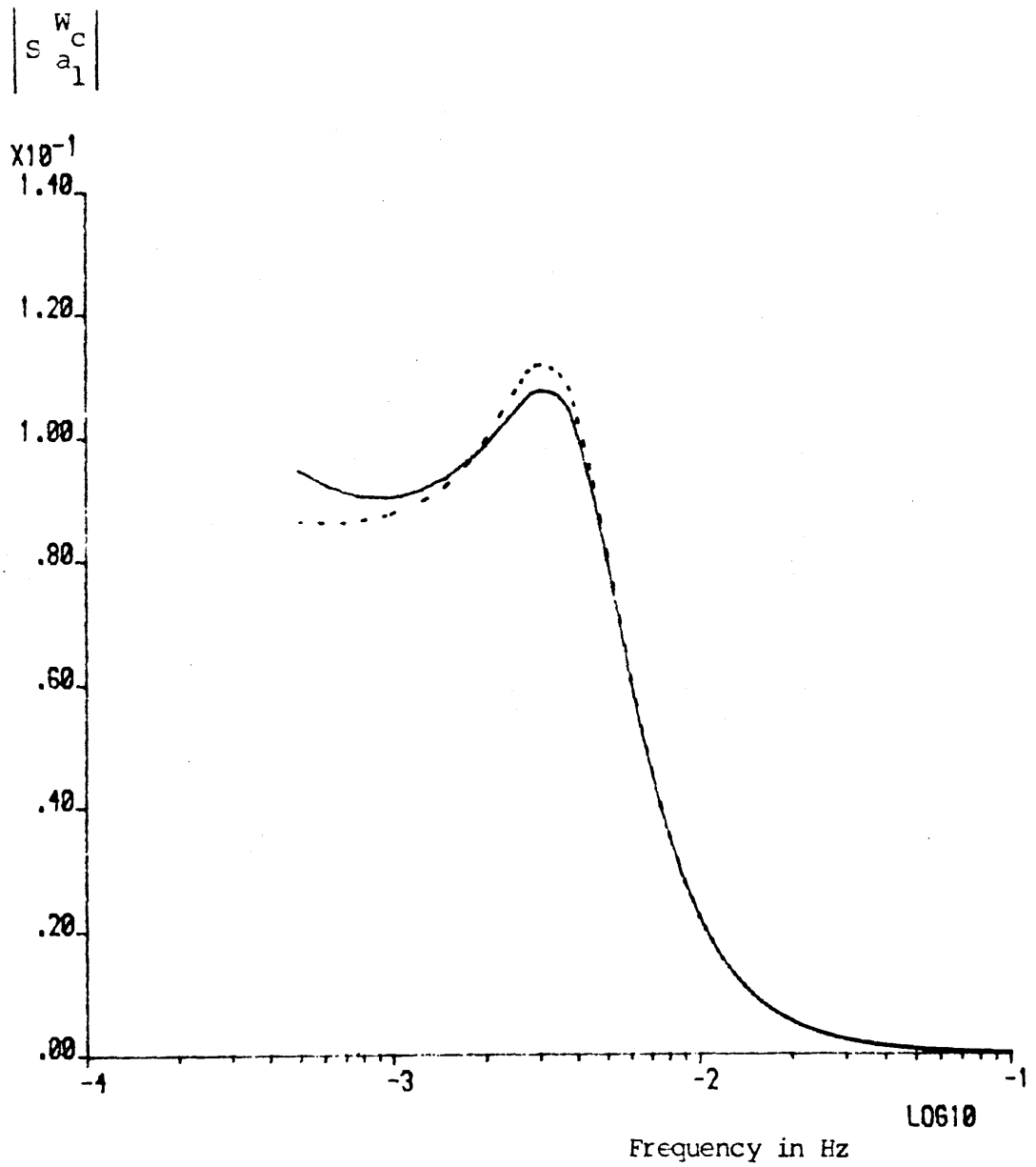


Figure 46 Frequency spectra of the sensitivity function $\left| S_{a_1}^{W_c} \right|$ generated by the perturbation method and the filter method

——— response generated by the cosystem filter method
- - - - response generated by the small perturbation method

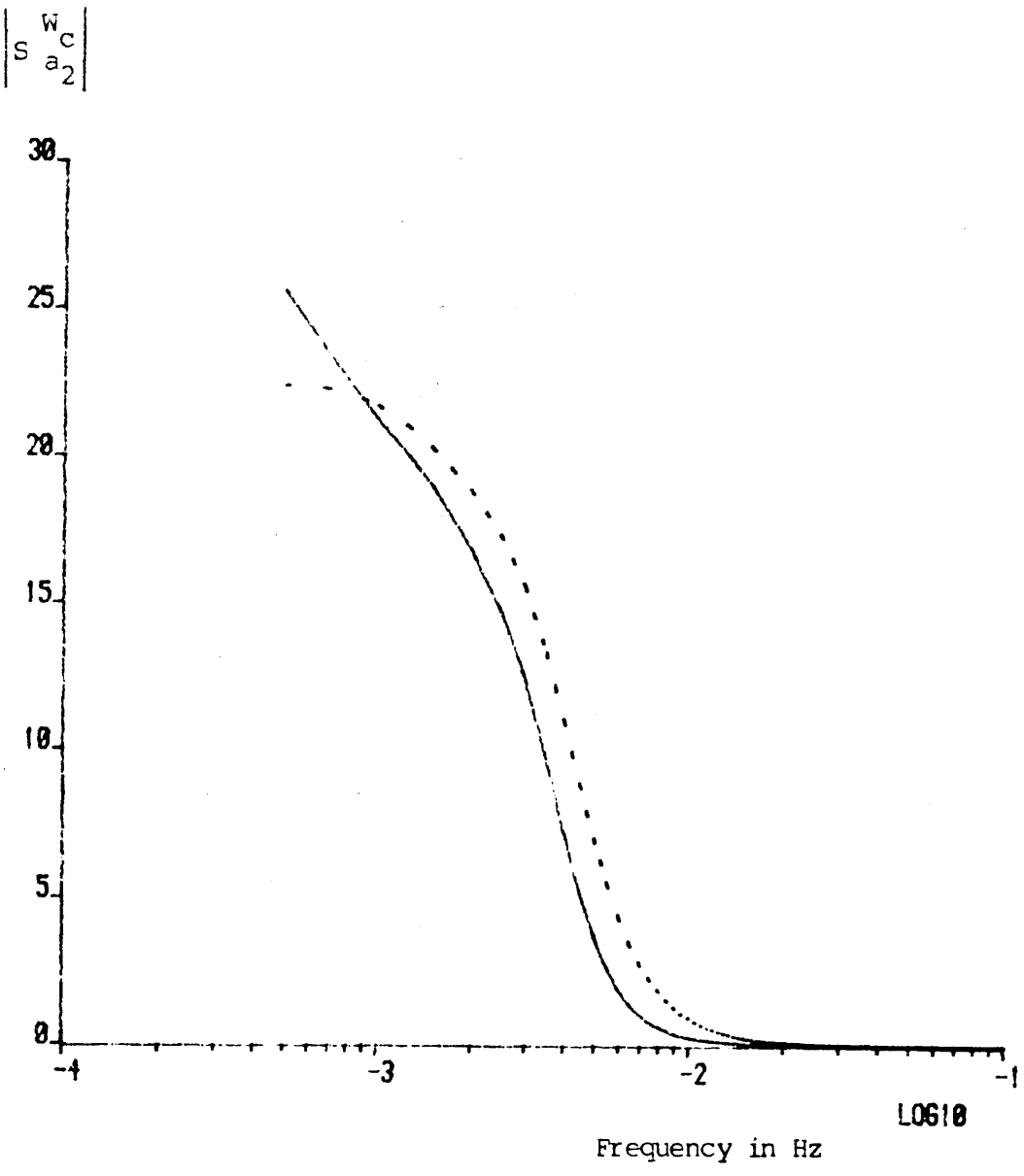


Figure 47 Frequency spectra of the sensitivity function $\left|S_{a_2}^{W_c}\right|$ generated by the perturbation method and the filter method

——— response generated by the cosystem filter method
- - - - response generated by the small perturbation method

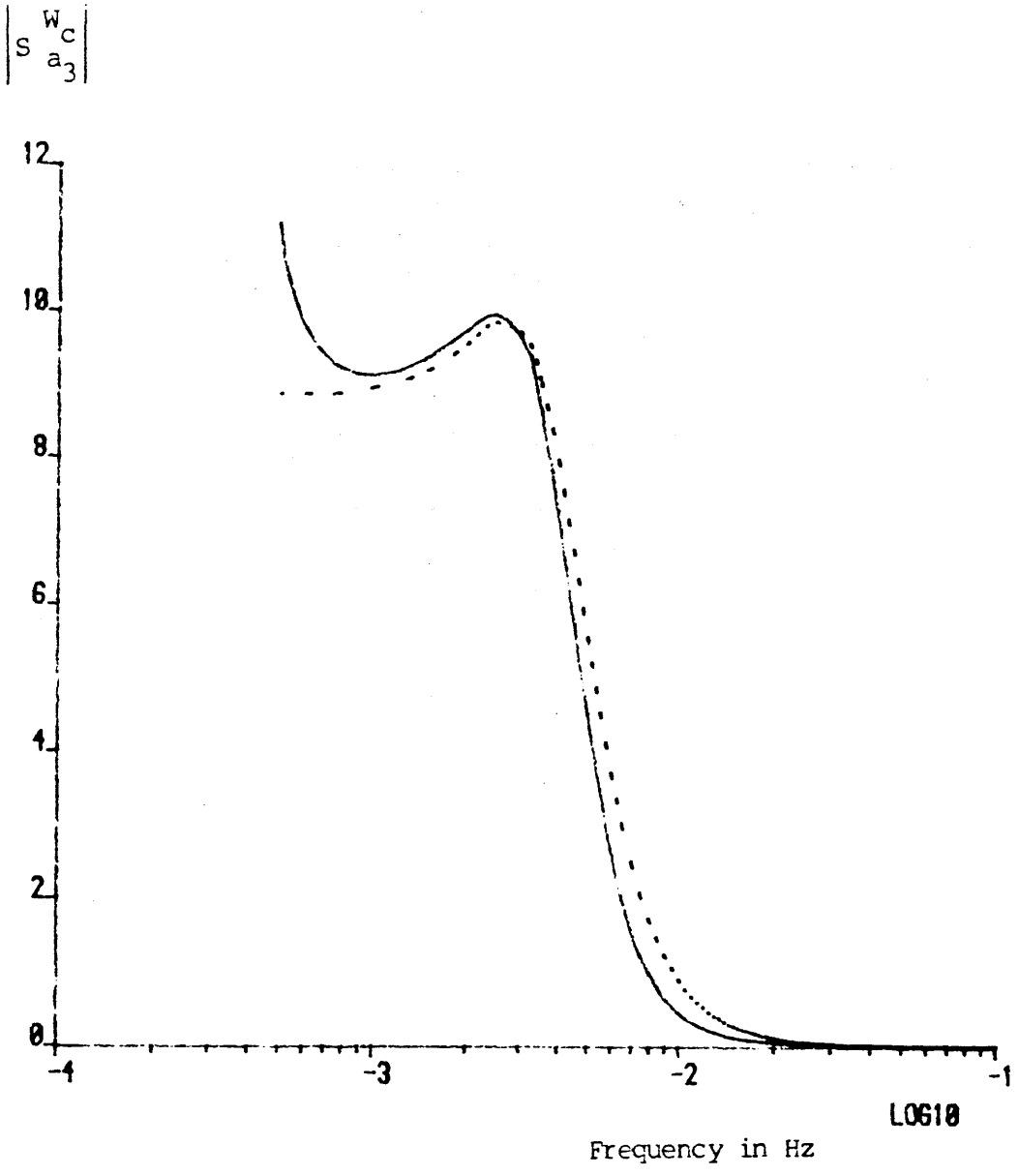


Figure 48 Frequency spectra of the senstivity function $\left|S_{a_3}^{W_c}\right|$ generated by the perturbation method and the filter method

——— response generated by the cosystem filter method
- - - - response generated by the small perturbation method

Next each of the compensator's parameters R_1, R_2 and G is increased by 100%. The effect of the parameter variation on the sensitivity functions are shown in figures 49-57. A brief description of figures 49-57 is tabulated below.

	R_1 increased by 100%	R_2 increased by 100%	R_3 increased by 100%
$\left\| S_{a_1}^{W_c} \right\|$	Fig. 49	Fig. 50	Fig. 51
$\left\| S_{a_2}^{W_c} \right\|$	52	53	54
$\left\| S_{a_3}^{W_c} \right\|$	55	56	57

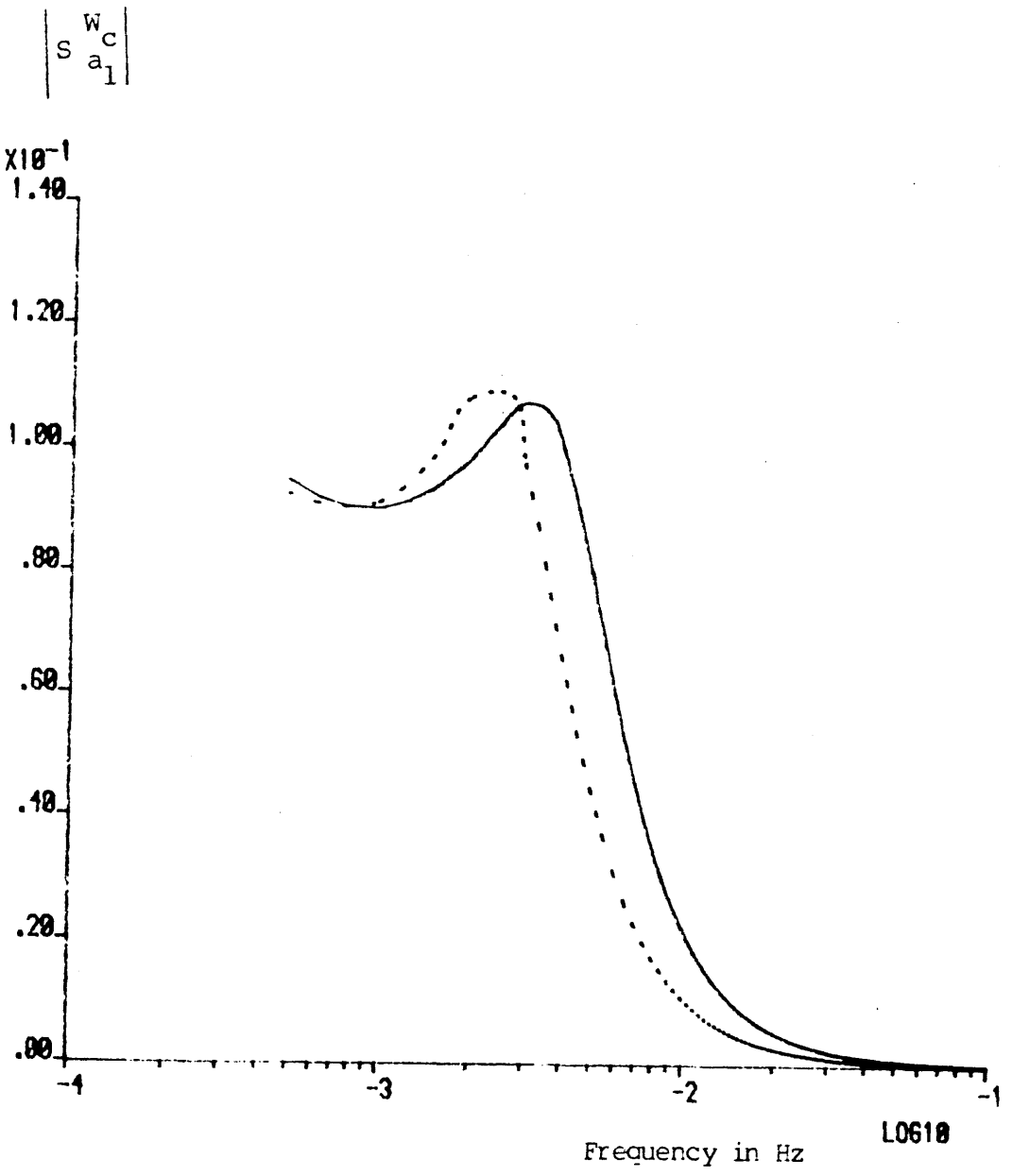


Figure 49 Frequency spectra of the sensitivity function $\left|S_{a_1}^{W_c}\right|$ before and after the parameter R_1 is increased by 100%.

——— response before the parameter changes.
- - - - response after the parameter has changed.

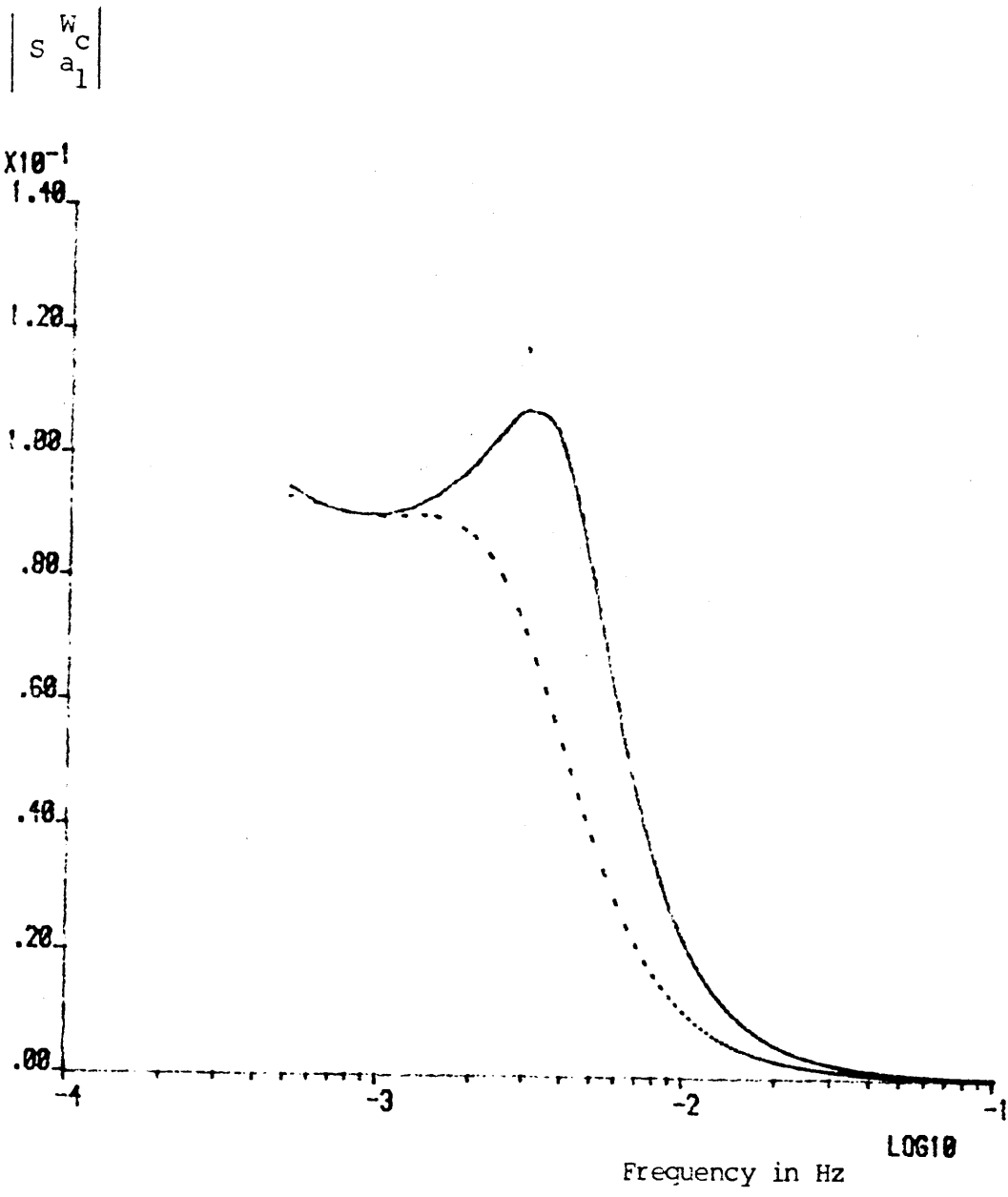


Figure 50 Frequency spectra of the sensitivity function $\left| S_{a_1}^{W_c} \right|$ before and after the parameter R_2 is increased by 100%

——— response before the parameter changes.
– – – – response after the parameter has changed.

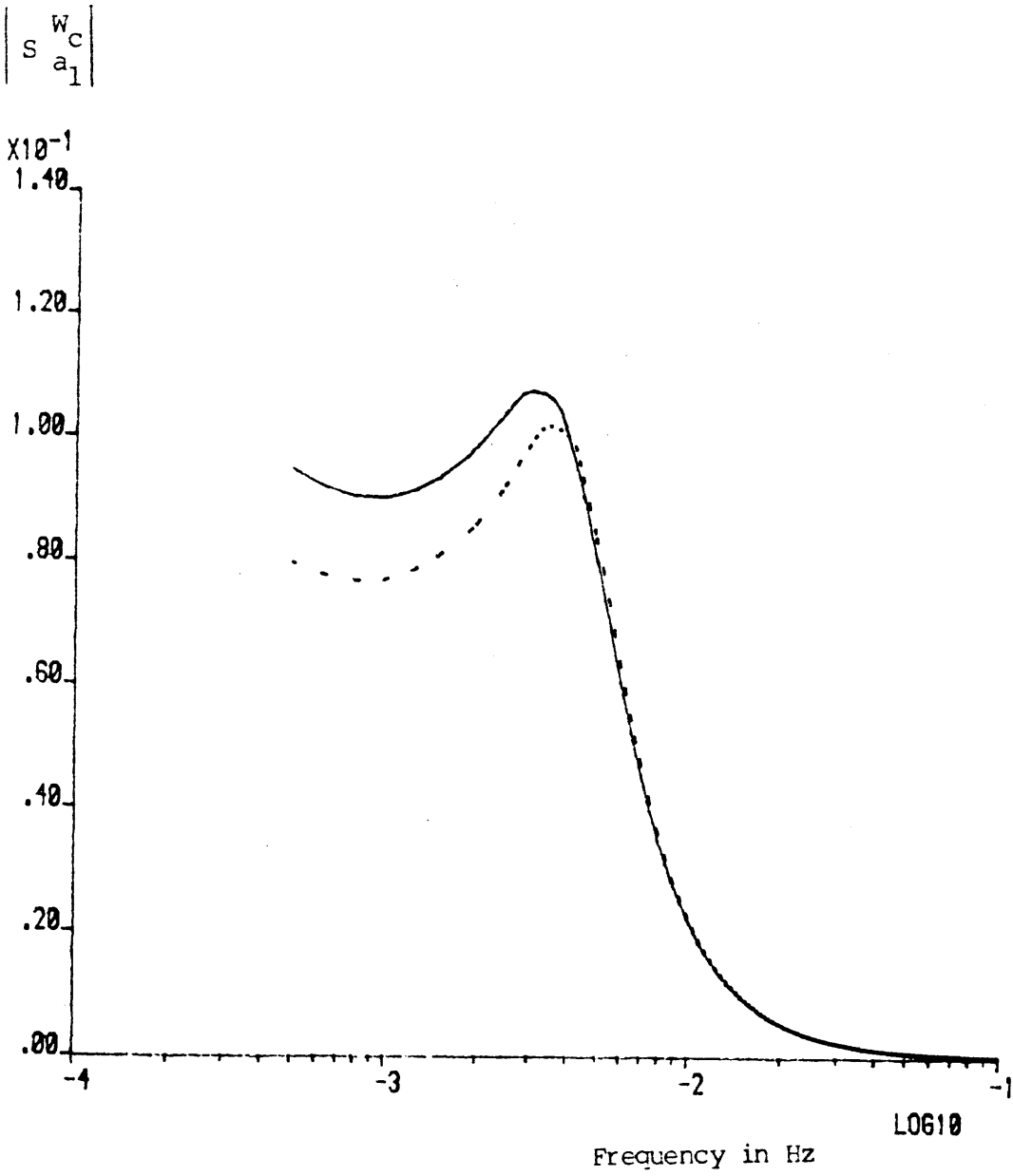


Figure 51 Frequency spectra of the sensitivity function $\left| S_{a_1}^{W_c} \right|$ before and after the parameter G is increased by 100%

——— response before the parameter changes.
- - - - response after the parameter has changed.

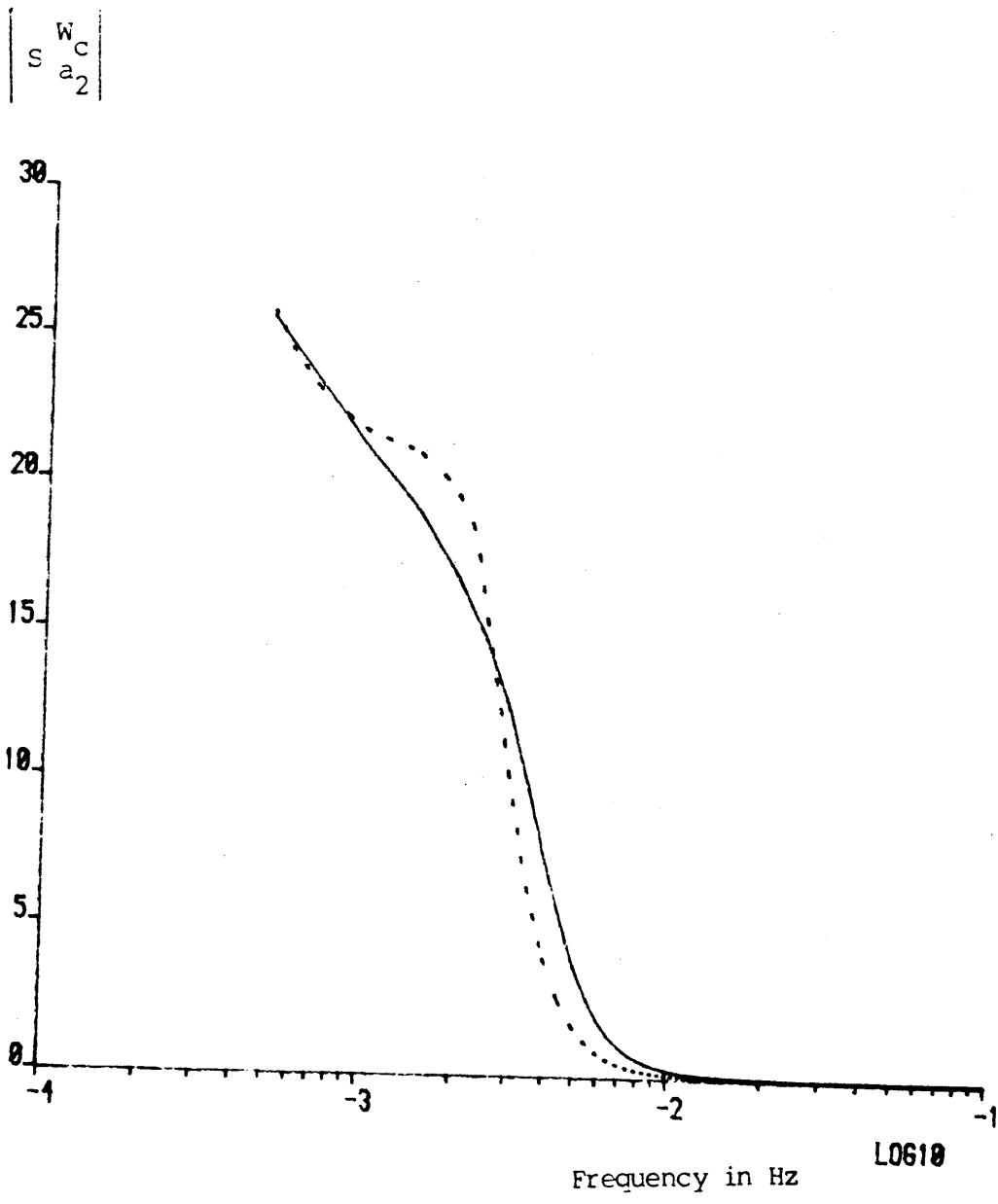


Figure 52 Frequency spectra of the sensitivity function $\left| S_{a_2}^{W_c} \right|$ before and after the parameter R_1 is increased by 100%

——— response before the parameter changes.
– – – – response after the parameter has changed.

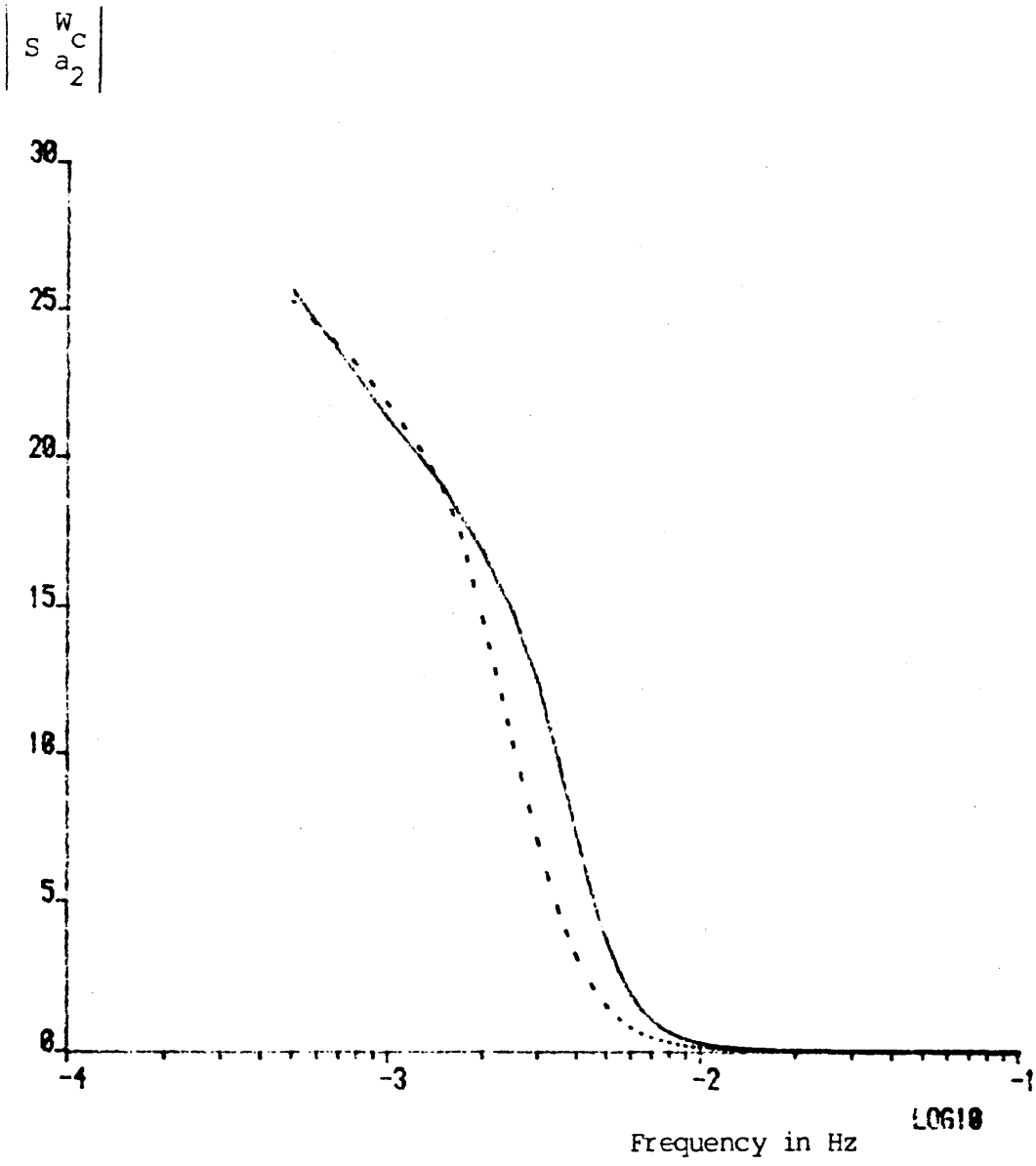


Figure 53 Frequency spectra of the sensitivity function $\left| S_{a_2}^{w_c} \right|$ before and after the parameter R_2 is increased by 100%.

——— response before the parameter changes.
- - - - response after the parameter has changed.

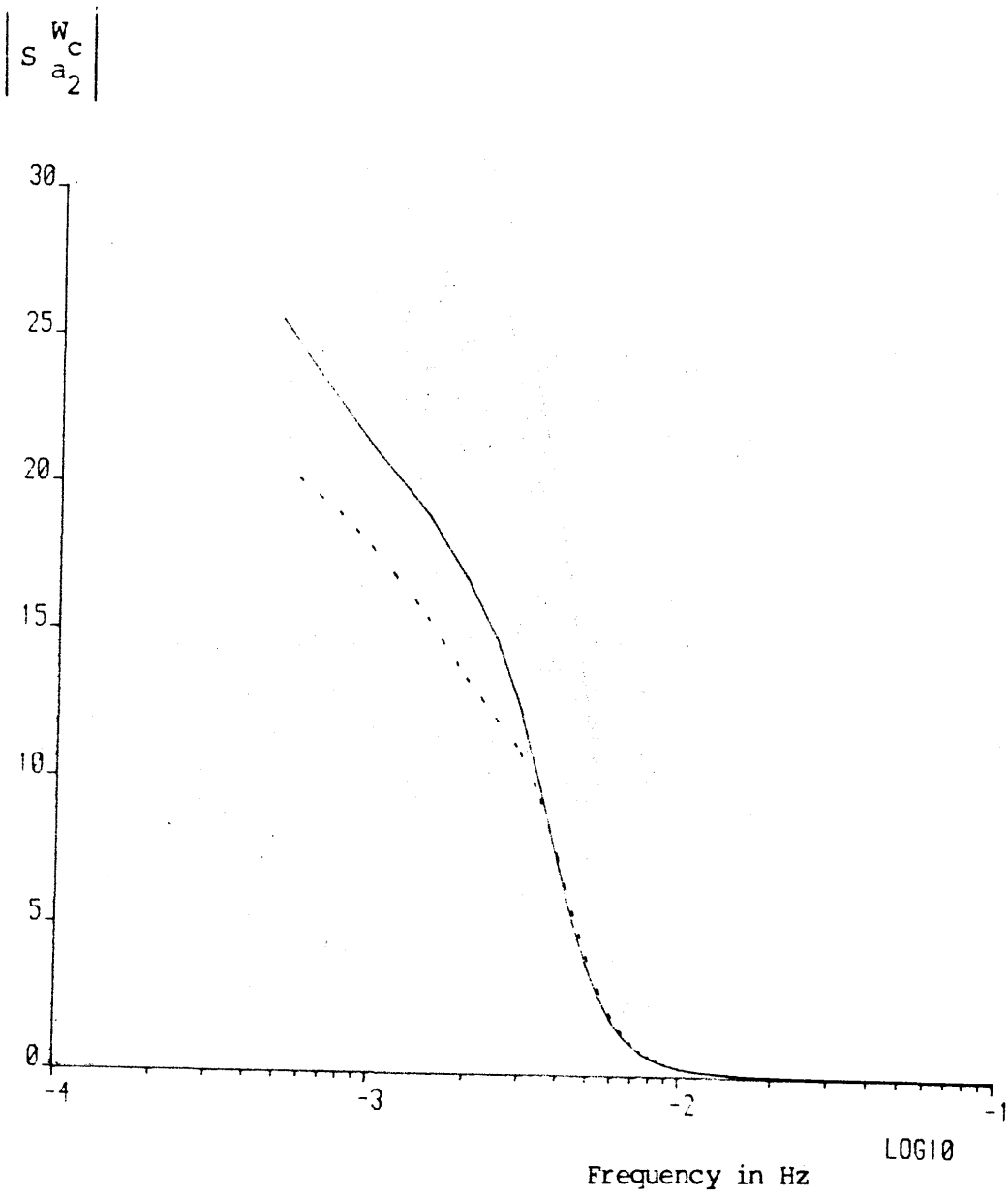


Figure 54 Frequency spectra of the sensitivity function $\left| S_{a_2}^{w_c} \right|$ before and after the parameter G is increased by 100%.

——— response before the parameter changes.
- - - - response after the parameter has changed.

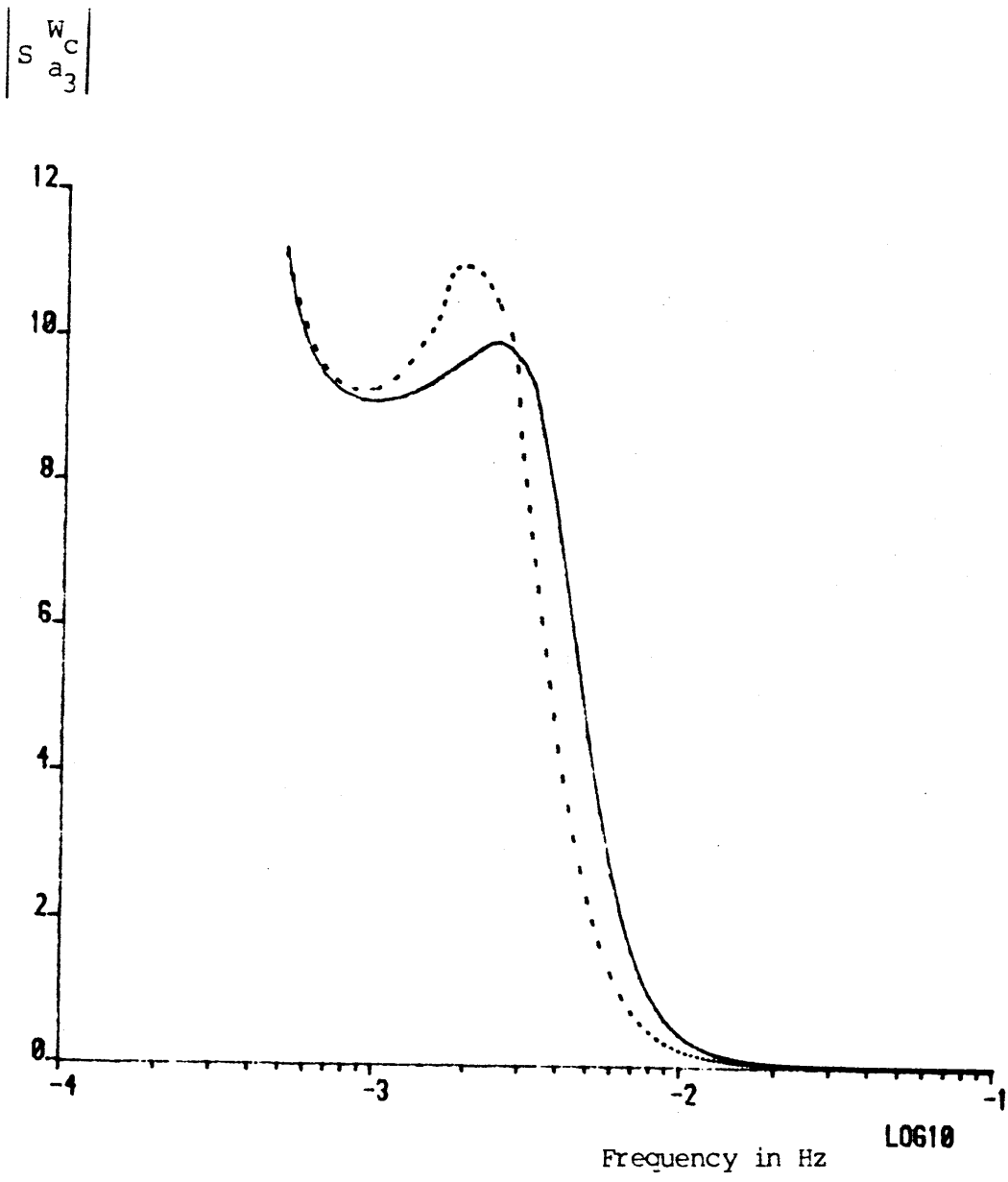


Figure 55 Frequency spectra of the sensitivity function $\left|S^{W_c}_{a_3}\right|$ before and after the parameter R_1 is increased by 100%.

——— response before the parameter changes.
- - - - response after the parameter has changed.

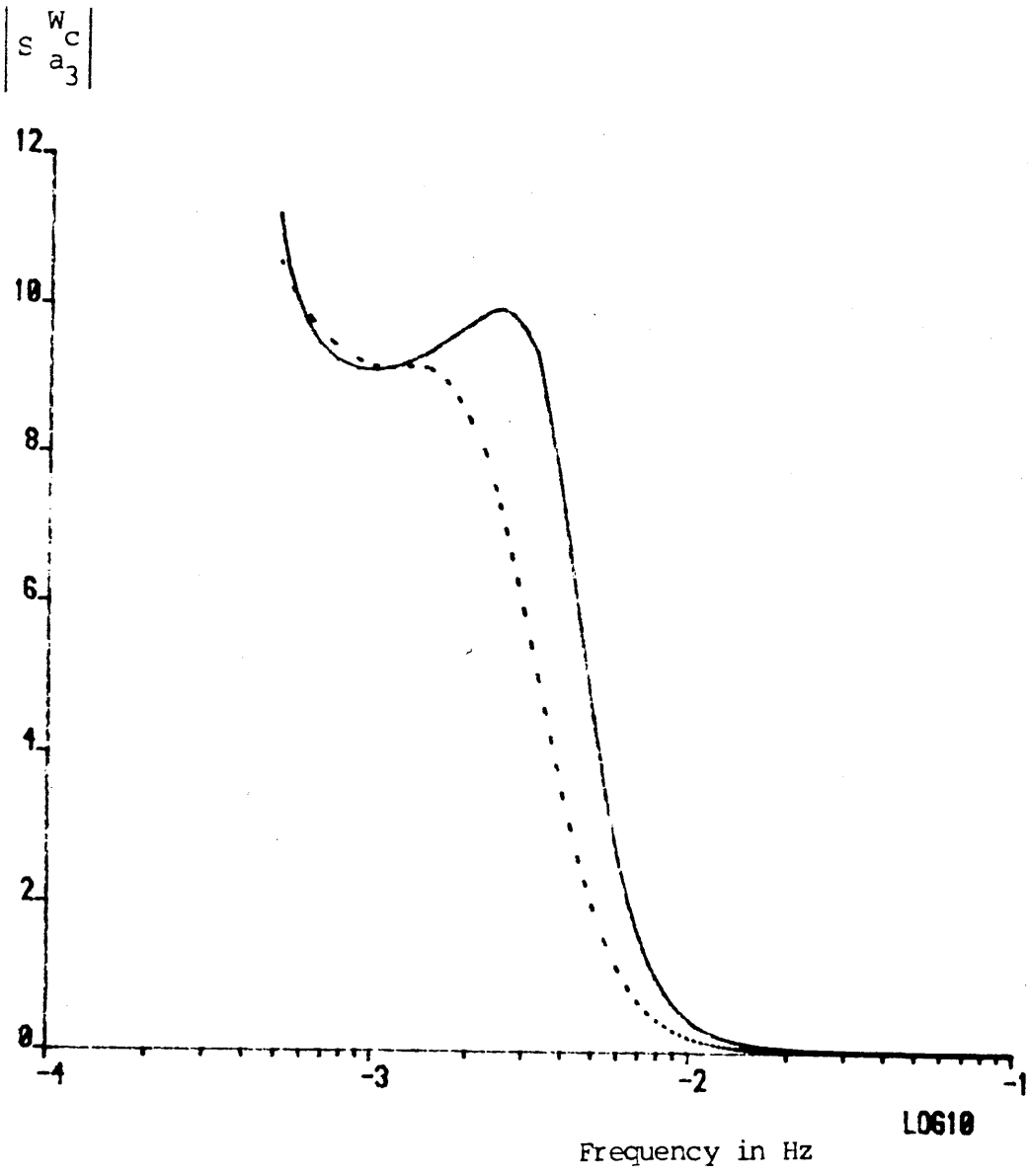


Figure 56 Frequency spectra of the sensitivity function $\left|S_{a_3}^{W_c}\right|$ before and after the parameter R_2 is increased by 100%.

——— response before the parameter changes.
- - - - response after the parameter has changed.

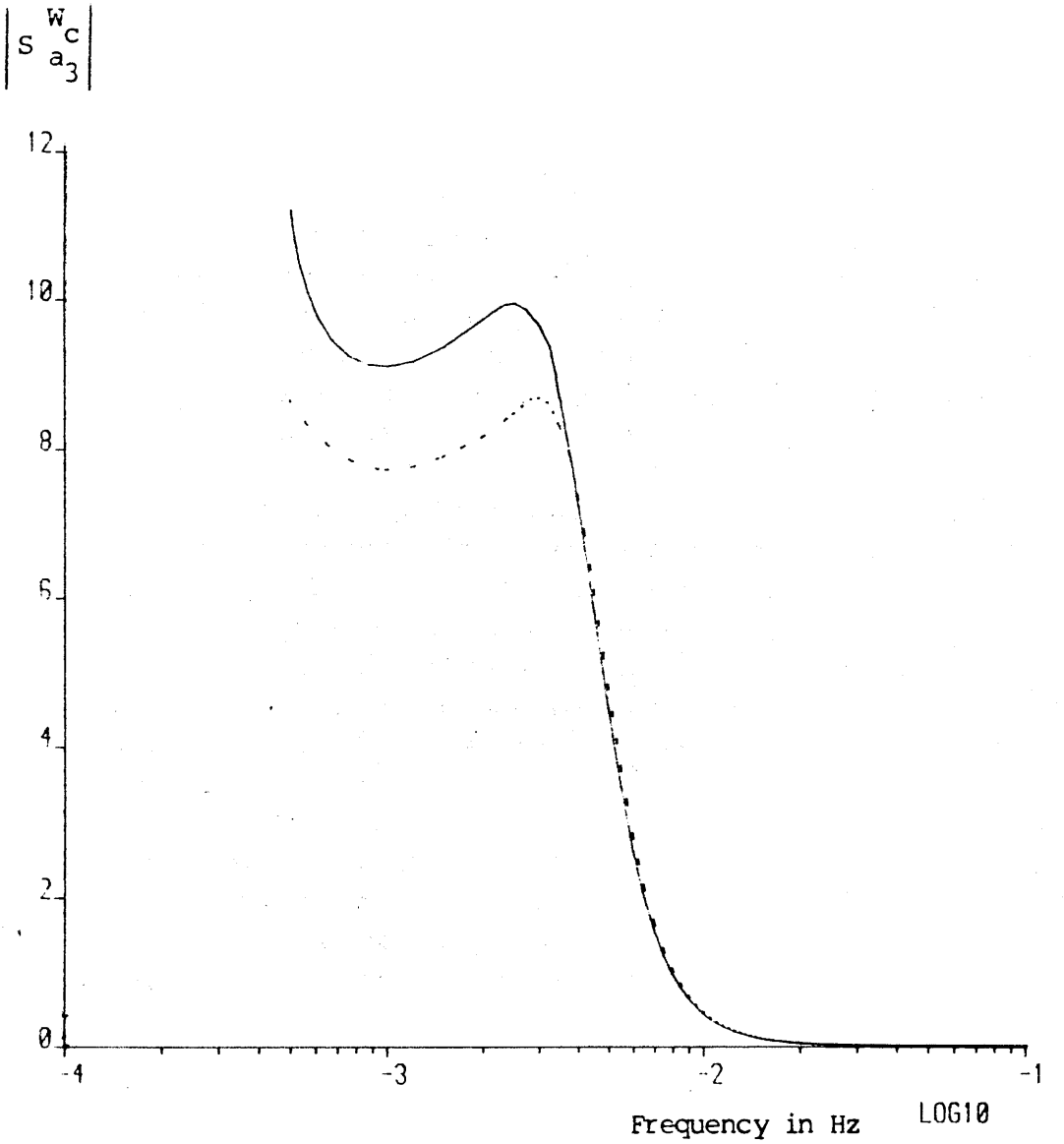


Figure 57 Frequency spectra of the sensitivity function $\left| S^{w_c}_{a_3} \right|$ before and after the parameter G is increased by 100%.

——— response before the parameter changes.
- - - - response after the parameter has changed.

4.7 System Response Optimisation based on Sensitivity Functions

Sensitivity functions were used by Winning[17], El-Shirbeeny[18] and Murray-Smith[19] for online optimsation of parameter setting in an excitation controller. The optimisation involved the use of the least square method and the sensitivity functions to obtain optimun change of parameter which forced the system response to approach a desirable form.

Sensitivity functions can also be used for fault detection and accommodation in an automatic control system. If a small change or fault does occur within the system, it is possible to make alternations on some or all of the parameters in the system controller so as to acquire a normal system response even though the system is faulty. In the cases when the fault is within an acceptable range, the optimisation can be applied and the desired system response can be maintained at all times. Fault diagnosis or fault correction can be carried out without interrupting the system process. In some cases when the fault is vigorous, a desired system response can still be sustained after the application of parameter optimisation even though some state variable may be shifted to a dangerous level. Under such circumstances, the system should be shut down. Sensitivity method used for parameter adjustment has been implemented in the time domain. In the following, the sensitivity will be implemented in the frequency domain. The optimisation is achieved by analysing the frequency magnitude spectra of the sensitivity functions. The magnitude spectrum of a sensitivity function can be derived by using the method described in the last section.

Figure (35) shows a diagram of a two-tank flow control system which is controlled by a cascade conpensator. The cascade compensator is there to ensure that the overall system response meets the given specifications on system performance. The compensators C(s) contains m adjustable parameters. G(s) is the transfer function of the plant being monitored.

The transfer function T(s) of the overall system can be expressed in polar form as follows :

$$T(jf) = M(f)e^{j\phi(f)} \tag{4.67}$$

where M(f) is the magnitude spectrum of T(jf)
and $\phi(f)$ is the phase spectrum of T(jf)

In the following section, only the magnitude spectrum will be used for optimisation of the system response. When a small change in the plant G(s) occurs, the overall system response will change. The magnitude spectrum of the system response after the parameter change has occured can be expressed by the truncated series expansion :

$$M(f, a_0 + \Delta a) \simeq M(f, a_0) + \sum_{i=1}^{m}\left(\frac{\partial M(f)}{\partial a_i}\Delta a_i\right), \quad \text{where } m = \text{no. of variable parameters}$$

$$= M(f, a_0) + \sum_{i=1}^{m}\left(S_{a_i}^{M}(f)\,\Delta a_i\right) + R_M(f) \tag{4.68}$$

$M(f, a_0 + \Delta a)$ and $M(f, a_0)$ represent the magnitude spectra after and before the parameter change respectively. If the sensitivity functions ate known, and provided $R_M(f)$ is small, one can make the spectrum $M(f,a_0)$ resemble $M(f, a_0+\Delta a)$. This can be done by simply making appropriate changes of the parameters a_i.

Now supposing $M(f, a_0+\Delta a)$ is the desirable response $M_d(f)$ gives:

$$M_d(f) = M(f, a_0+ \Delta a) \tag{4.69}$$

From equation (4.68), $M_d(f)$ can be expressed as :

$$M_d(f) = M(f, a_0) + \sum_{i=1}^{m} [S^M_{a_i} (f) . \Delta a_i] + R_M(f) \qquad (4.70)$$

The desirable change ΔM_d is defined as the change which will make $M_d(f)$ approximately equal to $M(f, a_0)$. It is given by :

$$\Delta M_d(f) = M_d(f) - M(f, a_0) \qquad (4.71)$$

$$= \sum_{i=1}^{m} \left[S^M_{a_i} (f) . \Delta a_i \right] + R_M(f) \qquad (4.72)$$

The sensitivity terms in equation (4.72) represent the change of system response caused by appropriate adjustments of the parameter $a_1, a_2, \ldots, a_m$. This change in system response is called the 'synthesis change', $\Delta M_s(f)$, which is defined as :

$$\Delta M_s(f) = \sum_{i=1}^{m} S^M_{a_i} (f) . \Delta a_i \qquad (4.73)$$

Subsitiuting (4.73) into (4.72) gives

$$R_M(f) = \Delta M_d(f) - \Delta M_s(f) \qquad (4.74)$$

As it can be seen from equation (4.74) that it is necessary to miminise the value of $R_M(f)$ in order to make the synthesis change as close to the desired change as possible. In an ideal case when $\Delta M_d(f)$ is equal to $\Delta M_s(f)$, equation (4.70) can be written as :

$$M_d(f) = M(f, a_0) + \sum_{i=1}^{m} [S^M_{a_i} (f) . \Delta a_i] \qquad (4.75)$$

In the other words, the spectrum $M(f, a_0)$ can be made equal to $M_d(f)$ by changing the parameters $a_1, a_2, \ldots, a_m$. However in

practical situations, $\Delta M_d(f)$ is not always equal to $\Delta M_s(f)$. Therefore it is necessary to minimise $R_M(f)$ so as to make $M_d(f)$ resemble $M(f, a_0)$ as closely as possible. One common minimisation method which can be used is the 'Intergral Least Square Error'. The idea is to minimise the value of J which is given by :

$$J = \int_{f_s/m}^{f_s} \{ R_M(f) \}^2 \, df \qquad (4.76)$$

f_s is the sampling frequency in Hertz.
f_s/m is the frequency of each sequence of signal in Hertz.

By focusing on the square of the error $R_M(f)$, it penalizes both positive and negative values of $R_M(f)$. Substituting equations (4.73) and (4.74) into equation (4.76) gives :

$$J = \int_{f_s/m}^{f_s} \{\Delta M_d(f) - \sum_{i=1}^{m} [S^M_{a_i}(f) . \Delta a_i] \}^2 \, df \qquad (4.77)$$

Equation (4.77) can be expressed in sampled data form as follow :

$$J = \sum_{l=0}^{g} \{\Delta M_d(l.\ f) - \sum_{i=1}^{m} [S^M_{a_i}(l.\ f) . a_i] \}^2 \Delta f \qquad (4.78)$$

where g = no. of sampled data

To find the minimum value of J with respect to a particular change a_k, the derivative of J is set to zero as shown below :

$$\frac{\partial J}{\partial a_k} = 0 \qquad \text{for all k} \qquad (4.79)$$

ie $$\sum_{l=c}^{g} 2\{\Delta M_d(l.\Delta f) - \sum_{i=1}^{m} [S^M_{a_i}(l.\Delta f) . a_i] \} \, S^M_{a_i}(l.\Delta f) \Delta f = 0$$

$$\text{for all k} \qquad (4.80')$$

From equation (4.80), the k simultaneous equations are obtained. $\Delta M_d(f)$ can be found from equation (4.71) and the sensitivity functions can be derived by using the method described in section (4.6). Equation (4.80) will represent m linear equations with m

variables a_1, a_2, ..., a_m. The values of these parameters can be obtained by solving the m linear equations. These values represent the set of parameter changes which will minimise the difference between the synthesised change $\Delta M_s(f)$ and the desired desired change $\Delta M_d(f)$. The result of the minimisation can be improved by repeating the minimisation process several times.

The optimisation process is carried out on the system as shown in figure 35. The value of the parameter R_2 is increased from 226s/m^2 to 330s/m^2. In the first test, optimisation is carried by varying all the variables of the compensator, i.e. a_1, a_2 and a_3. The frequency spectra are plotted in figure 58. In the second test, only the variable a_1 and a_2 are varied. The frequency spectra are shown in figure 59. In the third test, only a_1 is varied. The frequency spectra are shown in figure 60. In each of the three tests as mentioned above, the system frequency responses, after the optimisation process has been carried five times, are shown in figures 61, 62 and 63. It can be seen from these graphs that the optimisation achieved by varying three variables produces better result than that by varying only two or one of the variables. The program, which generates the sensitivity functions using the cosystem filter approach and carries out the optimisation process, is shown in appendix D.

This optimisation technique using sensitivity functions requires a well-defined controller of which the parameters can be varied. In case when no controller is involved in the system or when the specifications of the controller is not given, then it is necessary to define part of the plant of which the parameters can be controlled and altered accordingly.

In the situation when no specification of the system under controlled is given at all, the whole system is treated as a "black box". The optimisation technique using the system sensitivity functions cannot be applied because it requires specification of at least part of the system itself. This is a major drawback of using such optimisation technique.

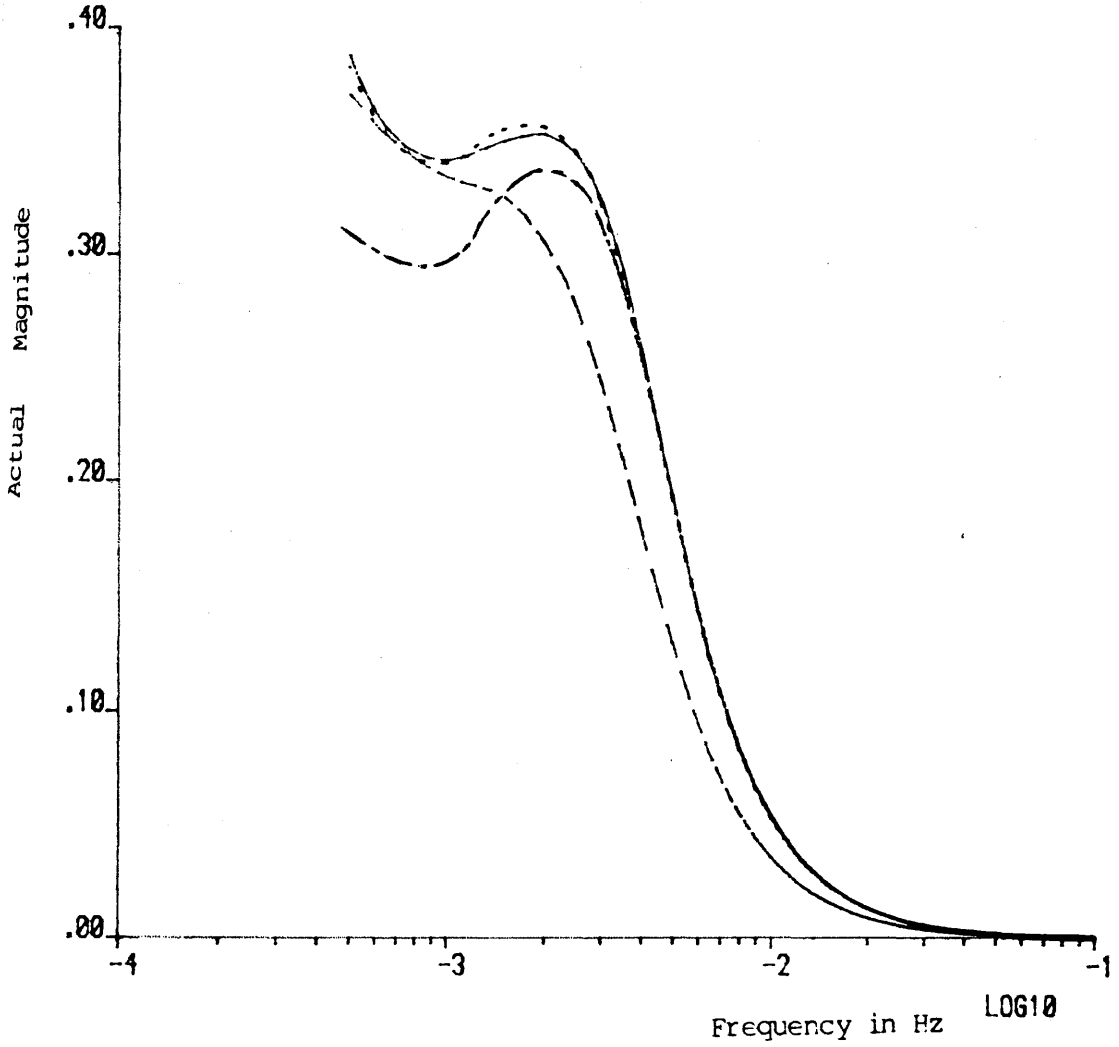


Figure 58 Frequency responses of the two tank system.
Optimisation is carried by varying parameters a_1, a_2 and a_3.

—— normal system response.

Test 1 : — — system response when fault occurs.

Test 2 : —·— system response after the first optimisation process has been carried out.

Test 3 : ···· system response after the second optimisation process has been carried out.

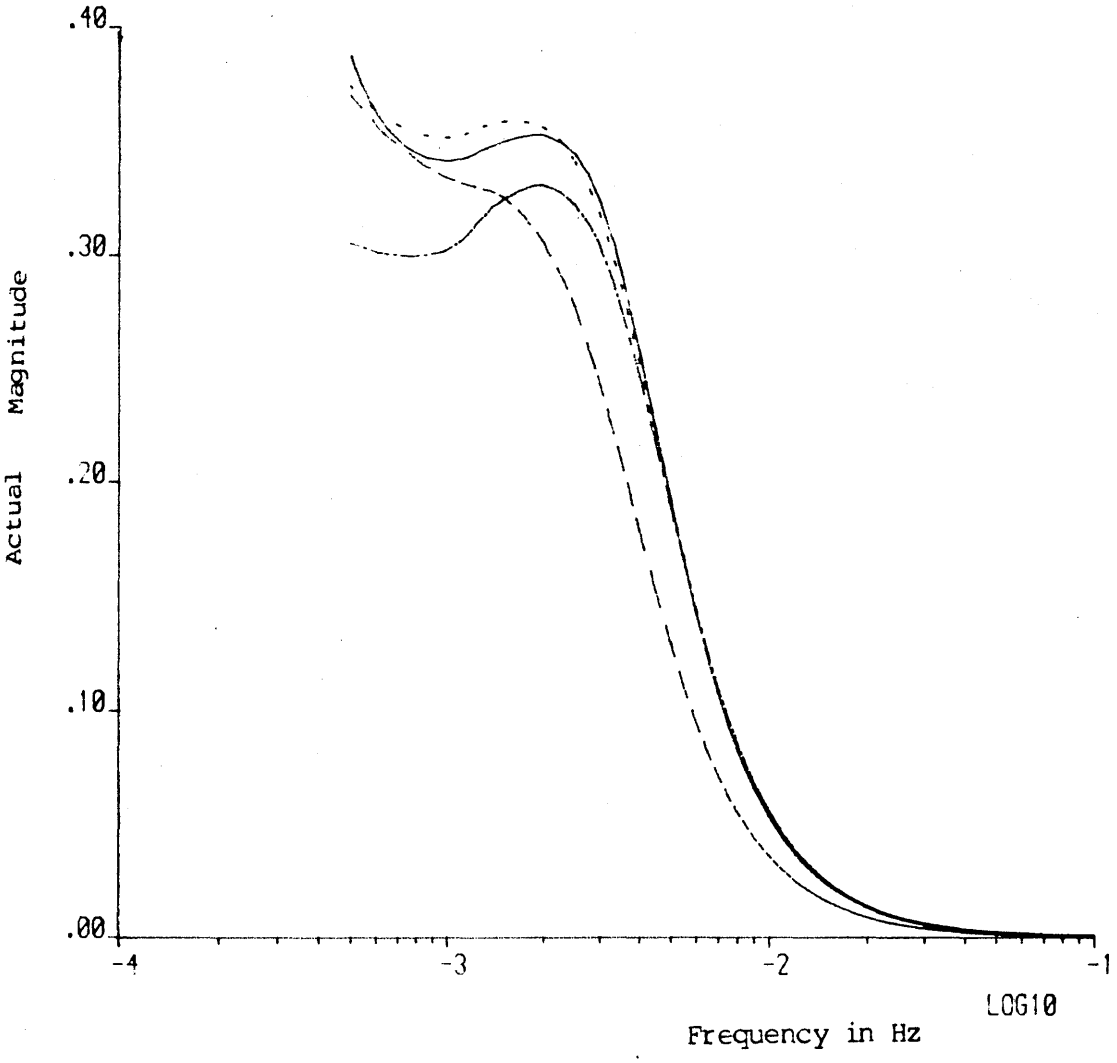


Figure 59 Frequency responses of the two tank system.
Optimisation is carried by varying parameters a_1 and a_2.

—— normal system response.

Test 1 : — — system response when fault occurs.

Test 2 : —·— system response after the first optimisation process has been carried out.

Test 3 : ---- system response after the second optimisation process has been carried out.

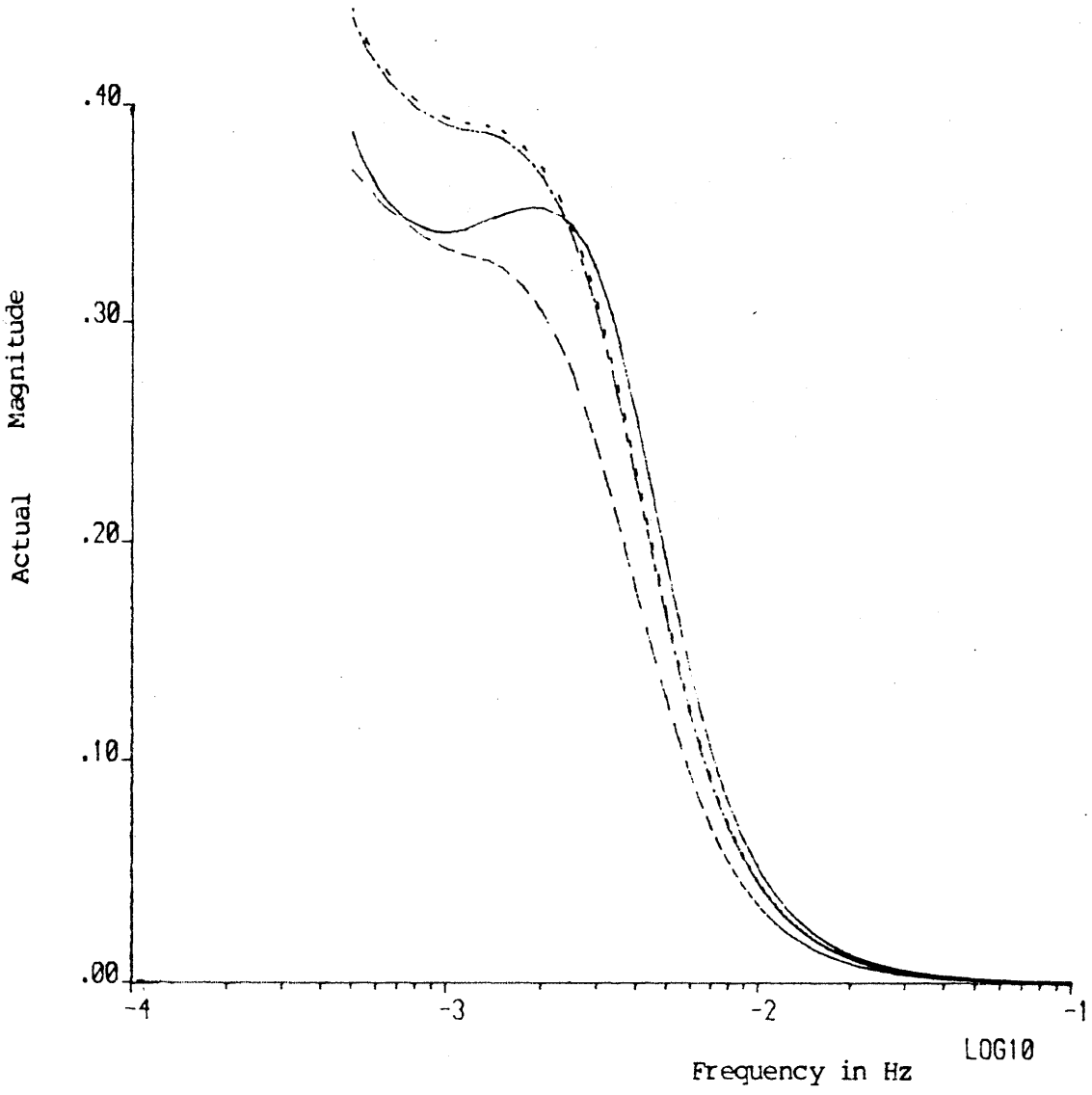


Figure 60 Frequency responses of the two tank system.
Optimisation is carried by varying the parameter a_1.

—— normal system response.

Test 1 : — — system response when fault occurs.

Test 2 : —·— system response after the first optimisation process has been carried out.

Test 3 : ---- system response after the second optimistion process has been carried out.

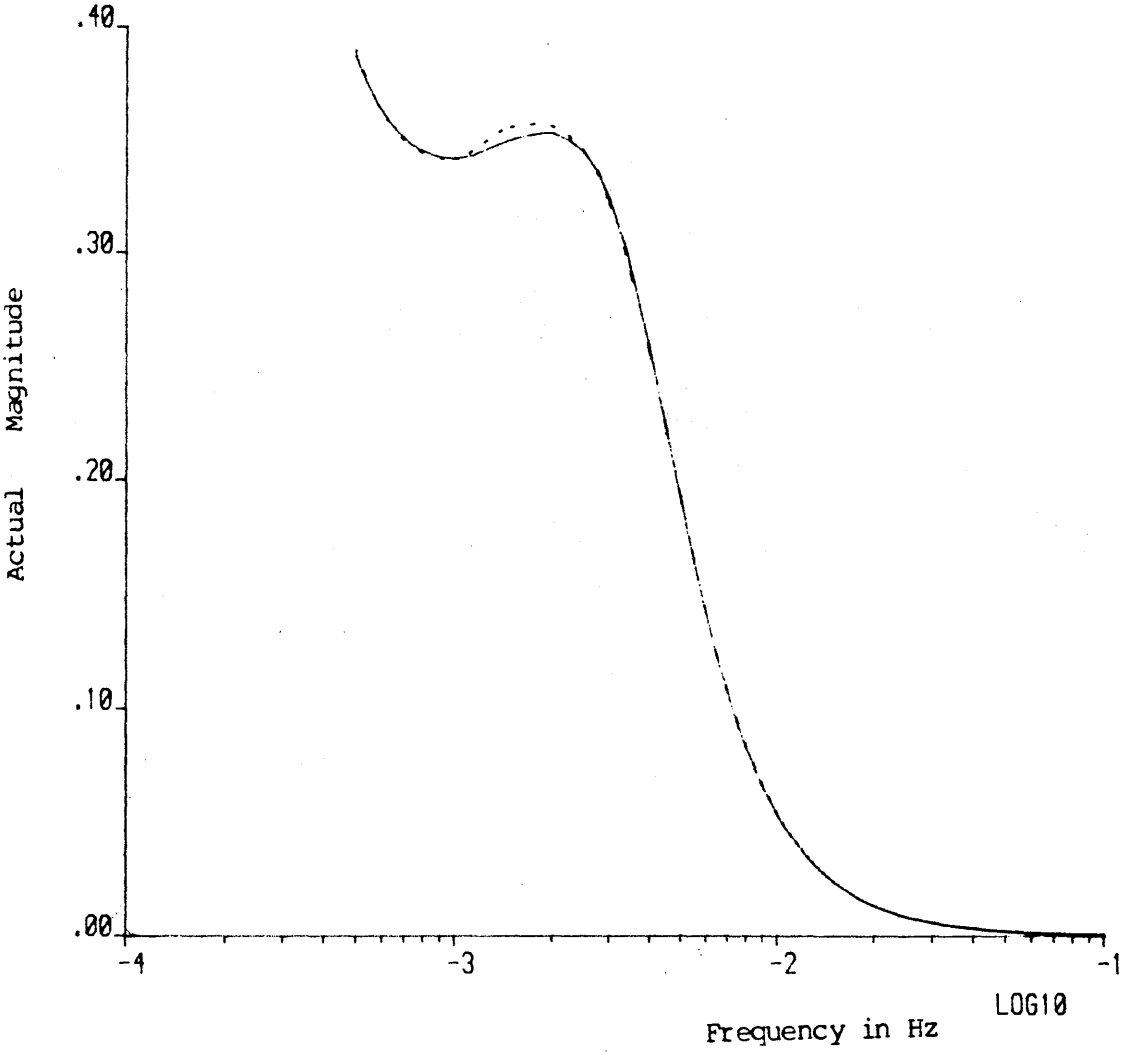


Figure 61 Frequency responses of the two tank system.
Optimisation is carried by varying parameters a_1, a_2 and a_3.
—— normal system response.
Test 6 : ---- system response after the fifth optimisation process has been carried out.

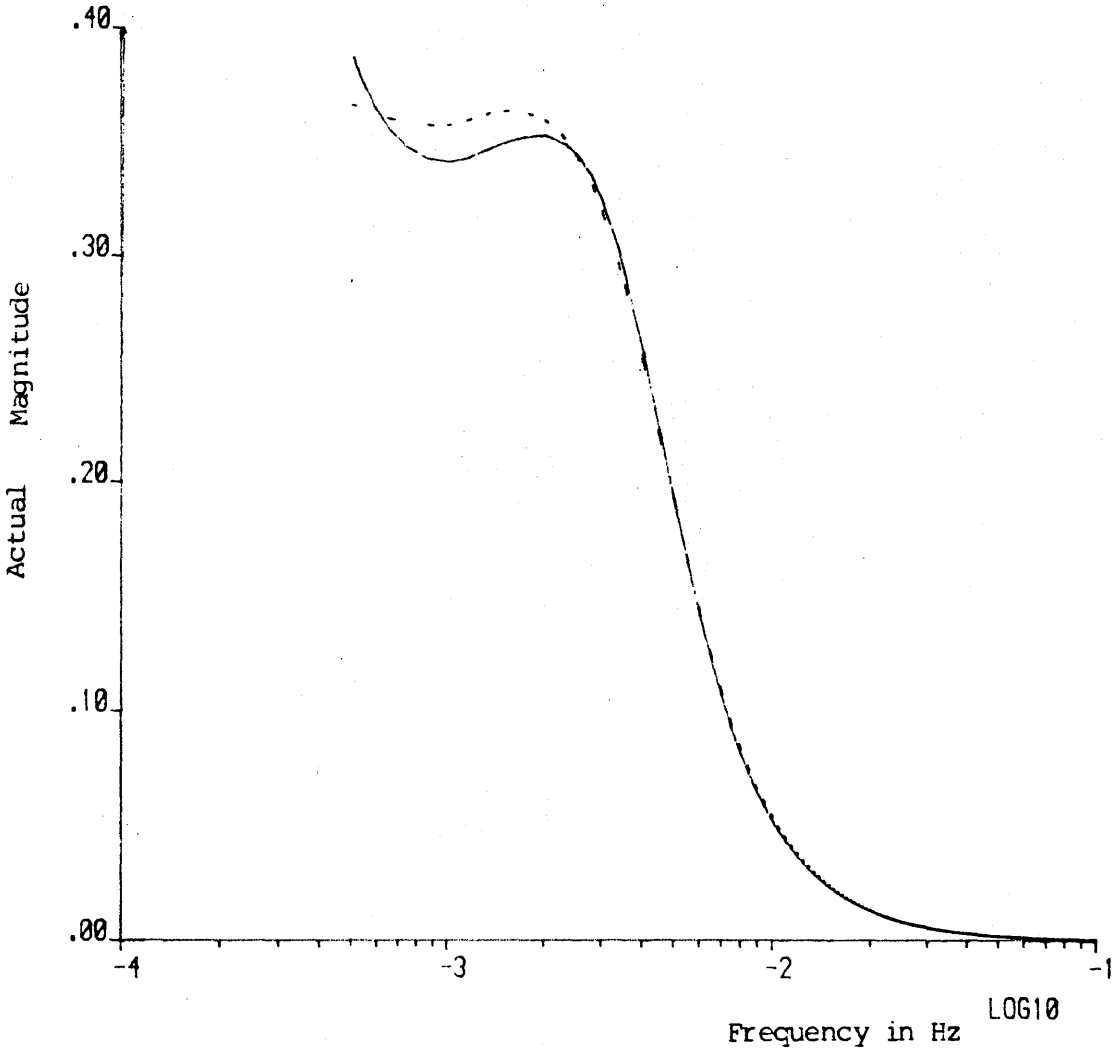


Figure 62 Frequency responses of the two tank system.
Optimisation is carried out by varying parameters a_1 and a_2.
—— normal system response.
Test 6 : ---- system response after the fifth optimisation process has been carried out.

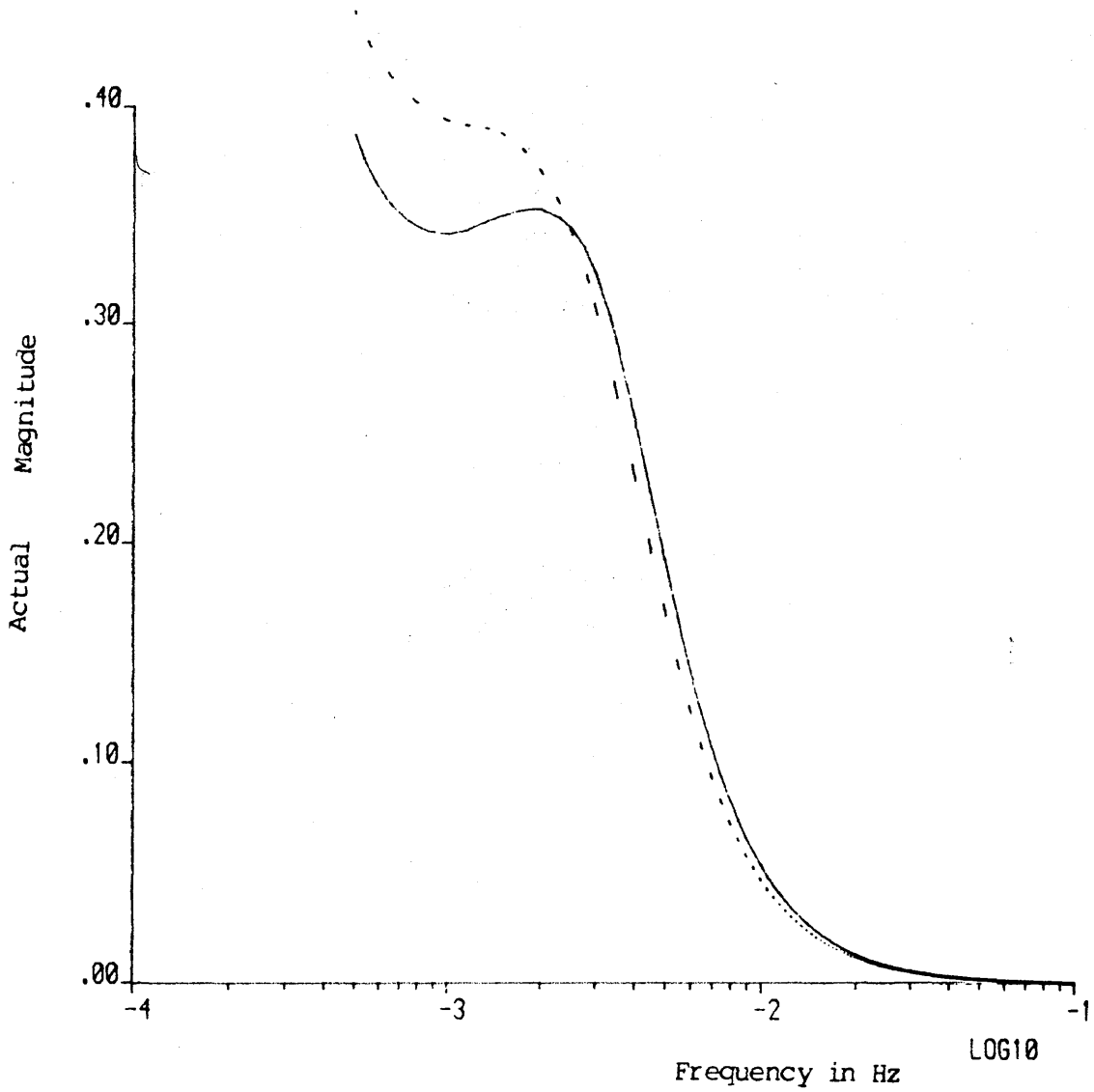


Figure 63 Frequency responses of the two tank system.
Optimisation is carried by varying the parameter a_1.
—— normal system response.
Test 6 : ---- system response after the fifth optimisation process has been carried out.

4.8 Convergence Criteria

As mentioned above, the minimisation of J can be improved by repeating the minimisation process several times. A point will be reached when no further minimisation can improve the result. Therefore it is essential to introduce a performance index which indicates the degree of resemblance between the desired response and the actual response. It can also indicate when any repetition of the minimisation process would not give any significant benefit.

The performance index is defined as :

$$P = \int_{fs/m}^{fs} [\, M(f) - M_d(f) \,]^2 \, df \qquad (4.81)$$

or in digital form :

$$P = \sum_{l=1}^{m} \{ [\, M(l.\Delta f) - M_d(l.\Delta f) \,]^2 . \Delta f \} \qquad (4.82)$$

For each value of P, the ratio between its current value and its previous value i.e. P_n / P_{n-1} provides an indication of the degree of convergence of the system response toward a desirable form. When the ratio is approaching unity, convergence can be inferred. This means that very little or no improvement of optimisation can be obtained by repeating the minimisation process. However if the ratio is less than unity, then improvement of the system response can still be obtained by further minimisations.

The performance indices of the optimisation process carried out in section 4.7 are tabulated as follows.

R_2 CHANGED FROM 226s/m^2 TO 330s/m^2

Number of tests	Varying a_1, a_2, a_3	Varying a_1, a_2	Varying a_1
1	5.067757E-5	5.067757E-5	5.067757E-5
2	6.209528E-6	7.402507E-6	1.114045E-5
3	6.220910E-8	4.552550E-7	1.086281E-5
4	6.366821E-8	5.573858E-7	1.085892E-5
5	6.798405E-8	8.036273E-7	1.085761E-5

A graph of the performance index versus the number of tests is shown on the next page. It can be seen that the variation of three parameters gives better optimisation results than the variation of only two or one of the parameter.

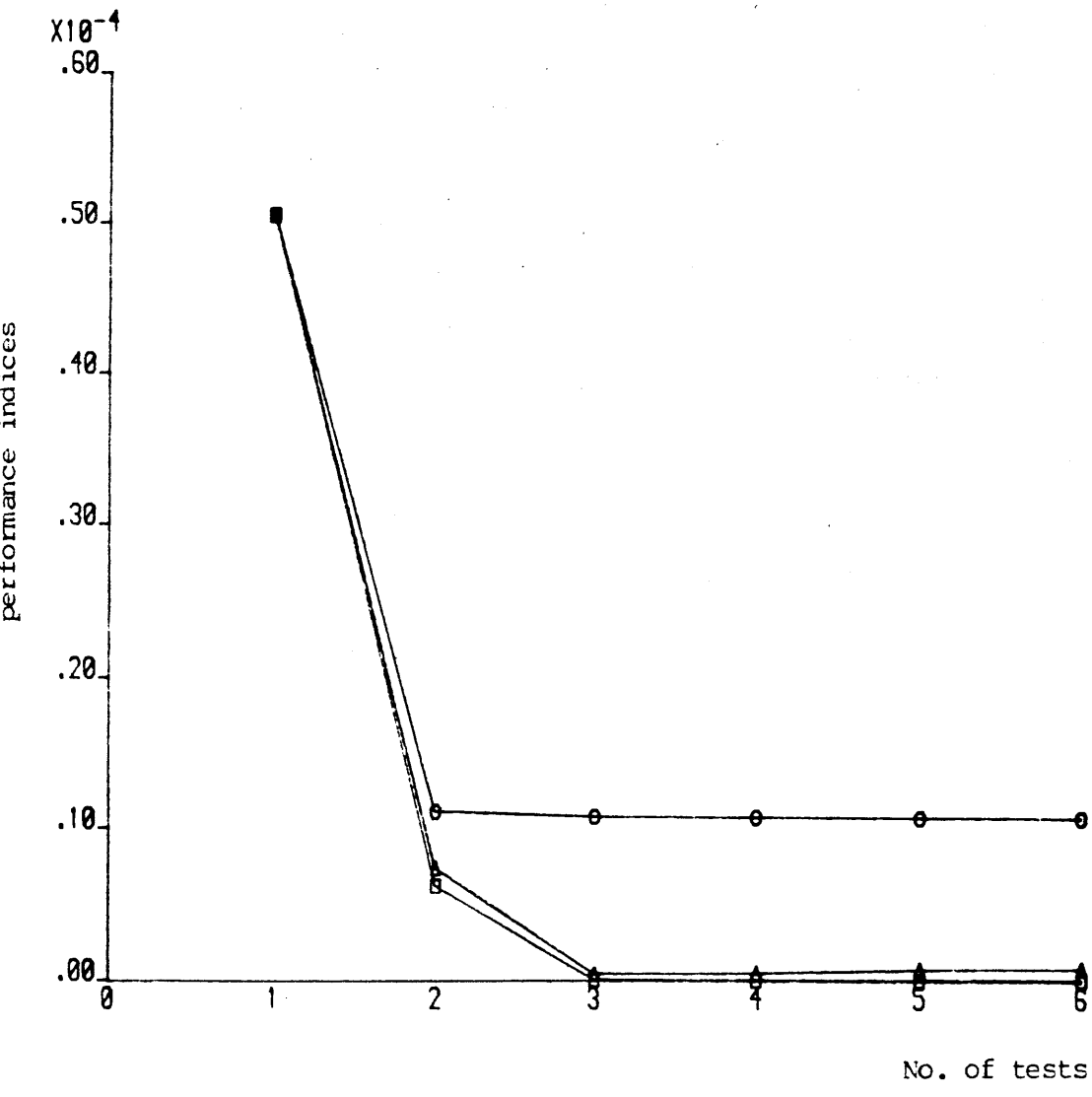


Figure 64 Plot of Performance Indices versus No. of Tests

o——o optimisation by varying a_1, a_2 and a_3.
△——△ optimisation by varying a_1 and a_2.
□——□ optimisation by varying a_1.

CHAPTER FIVE

5. DISCUSSION AND CONCLUSIONS

5.1 Discussion

Many fault detection techniques used in the past required a complete specification of both the system parameters and structure which are not always provided in practical situations. The system identification technique based on the frequency domain, as shown in chapters 2 and 3, has been achieved without needing any detailed knowledge of the system dynamics. The frequency response of the system is obtained by Fourier tranforming the system input and output signals. System fault can then be shown and identified from the variation of the system frequency response. This fault detection method needs only the knowledge of the input and output signal, thus making it applicable in circumstances when the system specifications are not known.

In all the experiments which have been carried out, the mixed radix-2 Fast Fourier Transformation Algorithm has been used successfully to transform the system input and output signals. The use of the mixed radix-2 algorithm avoided the incompatability between the pseudo-random-binary-sequence signal and the conventional radix-2 Fast Fourier Tranformation. However the implementation of the mixed radix-2 FFT is slower than that of the conventional radix-2 FFT. For this reason, the latter is more suitable to apply on an on-line situation than the former. The incompatibility between the pseudo-random-binary-sequence and the conventional radix-2 FFT can be solved by using different sampling rate as suggested by Poussart and Ganguly(13).

From the graphs plotted in figures 20-31, it can be seen that the change of each parameter has a distinguishable effect on the system frequency response. By identifying the variation of the system response with respect to a parameter change, system faults occurring in the system can be classified. This fault detection technique has a drawback when there are two or more parameters which have the same effect on the system response. Under such circumstances, this fault detection technique will be unable to detect the exact parameter which causes the system response variation. From the diagram shown in figure 15, it can be seen that the parameter A_1 has the same effect on the system response as the parameter R_1, with a similar situation between parameters A_2 and R_2.

Although the exact fault parameter cannot be precisely identified in some cases, this fault detection techniques based on the observation of system frequency response can certainly provide some valuable information about the size, location and possiblity of the system fault.

Apart from having observed the system frequency response upon parameter changes, similiar observations have been made on the system sensitivity functions.

System sensitivity functions, showing the sensitivity of selected output variables to the vatiation of controller parameters, were generated by the small perturbation method (as shown in section 4.2) and by the sensitivity filter method (as shown in section 4.3). The sensitivity functions generated by the two methods show a slight difference between them. The difference is partly caused by the approximation and limitation which apply in both methods. The graphs shown in figures indicate that each parameter change has its unique effect on each sensitivity function. The effect of parameter change upon the sensitivity functions can be identified, therefore allowing the classification of any system failure to be carried out.

The small perturbation method is impractical on an on-line situation because n+1 separate tests are required to obtain the sensitivity functions of n parameters. If n is large, the computation time for calculating n sensitivity functions will become very long and impractical. The sensitivity filter method, however, can produce a simultaneous estimates of the sensitivity functions. For this reason, the sensitivity filter method is more appropriate to use for on-line situation than the small perturbation method.

Although the generation of the system sensitivity functions for controller parameters does not require any knowledge of the plant under controll, it does however need a precise specification of the system compensation controller or the feedback controller. In cases where interest is focused on plant variations then at least the specification of part of the plant must be given to enable the derivation of the sensitivity functions. In chapter four, the sensitivity functions were generated based upon knowledge of the compensation controller only.

The sensitivity functions obtained in chapter four were used to carry out optimisation on the two tanks liquid flow control system. The optimisation is based upon an integral squared error criterion. It aimed to force a faulty system response back to as close to its normal response as possible. The optimisation of the three paramters a_1, a_2 and a_3 produced better results than that of parameters a_1 and a_2 or of parameter a_1 only.

The optimisation process provides a means by which a faulty system can be forced to give normal system response, thus maintaining a normal system performance and producing a normal system output. It enables the plant to continue its operation even when the system is faulty. To some extent, the optimisation is not wanted because it will conceal the effect of the system fault on the system response, thus covering up the system fault when it occurs. This will make the fault detection process difficult to carry out. Optimisation should only be used when the system fault is tolerable and will not cause any dangerous hazard to the system. The process may then be allowed to continue. On the other hand, when the fault is not tolerable then appropriate action should be imposed e.g. change of operation or even stopping the whole process.

5.2 Conclusions

In the past, the potential of the frequency response approach for fault identification in closed loop systems has not been fully realised. However in this thesis, an attempt is made to apply frequency domain methods.

The system frequency response and the system seneitivity function were expressed in the frequency domain and could be used for fault identification. The fault identificatin techniques, as described in this thesis, do not require a full specfication of the system under controlled. These methods are particularly useful in some practical situations where the detailed structural and parametric information is not given.

The fault detection techniques which are based upon the effect of parameters upon the system frequency response and the system sensitivity functions has a major drawback. This happens when there are two or more parameters which have the same effect of the system frequency response of the system sensitivity function, it becomes impossible to identify the exact system fault or its location. Nevertheless, the information obtained from the system response and system sensitivity functions is important for the classification of possible faults which could occur in the system.

System optimisation based upon the sensitivity functions in the frequency domain has been carried out successfully as shown in chapter four. The technique can be used for the optimisation of a closed-loop system response when a small system fault occurs.

The fault detection techniques decribed above require a detail study of each parametric effect on the system response and the system sensitivity functions. By using a model system, each kind of fault and its unique frequency signature can be tabulated and with this information future occurances of similar faults may be found in a real system.

LIST OF REFERENCES

1. JERVIS M.W. : 'Control and instrumentation of large nuclear power stations : A review of future trends', IEE Proceedings, Vol. 131, Pt. A, No. 7, September 1984.

2. LEES F.P. : 'Some work on Instrument Malfunction Detection in the Process Industries', IEE Colloquium on Fault Detection, Colloquium Digest No. 1984/84, 29 December 1984.

3. ISERMANN R. : 'Process Fault Detection Based on Modelling and Estimation Method - A Survey', Automatica, Vol. 20, No. 4, pp.387-404, 1984.

4. HAM P.A.L. : 'The Application of Redundancy in Controllers for High Capital Cost or High Integrity Plant', Fourth International Conference on Trends in On-Line Computer Control Systems, Warwick, England, 5-8 April 1982 (London, England : IEE 1982), pp.14-17.

5. JOHNSON E. : 'Redundancy for On-Line Computer Control System', Reliability Engineering, 4(1983), pp.105-125.

6. NAGRATH I.J. and GOPAL M. : 'Control Systems Engineering', Wiley, 1982, pp.509-511.

7. RICHARDS R.J. : 'An introduction to dynamics and control', Longman, 1979, pp.487-501.

8. CORBIN M.J. and JONES J.G. : 'Analytical Redundancy using Band-Limiting Filters, Part 2 : Filter Design', Technical Memorandum FS(F) 573, Space 343, Royal Aircraft Establishment, 1984.

9. WELLSTEAD P.E. : 'Spectral Analysis and Applications', Lecture Notes, Control System Centre, UMIST.

10. McLEAN D. and AL-KHATIB K. : 'Simulation Study of and Analytically-Redundant Flight Control System', in 'Proceedings 1984 UKSC Conference on Computer Simulation', pp.275-288, Butterworths, London, 1984.

11. DAVIES W.D.T. : 'System Identification for Self-Adaptive Control', Wiley, 1970, pp.25-83.

12. GOLTEN J.W., RYLEY A. and MATTERSON A.J. : 'Frequency-response estimates using discrete signals and the radix-2 f.f.t.', Proc. IEE, Vol. 125, No. 9, 1978, pp.891-894.

13. POUSSART D. and GANGULY U.S. : 'Rapid Measurement of System Kinetics - An Instrument for Real-Time Transfer Function Analysis', Proceedings of the IEE, Vol. 65, No. 5, May 1977.

14. LAMB J.D. and REES D. : 'Digital processing of system responses to pseudo random binary sequences to obtain frequency response characteristics using the fast Fourier transform' in 'The digital use of conputers', IEE Conf. Publ. 103, 1973, pp.141-146.

15. JENKINS G.M. and WATTS D.G. : 'Spectral Analysis and its Applications', Holden-Day, 1968, pp.356-359.

16. SINGLETON R.C. : 'An algorithm for computing the mixed radix FFT', IEEE Trans. Audio Electroacoust., Vol. AU-17, pp.93-103, 1969.

17. WINNING D.J., EL-SHIRBEENY E.H.T., THOMPSON E.C. and MURRAY-SMITH D.J. : 'Sensitivity method for online optimisation of a synchronous-generator excitation controller', Proc. IEE, Vol. 124, No. 7, July 1977.

18. EL-SHIRBEENY E.H.T. : 'Technique for Direct Evaluation of Parameter Sensitivity Functions', Electronics Letters, 12th December, 1974, Vol. 10, No. 25/26.

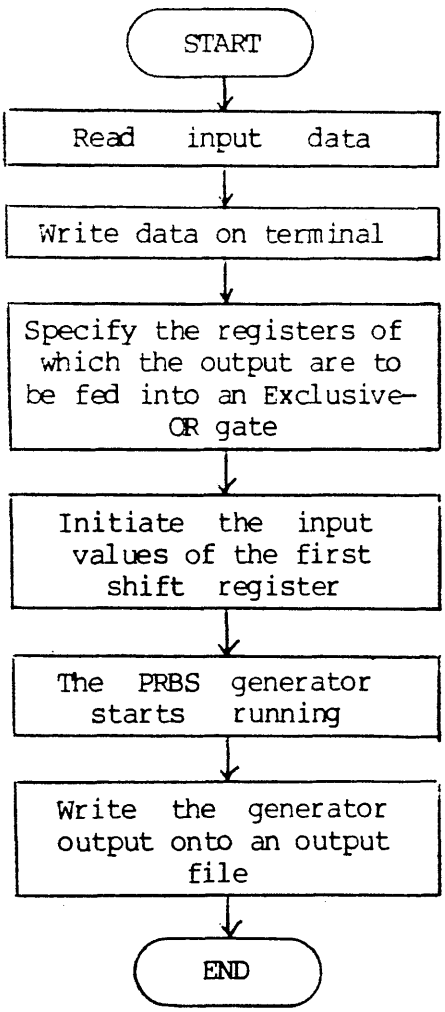
19. MURRAY-SMITH D.J. : 'Investigation of Methods for the Direct Assessment of Parameter Sensitivity in Linear Closed-Loop Control Systems', Preprints IMACS 11th World Congress, Vol. 4, pp.41-44, Norwegian Soc. Auto. Control, 1985.

20. ÅSTRÖM K.J. and EYKHOFF P. : 'System identification - a survey', Automatica, vol. 7, pp.123-162.

APPENDIX A

A Digital Simulation of a Pseudo-random-binary-sequence Generator

A computer program was written to generate a pseudo-random-binary-sequence. A flow diagram for the computer program is shown below:



A listing of the program is shown in the following pages.

```
THIS PROGRAM GENERATES A PSEUDO-RANDOM BINARY SEQUENCE (PRBS).
ARRAYS AND VARIABLES USED :
IOUTP(N) = OUTPUT OF THE NTH SHIFT REGISTER
INP(N) = INPUT OF THE NTH SHIFT REGISTER
INEX(1) AND INEX(2) ARE THE TWO INPUTS TO THE EXCLUSIVE-OR GATE
IOUTEX(N) = OUTPUT OF THE EXCLUSIVE-OR GATE
NST = NUMBER OF THE STAGES OF THE SHIFT REGISTER
NSP = NUMBER OF BITS FOR EACH BIT INTERVAL
NPER = NUMBER OF PERIODS OF PRBS
AMP = THE PEAK TO PEAK MAGNITUDE OF THE PRBS
NSEC = THE NUMBER OF BIT IN EACH PRBS SEQUENCE


    DIMENSION IOUTP(100)
    DIMENSION IOUTEX(2000)
    DIMENSION INP(100)
    DIMENSION INEX(2)
    DIMENSION IEXST(2)
KIN, KVDU, NIN, NOUT, NLST SPECIFY THE NUMBER CHANNEL USED
KIN = CHANNEL ONE FOR INPUTTING DATA FROM VDU
KVDU = CHANNEL TWO FOR OUTPUTTING DATA TO VDU
NIN = CHANNEL THREE FOR INPUTTING DATA FROM A FILE
NOUT = CHANNEL FOUR FOR OUTPUTTING DATA TO A FILE
NLST = CHANNEL SIX FOR OUTPUTTING DATA TO A FILE

    DATA KIN, KVDU, NIN, NOUT, NLST/ 1,2,3,4,6/

READ INPUT DATA

    WRITE (KVDU,5)
    READ(NIN,*)NST
    WRITE (KVDU,15)
    READ (NIN,*)NSP
    WRITE (KVDU,17)
    READ (NIN,*)NPER
    WRITE (KVDU,20)
    READ (NIN,*)AMP
    FORMAT (' TYPE IN THE NUMBER OF STAGES FOR THE PRBS GENERATOR'/)
    FORMAT (' TYPE IN THE NUMBER OF SAMPLES FOR EACH BIT INTERVAL'/)
    FORMAT (' TYPE IN THE NUMBER OF PERIODS'/)
    FORMAT (' TYPE IN THE PEAK-PEAK MAGNITUDE THE OUTPUT SIGNAL'/)

THE FOLLOWING SPECIFIES THE NUMBER OF THE REGISTER OUTPUTS
   WHICH ARE TO BE APPLIED TO THE EXCLUSIVE-OR GATE.

    IF (NST.NE.2) GOTO 110
    IEXST(1) = 2
    IEXST(2) = 1
    IF (NST.NE.3) GOTO 120
    IEXST(1) = 3
    IEXST(2) = 1
    IF (NST.NE.4) GOTO 130
    IEXST(1) = 4
    IEXST(2) = 3
    IF (NST.NE.5) GOTO 140
    IEXST(1) = 5
    IEXST(2) = 2
    IF (NST.NE.6) GOTO 150
    IEXST(1) = 6
```

```
      IEXST(2) = 1
      IF (NST.NE.7) GOTO 160
      IEXST(1) = 7
      IEXST(2) = 3
      IF (NST.NE.9) GOTO 170
      IEXST(1) = 9
      IEXST(2) = 4
      IF (NST.NE.10) GOTO 180
      IEXST(1) = 10
      IEXST(2) = 3
      IF (NST.NE.11) GOTO 190
      IEXST(1) = 11
      IEXST(2) = 2
      CONTINUE

INITIATE THE INPUT VALUES OF THE SHIFT REGISTERS AND
   THE OUTPUT VALUE  OF THE FIRST SHIFT REGISTER.
THIS IS TO AVOID THE OUTPUT OF ALL SHIFT REGISTERS TO
   TO BE ALL ZERO BECAUSE THE SYSTEM WILL BE STUCK IN THIS STATE.

      IOUTEX(1) = 0
      INP(1) = IOUTEX(1)
      DO 300 I = 2, NST
      INP(I) = 1
      CONTINUE

THE PRBS GENERATOR STARTS RUNNING

      NSEC = (2**NST) - 1
      DO 1000 NSECN = 1,NSEC
         DO 900 NSTNO = 1, NST
         IOUTP(NSTNO) = INP(NSTNO)
         IF (NSTNO.EQ.IEXST(1)) INEX(1) = IOUTP(NSTNO)
         IF (NSTNO.EQ.IEXST(2)) INEX(2) = IOUTP(NSTNO)
         CONTINUE

CARRY OUT THE LOGIC OF AN EXCLUSIVE-OR GATE

      IF (INEX(1).EQ.INEX(2)) IOUTEX(NSECN + 1) = 0
      IF (INEX(1).NE.INEX(2)) IOUTEX(NSECN + 1) = 1

CARRY OUT THE SHIFTING OF THE SHIFT REGISTERS

      INP(1) = IOUTEX(NSECN + 1)
      DO 950 I = 2, NST
      INP(I) = IOUTP(I - 1)
      CONTINUE
0     CONTINUE
      WRITE (2,25)
      FORMAT (' THE PSEUDO-RANDOM BINARY SEQUENCE IS '/)

THE PRBS GENERATED IS IN BINARY FORM WHICH ONLY TAKES
   THE VALUES 0 OR 1.
THE FOLLOWING CHANGE THE MAGNITUDE OF THE PRBS SO THAT
   IT TAKES THE VALUES (-AMP*2) OR  (AMP*2),
   i.e. IT HAS A PEAK TO PEAK VALUE OF AMP

      DO 3000 K = 1, NPER
      DO 2000 I = 1, NSEC
         DO 1500 J = 1, NSP
         IF (IOUTEX(I).EQ.1) FOUT=AMP/2.0
```

```
    IF (IOUTEX(I).EQ.Ø) FOUT=Ø.Ø-(AMP/2.Ø)
    WRITE(4,3Ø) FOUT
    FORMAT (G17.7)
    CONTINUE
CONTINUE
CONTINUE
END
```

APPENDIX B

A Digital Simulation of the Two Tank Flow Control System

The MIMESIS program was written to simulated the two tank flow control system. The program listing is shown below:

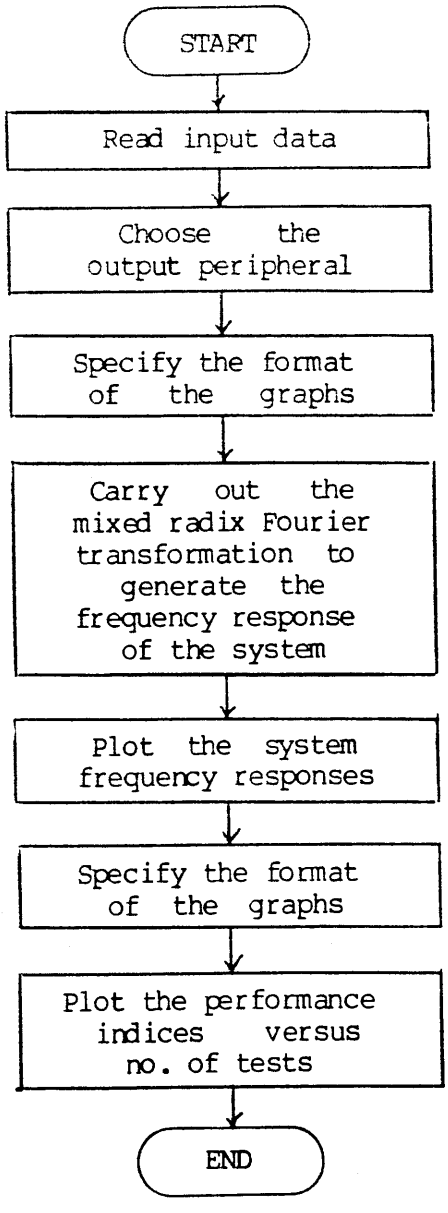
```
THIS IS A SIMULATION PROGRAM OF A TWO TANK  FLOW
   CONTROL SYSTEM.
THIS SYSTEM IS WRITTEN FROM A REFERENCE BOOK : PROCESS
   SYSTEMS ANALYSIS AND CONTROL, BY COUGHANOWR & KOPPEL,
   ON PAGE 95, PROBLEM 8.3.
THE BASIC SYSTEM CONSISTS OF TWO LIQUID TANKS : TANK 1 & 2.
THE OUTFLOW OF TANK 1 IS CONNECTED TO THE INFLOW OF TANK 2.
DATA : CROSS-SECTIONAL AREA OF TANK 1 (A1) = 10.0 SQ.FT
                                          = 0.9290304 SQ.METER
       CROSS-SECTIONAL AREA OF TANK 2 (A2) = 10.0 SQ.FT
                                          = 0.9290304 SQ.METER
       HYDRAUTIC RESISTENCE OF THE OUTFLOW PIPE OF TANK 1
                                          = 0.1 FT/CFM
                                          = 64.5834625 M/CUBE-M/S
       HYDRAUTIC RESISTENCE OF THE OUTFLOW PIPE OF TANK 2
                                          = 0.35 FT/CFM
                                          = 226.0421187 M/CUBE-M/S

PARAMETERS DEFINITION :
H1 = SMALL CHANGE OF FLUID HEAD OR HEIGHT(M) IN TANK 1
H2 = SMALL CHANGE OF FLUID HEAD(M) IN TANK 2
Q0 = SMALL CHANGE OF LIQUID INFLOW RATE(CUBE-M/S) OF TANK 1
Q1 = SMALL CHANGE OF LIQUID INFLOW RATE(CUBE-M/S) OF TANK 2
Q2 = SMALL CHANGE OF LIQUID OUTFLOW RATE(CUBE-M/S) OF TANK 2
G = FEEDBACK GAIN OF THE OUTPUT(Q2)
PRBS = PSEUDO RANDOM BINARY SEQUENCE
NCH = NUMBER OF CHANNELS OR VARIABLES IN THE LIST
      FOR THE TYPE STATEMENT
NSA = NUMBER OF SAMPLES
SAM = SAMPLING INTERVAL

  NCH = 7
  SAM = 3.9138943

INITIAL VALUES OF VARIABLES

  G = 1.0
  H1 = 0.0
  H2 = 0.0
  Q0 = 0.0
  Q1 = 0.0
  Q2 = 0.0
  A1 = 0.9290304
  A2 = 0.9290304
  R1 = 64.5834625
  R2 = 226.0421187
```

```
INPUT THE PSEUDO RANDOM BINARY SEQUENCE

  ACCEPT PRBS

INITIAL CONDITIONS

  X1Ø = Ø.Ø
  X2Ø = Ø.Ø

STEP SIZE AND NUMBER OF SAMPLES

  STP = 5ØØØ.Ø
  CINTERVAL (SAM)
  NSA = IFIX (STP/SAM)

INFORMATION WRITTEN TO OUTPUT FILE

  TYPE NCH
  TYPE NSA
  TYPE SAM

VALUES FOR INTEGRATION PARAMETERS AT DEFAULT VALUES
NEXT THREE LINES NEEDS NOT TO BE INCLUDED

  MINTERVAL (1.ØE-6)
  ABSERR (1.ØE-3)
  RELERR (1.ØE-3)

START THE SIMULATION PROCESS

  DYNAMIC
     DERIVATIVE
     X = QØ - Q1
     X1 = INTEG (X,X1Ø)
     H1 = X1/A1
     Q1 = H1 / R1
     Y = Q1 - Q2
     X2 = INTEG (Y,X2Ø)
     H2 = X2 / A2
     Q2 = H2 / R2
     QØ = PRBS - (Q2 * G)
     DERIVATIVE END
  TYPE T, PRBS, QØ, Q1, Q2, H1, H2
  ACCEPT PRBS
  IF (T-STP) 1Ø, 1Ø, 12
  DYNAMIC END
  STOP
  END
```

APPENDIX C

A Fortran Program for computing the Frequency Response of the Two Tank Flow Control System

A Fortran program was written to read the input and output signals of the two tank flow control system. It then uses the mixed radix Fourier algorithm to generate the frequency response of the system. A flow diagram of the program is shown below:



A listing of the program is shown in the following pages.

```
THIS PROGRAM READS IN DATA FROM INPUT FILE &CMMPOU5
   THE DATA INCLUDES THE INPUT AND OUTPUT SIGNALS OF THE
   TWO TANK FLOW CONTROL SYSTEM.
   THE INPUT SIGNAL IS A PSEUDO RANDOM BINARY SEQUENCE
   OF THE ORDER NINE.
THE FREQUENCY RESPONSE OF THE SYSTEM IS THEN CALCULATED.
A MAGNITUDE SPECTRUM AND A PHASE SPECTRUM ARE PLOTTED.
ARRAY DEFINITION:
   X1 = THE IMAGINARY PART OF THE NTH BIT OF THE SYSTEM
           INPUT SIGNAL
   X2 = THE IMAGINARY PART OF THE NTH BIT OF THE SYSTEM
           OUTPUT SIGNAL
   T = THE TIME INTERVAL OF THE SIGNAL
   F = THE FREQUENCY INTERVALS
   PRBS = THE INPUT PSEUDO-RANDOM-BINARY-SEQUENCE
   H1 = SMALL CHANGE OF FLUID HEAD OR HEIGHT(M) IN TANK 1
   H2 = SMALL CHANGE OF FLUID HEAD(M) IN TANK 2
   Q0 = SMALL CHANGE OF LIQUID INFLOW RATE(CUBE-M/S) OF TANK 1
   Q1 = SMALL CHANGE OF LIQUID INFLOW RATE(CUBE-M/S) OF TANK 2
   Q2 = SMALL CHANGE OF LIQUID OUTFLOW RATE(CUBE-M/S) OF TANK 2
   UMA = MAGNITUDE SPECTRUM OF THE INPUT SIGNAL
   UAR = PHASE SPECTRUM OF THE INPUT SIGNAL
   YMA = MAGNITUDE SPECTRUM OF THE OUTPUT SIGNAL
   YAR = PHASE SPECTRUM OF THE OUTPUT SIGNAL
   GAIN = MAGNITUDE SPECTRUM OF THE SYSTEM TRANSFER FUNCTION
   PHASE = PHASE SPECTRUM OF THE SYSTEM TRANSFER FUNCTION

  IMPLICIT DOUBLE PRECISION (D)
  DIMENSION Q0(511), H2(511), Q1(511)
  DIMENSION PRBS(511)
  DIMENSION Q2(511), H1(511)
  DIMENSION T(511), U(511), F(511), X1(511)
  DIMENSION YAR(511), YMA(511)
  DIMENSION X2(511)
  DIMENSION GAIN(511), PHASE(511)
  DIMENSION Y(511), UAR(511), UMA(511), A1(511), B1(511)

KIN, KVDU, NIN, NOUT and NLST SPECIFY THE CHANNEL NUMBER
   KIN = CHANNEL ONE FOR INPUTTING DATA FROM VDU
   KVDU = CHANNEL TWO FOR OUTPUTTING DATA TO VDU
   NIN = CHANNEL THREE FOR INPUTTING DATA FROM A FILE
   NOUT = CHANNEL FOUR FOR INPUTTING DATA ONTO A FILE
   NLST = CHANNEL SIX FOR OUTPUTTING DATA ONTO A FILE

  DATA KIN, KVDU, NIN, NOUT, NLST/1,2,3,4,6/

READ HEADER INFORMATION FROM MIMESIS OUTPUT FILE

  READ (NIN) NCH
  READ (NIN) NSA
  READ (NIN) SAM

N = NUMBER OF SAMPLES FOR EACH PRBS SEQUENCE

  N = 511
  AN = N

READ INPUT DATA
THE FIRST SEQUENCE OF INPUT DATA IS OMMITED
```

```
   BECAUSE THE SYSTEM RESPONSE IS NOT YET REACHED
   STEADY STATE.
THEREFORE THE INPUT DATA IS ONLY STORED IN A VARIBLE

  DO 22 I = 1, N
  READ (NIN) T(I), AA, AA, AA, AA, AA, AA
     DO 3Ø I = 1, N
     READ (NIN) AA, PRBS(I), QØ(I), Q1(I), Q2(I),
*               H1(I), H2(I)
     CONTINUE
  DO 4Ø I = 1, N
  U(I) = PRBS(I)
  Y(I) = Q2(I)
  CONTINUE

CALCULATE THE FREQUENCY INTERVALS ON THE FREQUENCY AXIS

  FRE = 1.Ø / (AN * SAM)
  DO 9Ø I = 1, N
  TI = I
  F(I) = TI * FRE

INITIATE THE IMAGINARY PART OF THE INPUT AND OUTPUT SIGNALS
   TO ZERO.

  X1(I) = Ø.Ø
  X2(I) = Ø.Ø
  CONTINUE

CARRY OUT FOURIER TRANSFORM ON THE INPUT SIGNAL.

  CALL FFT(U, X1, N, N, N, 1)
  DO 1ØØ I = 1, N
  A1(I) = U(I)
  B1(I) = -X1(I)
  CONTINUE
  DO 11Ø I = 1, N
  UAR(I) = ATAN2 ( B1(I), A1(I) ) * 18Ø.Ø / 3.141592654
  IF (UAR(I) .GT. 18Ø.Ø) GO TO 152
  IF (UAR(I) .LT. -18Ø.Ø) UAR(I) = UAR(I) + 36Ø.Ø
  GO TO 154
  UAR(I) = UAR(I) - 36Ø.Ø
  CONTINUE
  UAR(I) = A1(I)
  UMA(I) = SQRT ( B1(I)**2 + A1(I)**2 )
  UMA(I) = 2Ø.Ø * LOG1Ø( UMA(I) )
  CONTINUE

CARRY OUT FOURIER TRANFORM ON THE OUTPUT SIGNAL

  CALL FFT (Y, X2, N, N, N, 1)
  DO 158 I = 1, N
  A1(I) = Y(I)
  B1(I) = -X2(I)
  YAR(I) = ATAN2 ( B1(I), A1(I) ) * 18Ø.Ø / 3.141592654
  IF (YAR(I) .GT. 18Ø.Ø) GO TO 156
  IF (YAR(I) .LT. -18Ø.Ø) YAR(I) = YAR(I) + 36Ø.Ø
  GO TO 157
  YAR(I) = YAR(I) - 36Ø.Ø
  CONTINUE
  YMA(I) = SQRT ( B1(I)**2 + A1(I)**2 )
```

```
  YMA(I) = 20.0 * LOG10( YMA(I) )

CALCULATE THE MAGNITUDE AND PHASE CHARACTERISTICS OF THE SYSTEM
   TRANSFER FUNCTION.

  GAIN(I) = YMA(I) - UMA(I)
  PHASE(I) = YAR(I) - UAR(I)
  IF (PHASE(I) .GT. 180.0) GO TO 155
  IF (PHASE(I) .LT. -180.0) PHASE(I) = PHASE(I) + 360.0
  GO TO 159
  PHASE(I) = PHASE(I) - 360.0
  CONTINUE
  CONTINUE

THE FOLLOWING SPECFIES THE OUTPUT PERIPHERAL

  WRITE (KVDU, 120)
  FORMAT(' DO 1 FOR BENSON, 2 FOR SIGMA, 3 FOR 4014,')
  WRITE (KVDU, 130)
  FORMAT('     4 FOR 4010, 5 FOR HP7470')
  READ (KIN,*)K
  WRITE (KVDU, 140)
  FORMAT(' READY TO PLOT')
  IF (K.EQ.1) CALL B1302
  IF (K.EQ.1) GO TO 5
  IF (K.EQ.2) CALL S5600
  IF (K.EQ.3) CALL T4014
  IF (K.EQ.4) CALL T4010
  IF (K.EQ.5) GO TO 150
  GO TO 160
  CALL HP747
  CALL CHASWI(1)
  CONTINUE
  CALL PICCLE
  CONTINUE

PLOT THE SYSTEM FREQUENCY RESPONSE

  WRITE (KVDU, 230)
  FORMAT(' PLOT OF SYSTEM RESPONSE vs FREQUENCY RANGE OF INTEREST')
  CALL WIN
  CALL AXIPOS (1, 40.0, 180.0, 130.0, 1)
  CALL AXISCA (4, 10, 0.0001, 0.1, 1)
  CALL AXIDRA (2, 0, 1)
  CALL AXIPOS (1, 40.0, 20.0, 130.0, 1)
  CALL AXISCA (4, 10, 0.0001, 0.1, 1)
  CALL AXIDRA (2, 1, 1)
  CALL AXIPOS (1, 40.0, 20.0, 75.0, 2)
  CALL AXISCA (3, 6, -180.0, 0.0, 2)
  CALL AXIDRA (-1, 0, 2)
  CALL GRACUR (F, PHASE, 50)
  CALL AXIPOS (1, 170.0, 20.0, 75.0, 2)
  CALL AXIDRA (-1, 1, 2)
  CALL AXIPOS (1, 40.0, 95.0, 10.0, 2)
  CALL AXISCA (3, 1, 0.0, 1.0, 2)
  CALL AXIDRA (-1, 0, 2)
  CALL AXIPOS (1, 170.0, 95.0, 10.0, 2)
  CALL AXIDRA (-1, 0, 2)
  CALL AXIPOS (1, 170.0, 105.0, 75.0, 2)
  CALL AXISCA (3, 6, -60.0, 0.0, 2)
  CALL AXIDRA (-1, 0, 2)
```

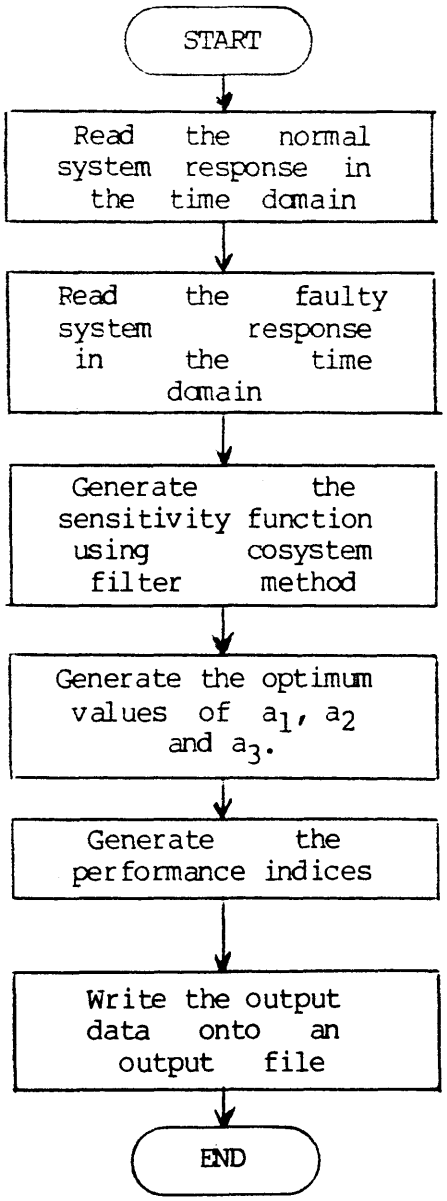
```
CALL AXIPOS (1, 40.0, 105.0, 75.0, 2)
CALL AXIDRA (-1, -1, 2)
CALL AXIPOS (1, 40.0, 105.0, 130.0, 1)
CALL GRACUR (F, GAIN, 50)
CALL CHAMOD
CALL DEVEND
STOP
END
```

A complete program listing of the subroutine FFT which implements the mixed radix Fourier transformation can be found in the Singleton's paper[16].

APPENDIX D

A program for generating the system sensitivity functions and carrying out the optimisation process

D.1 A program was written to generate the system sensitivity functions and carry out the optimisation process. The optimisation process was implemented by varying parameters a_1, a_2 and a_3. A flow diagram of the program is shown below:



The program listing is shown in the following pages.

```
THIS PROGAM READS IN DATA FROM INPUT FILES : &CMMPOU5 and &CMMPOU6
   &CMMPOU5 CONSISTS OF DATA OF SYSTEM RESPONSE BEFORE PARAMETER
   CHANGES
   &CMMPOU6 CONSISTS OF DATA OF SYSTEM RESPONSE AFTER PARAMETER
   CHANGES
THE INPUT SIGNAL IS AN UNIT IMPULSE FUNCTION
THE SENSITIVITY FUNCTIONS ARE CALCULATED AND THEN PLOTTED.
ARRAY DEFINITION :
   ADIFFE = THE DIFFERENCE OF TWO SYSTEM RESPONSES
   ADIFF = THE SQUARE OF THE DIFFERENCE OF TWO SYSTEM RESPONSES
   WMA = MAGNITUDE SPECTRUM OF THE SYSTEM RESPONSE
   U = INPUT SIGNAL
   YA = SYSTEM OUTPUT SIGNAL BEFORE PARAMETER CHANGES
   YB = SYSTEM OUTPUT SIGNAL AFTER PARAMETER CHANGES
   F = FREQUENCY INTERVALS ON THE FREQUENCY DOMAIN
   T = TIME INTERVALS
   UMA = MAGNITUDE SPECTRUM OF THE INPUT SIGNAL
   UAR = PHASE SPECTRUM OF THE OUTPUT SIGNAL
   YAMA = MAGNITUDE SPECTRUM OF THE OUTPUT SIGNAL BEFORE PARAMETER
          CHANGE
   YBMA = MAGNITUDE SPECTRUM OF THE OUTPUT SIGNAL AFTER PARAMETER
          CHANGE
   ZMA = MAGNITUDE SPECTRUM OF THE SENSTIVITY FUNCTION
   ZAR = PHASE SPECTRUM OF THE SENSITIVITY FUNCTION
   ZA1, ZA2 and ZA3 = SENSITIVITY FUNCTIONS
   Y, A, B, C, DA, DB, DC and DWKSPC ARE WORKING ARRAYS

  IMPLICIT DOUBLE PRECISION (D)
  DIMENSION ADIFFE(511)
  DIMENSION ADIFF(511)
  DIMENSION A(4,4), B(6,1),C(6,1),DA(4,4),DB(6,1),DC(6,1)
  DIMENSION DWKSPC(1000)
  DIMENSION WMA(511)
  DIMENSION U(511), YA(511), YB(511)
  DIMENSION Y(511), T(511)
  DIMENSION ZA1(511), ZA2(511), ZA3(511)
  DIMENSION UMA(511), YAMA(511), YBMA(511), UAR(511)
  DIMENSION ZMA(511), ZAR(511)
  DIMENSION F(511)

KIN, KVDU, NIN, NOUT and NLST SPECIFY THE CHANNEL NUMBER
   KIN = CHANNEL ONE FOR INPUTTING DATA FROM VDU
   KVDU = CHANNEL TWO FOR OUTPUTTING DATA TO VDU
   NIN = CHANNEL THREE FOR INPUTTING DATA FROM A FILE
   NOUT = CHANNEL FOUR FOR OUTPUTTING DATA ONTO A FILE
   NLST = CHANNEL SIX FOR OUTPUTTING DATA ONTO A FILE

  DATA KIN, KVDU, NIN, NOUT, NWITH,NLST/1,2,3,4,5,6/

READ HEADER INFORMATION FROM MIMESIS OUTPUT FILE
THE FILE READ FROM CHANNEL THREE(NIN) CONSISTS OF DATA OF SYSTEM RESPON:
   BEFORE PARAMETER CHANGES.


  READ (NIN) NCH
  READ (NIN) NSA
  READ (NIN) SAM
  READ (NIN) R2
```

```
  READ (NIN) CA11
  READ (NIN) CA22
  READ (NIN) CA33
  READ (NIN) ADCA1
  READ (NIN) ADCA2
  READ (NIN) ADCA3

N = NUMBER OF SAMPLES FOR EACH PRBS SIGNAL

  N = 511

READ IN THE FIRST SEQUENCE

  DO 10 I = 1, N
  READ (NIN) T(I),AAA,AA,AA,AA,AA,AA,AA,AA,AA
  CONTINUE

READ HEADER INFORMATION FROM MIMESIS OUTPUT FILE
DATA READ FROM CHANNEL FIVE(NWITH) CONSISTS OF SYSTEM RESPONSE
   AFTER PARAMETER CHANGES

  READ (NWITH) NCH
  READ (NWITH) NSA
  READ (NWITH) SAM
  READ (NWITH) R2
  WRITE (KVDU,*) NCH, NSA, SAM, R2
  READ (NWITH) CA1
  READ (NWITH) CA2
  READ (NWITH) CA3
  READ (NWITH) ADCA1
  READ (NWITH) ADCA2
  READ (NWITH) ADCA3

INITIATE THE PERFORMANCE INDEX (PFMCID)

  PFMCID = 0.0
  AN = N

WRITE DATA TO OUTPUT FILE

  WRITE (NLST)NCH
  WRITE (NLST)NSA
  WRITE (NLST)SAM
  WRITE (NLST)R2
  WRITE (NLST)CA1
  WRITE (NLST)CA2
  WRITE (NLST)CA3
  WRITE (NLST)ADCA1
  WRITE (NLST)ADCA2
  WRITE (NLST)ADCA3

READ THE FIRST SEQUENCE OF DATA OF SYSTEM RESPONSE AFTER
   PARAMETER CHANGE

   DO 20 I = 1, N
  READ (NWITH) AA,AA,AA,AA,AA,AA,AA,AA,AA,AA
  CONTINUE

READ THE NEXT SEQUENCE FROM BOTH CHANNELS
AND WRITE DATA TO OUTPUT FILE
```

```
      DO 30 I = 1, N
      READ (NIN)AA,U(I),AA,AA,YA(I),AA,AA,AA,AA,AA
      READ (NWITH) AA,AA,AA,AA,YB(I),AA,AA,ZA1(I),ZA2(I),ZA3(I)
      WRITE (NLST)U(I),YA(I)
      WRITE (NLST)YB(I),ZA1(I),ZA2(I),ZA3(I)
      WRITE (KVDU,*)I,U(I),YA(I),YB(I)
      CONTINUE

CALCULATE THE FREQUENCY INTERVALS ON THE FREQUENCY AXIS

  FRE = 1.0 / (AN*SAM)
  DO 40 I = 1, N
  TI = I
  F(I) = TI * FRE
  CONTINUE

CARRY FOURIER TRANSFORMATION

  CALL FFTRAN (YA, YAMA, UAR, N)
  CALL FFTRAN (YB, YBMA, UAR, N)
  CALL FFTRAN (U, UMA, UAR, N)

CALCULATE THE SENSITIVITY FUNCTION

  DO 150 I = 1, 3
  IF (I.EQ.1) CALL FFTRAN (ZA1, ZMA, ZAR, N)
  IF (I.EQ.2) CALL FFTRAN (ZA2, ZMA, ZAR, N)
  IF (I.EQ.3) CALL FFTRAN (ZA3, ZMA, ZAR, N)
  DO 160 J = 1, N
  WMA(J) = YBMA(J) - UMA(J)
  U(J) = ZMA(J) - UMA(J)
  Y(J) = ZAR(J) - UAR(J)
  U(J) = 10.0 ** ( U(J) / 20.0 )
  WMA(J) = 10.0 ** ( WMA(J) / 20.0 )
  ZMA(J) = U(J)*WMA(J)*(COS(Y(J)*3.141592654/180.0))
  IF (I.EQ.1) ZA1(J) = ZMA(J)
  IF (I.EQ.2) ZA2(J) = ZMA(J)
  IF (I.EQ.3) ZA3(J) = ZMA(J)
  CONTINUE
  CONTINUE

THE MAGNITUDE IS IN DECIBEL UNIT
THE FOLLOWING CHANGE dB UNIT BACK TO ACTUAL VALUE
THE PERFORMANCE INDEX IS ALSO CALCULATED

  DO 50 I = 1, N
  YAMA(I) = YAMA(I) - UMA(I)
  YBMA(I) = YBMA(I) - UMA(I)
  YAMA(I) = 10.0 ** ( YAMA(I)/20.0 )
  YBMA(I) = 10.0 ** (YBMA(I)/20.0)
  ADIFF(I) = ( YAMA(I) - YBMA(I) ) * ( YAMA(I) - YBMA(I) )
  ADIFFE(I) =  ( YAMA(I) - YBMA(I) )
  WRITE (KVDU,*)I, YARE(I),YAIM(I),YBRE(I), YBIM(I)
  WRITE (KVDU,*)1,YAMA(I),YBMA(I),ADIFFE(I)
  PFMCID = PFMCID + ADIFF(I)
  CONTINUE
  PFMCID = PFMCID * FRE

THE FOLLOWING IS TO SOLVE THE SIMULTANEOUS EQUATIONS

  CALL SUMPDT (ZA1, ZA1, SUMA, 200)
```

```
A(1,1) = SUMA
CALL SUMPDT (ZA1, ZA2, SUMA, 200)
A(1,2) = SUMA
CALL SUMPDT (ZA1, ZA3, SUMA, 200)
A(1,3) = SUMA
CALL SUMPDT (ZA2, ZA1, SUMA, 200)
A(2,1) = SUMA
CALL SUMPDT (ZA2, ZA2, SUMA, 200)
A(2,2) = SUMA
CALL SUMPDT (ZA2, ZA3, SUMA, 200)
A(2,3) = SUMA
CALL SUMPDT (ZA3, ZA1, SUMA, 200)
A(3,1) = SUMA
CALL SUMPDT (ZA3, ZA2, SUMA, 200)
A(3,2) = SUMA
CALL SUMPDT (ZA3, ZA3, SUMA, 200)
A(3,3) = SUMA
CALL SUMPDT (ADIFFE, ZA1, SUMA, 200)
B(1,1) = SUMA
CALL SUMPDT (ADIFFE, ZA2, SUMA, 200)
B(2,1) = SUMA
CALL SUMPDT (ADIFFE, ZA3, SUMA, 200)
B(3,1) = SUMA

USE A SUBROUTINE IN THE NAG LIBRARY TO SOLVE THE SIMULTANEOUS EQUATION

NA = 3
MA = 1
IA = 4
IB = 6
IFAIL = 0
IC = 6
DO 170 I = 1, 3
DO 180 J = 1, 3
DA(I,J) = A(I,J)
CONTINUE
DB(I,1) = B(I,1)
CONTINUE
CALL F04AAF (DA,IA,DB,IB,NA,MA,DC,IC,DWKSPC,IFAIL)
DO 190 I = 1, 3
C(I,1) = DC(I,1)
CONTINUE
ADCA1 = ADCA1 + C(1,1)
ADCA2 = ADCA2 + C(2,1)
ADCA3 = ADCA3 + C(3,1)
CA1 = CA11 + ADCA1
CA2 = CA22 + ADCA2
CA3 = CA33 + ADCA3
WRITE (KVDU,*) PFMCID
WRITE (NLST) PFMCID
CALL WRIOUT (R2,CA1,CA2,CA3,ADCA1,ADCA2,ADCA3)
STOP
END
```

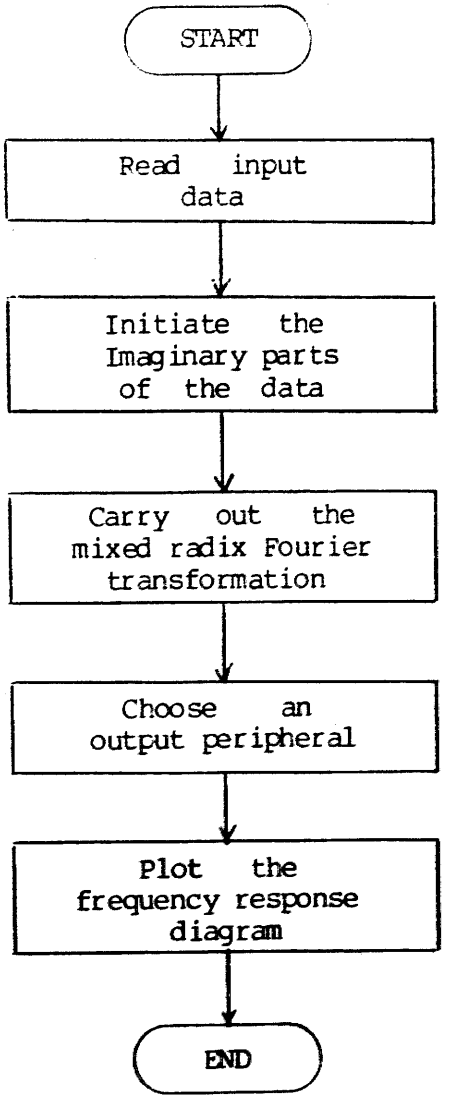
A program listing of the subroutine SUMPDT is shown below. This routine multiplies X(I) with Y(I) and stores the result in SUM(I), for I = 1 to N.

```
      SUBROUTINE SUMPDT (X, Y, SUM, N)
      DIMENSION X(511), Y(511)
      DATA KIN,KVDU,NIN,NOUT,NWITH,NLST/1,2,3,4,5,6/
      SUM = Ø.Ø
      DO 1Ø I = 1, N
      SUM = SUM + X(I) * Y(I)
1Ø    CONTINUE
      RETURN
      END
```

D.2 Program for plotting graphs of the system freqency response before and after the optimisation process has been carried out.

The program, as shown in appendix D.1, was implemented repeatedly for several times. Each time after the program has been implemented, the system input and output signals and a performance index were produced.

A program was written to generate a number of system frequency responses based upon several sets of the time domain signals. It would then plot a graph of system frequency responses and a graph of system performance indices versus number of tests. A flow diagram of this program is shown below:



A listing of the program is shown in the following pages.

```
C
C  THIS PROGRAM PLOTS FOUR FREQUENCY RESPONSES.
C     EACH FREQUENCY RESPONSE HAS UNDERTAKEN A CORRECTION
C     USING SENSITIVITY FUNCTIONS
C  ARRAYS USED :
C     U = SYSTEM INPUT SYSTEM
C     UMA = MAGNITUDE SPECTRUM OF THE INPUT SIGNAL
C     UAR = PHASE SPECTRUM OF THE INPUT SIGNAL
C     YA = OUTPUT SIGNAL OF THE SYSTEM BEFORE PARAMETER CHANGES
C     YB = OUTPUT SIGNAL OF THE SYSTEM AFTER PARAMETER CHANGES
C     YAMA = MAGNITUDE SPECTRUM OF YA
C     YBMA = MAGNITUDE SPECTRUM OF YB
C     YAR = PHASE SPECTRUM OF THE OUTPUT SIGNAL
C     TEST = NUMBER OF TEST UNDERTAKEN
C     F = FREQUENCY INTERVALS ALONG THE X-AXIS
C     PFMCID = PERFORMANCE INDEX
C     WORK = WORKING ARRAY
C
      IMPLICIT DOUBLE PRECISION (D)
      DIMENSION U(511), UMA(511), UAR(511)
      DIMENSION YA(511), YB(511), YAMA(511), YBMA(511)
      DIMENSION YAR(511), TEST(10), WORK(10), F(511)
      DIMENSION PFMCID(3,10)
C
C  KIN, KVDU, NIN, NOUT, NWITH and NLST SPECIFY THE CHANNEL NUMBER
C     KIN = CHANNEL ONE FOR INPUTTING DATA FROM VDU
C     KVDU = CHANNEL TWO FOR OUTPUTTING DATA TO VDU
C     NIN = CHANNEL THREE FOR INPUTTING DATA FROM A FILE
C     NOUT = CHANNEL FOUR FOR OUTPUTTING DATA TO A FILE
C     NWITH = CHANNEL FIVE FOR INPUTTING DATA FROM A FILE
C     NLST = CHANNEL SIX FOR OUTPUTTING DATA TO A FILE
C
      DATA KIN,KVDU,NIN,NOUT,NWITH,NLST/1,2,3,4,5,6/
C
C   N = NUMBER OF SAMPLES FOR EACH DISCRETE SEQUENCE
C
      N = 511
      AN = N
C
C   THE FOLLOWING IS TO SPECIFY THE OUTPUT PERIPHERAL
C
      WRITE (KVDU,10)
10    FORMAT(' DO 1 FOR BENSON, 2 FOR SUGMA, 3 FOR 4014,')
      WRITE (KVDU,20)
20    FORMAT ('   4 FOR 4010, 5 FOR HP7470')
      READ (KIN,*) K
      WRITE (KVDU,30)
30    FORMAT (' READY TO PLOT')
      IF (K.EQ.1) CALL B1302
      IF (K.EQ.1) GO TO 5
      IF (K.EQ.2) CALL S5600
      IF (K.EQ.3) CALL T4014
      IF (K.EQ.4) CALL T4010
      IF (K.EQ.5) GO TO 40
      GO TO 50
40    CALL HP747
      CALL CHASWI(1)
50    CONTINUE
      CALL PICCLE
5     CONTINUE
```

```
      DO 6Ø I = 1, 3
      CALL AXIPOS (1, 3Ø.Ø, 25.Ø, 13Ø.Ø, 1)
      CALL AXIPOS (1, 3Ø.Ø, 25.Ø, 12Ø.Ø, 2)
      CALL AXISCA (4, 4, Ø.ØØØ1, Ø.1, 1)
      CALL AXISCA (3, 4, Ø.Ø, Ø.4, 2)
      CALL AXIDRA (2, 1, 1)
      CALL AXIDRA (-1, -1, 2)
C
C   READ IN INPUT DATA
C     READ THE HEADER INFORMATION, INPUT AND OUTPUT SIGNAL,
C      FILTERED ACTUATION SIGNALS
C
C
         DO 7Ø J = 1, 1Ø
         READ(NIN) NCH
         READ (NIN) NSA
         READ (NIN) SAM
         READ (NIN) R2
         READ (NIN) CA1
         READ (NIN) CA2
         READ (NIN) CA3
         READ (NIN) ADCA1
         READ (NIN) ADCA2
         READ (NIN) ADCA3
            DO 8Ø K = 1, N
            READ (NIN) U(K), YA(K)
            READ (NIN) YB(K), ZA1, ZA2, ZA3
8Ø          CONTINUE
C
C   INPUT THE PERFORMANCE INDEX
C
         READ (NIN) PFMCID(I,J)
C
C   CALCULATED THE FREQUENCY INTERVAL ON THE X-AXIS
C
         FRE = 1.Ø / (AN * SAM)
            DO 9Ø K = 1, N
            TI = K
            F(K) = TI * FRE
9Ø          CONTINUE
C
C   CARRY OUT FOURIER TRANSFORMATION
C
         CALL FFTRAN (YA, YAMA, YAR, N)
         CALL FFTRAN (YB, YBMA, YAR, N)
         CALL FFTRAN (U, UMA, UAR, N)
C
C   OBTAIN THE MAGNITUDE RESPONSE OF THE SYSTEM
C   CHANGE THE DECIBEL VALUES TO ACTUAL VALUES
C
            DO 1ØØ K = 1, N
            YAMA(K) = YAMA(K) - UMA(K)
            YBMA(K) = YBMA(K) - UMA(K)
            YAMA(K) = 1Ø.Ø ** ( YAMA(K) / 2Ø.Ø )
            YBMA(K) = 1Ø.Ø ** ( YBMA(K) / 2Ø.Ø )
1ØØ         CONTINUE
C
C   PLOT THE FREQUENCY RESPONSE
C
         IF (J.GT.4) GO TO 13Ø
         IF (J.EQ.1) GO TO 11Ø
```

```
      GO TO 12Ø
Ø     CALL GRACUR (F, YAMA, 2ØØ)
      CALL CHAMOD
Ø     IF (J.EQ.1) CALL BROKEN(1)
      IF (J.EQ.2) CALL BROKEN(3)
      IF (J.EQ.3) CALL BROKEN(2)
      IF (J.EQ.4) CALL BROKEN(5)
      CALL GRACUR (F, YBMA, 2ØØ)
      CALL CHAMOD
      CALL BROKEN(Ø)
1Ø    CONTINUE
1   CONTINUE
    PAUSE
    CALL PICCLE
1   CONTINUE

  PLOT THE PERFORMANCE INDEX VERSUS THE NUMBER OF TESTS

    CALL BROKEN(Ø)
    CALL AXIPOS (1,4Ø.Ø, 2Ø.Ø, 13Ø.Ø, 1)
    CALL AXIPOS (1, 4Ø.Ø ,2Ø.Ø, 15Ø.Ø, 2)
    CALL AXISCA (3, 1Ø, Ø.Ø, 1Ø.Ø, 1)
    CALL AXISCA (3, 5, Ø.Ø, 5.ØE-5, 2)
    CALL AXIDRA (Ø, 1, 1)
    CALL AXIDRA (-1, -1, 2)
    DO 14Ø I = 1, 3
       DO 15Ø J = 1, 1Ø
       AJ = J
       TEST(J) = AJ
       WORK(J) = PFMCID(I,J)
5Ø     CONTINUE
    CALL GRAPOL (TEST, WORK, 1Ø)
    IF (I.EQ.1) NSYM = 5
    IF (I.EQ.2) NSYM = 1
    IF (I.EQ.3) NSYM = 7
    CALL GRASYM (TEST, WORK, 1Ø, NSYM, Ø)
4Ø  CONTINUE
    CALL DEVEND
    STOP
    END
```

A program listing of the subroutine FFTRAN is shown below:

```
      SUBROUTINE FFTRAN (X, UMA, UAR, N)
C   THIS SUBROUTINE PERFORMS FOURIER TRANSFORMATION ON THE
C        DATA X(I) AND PRODUCES THE MAGNITUDE SPECTRUM UMA(I)
C        AND THE PHASE SPECTRUM UAR(I).
C   THE ARRAYS X1(N) AND Y1(N) ARE WORKING ARRAYS
C
      DIMENSION X(1023), UMA(1023), UAR(1023)
      DIMENSION X1(1023), Y1(1023)
      DO 10 I = 1, N
      X1(I) = X(I)
      Y1(I) = 0.0
10    CONTINUE
      CALL FFT (X1, Y1, N, N, N, 1)
      DO 20 I = 1, N
      Y1(I) = - Y1(I)
20    CONTINUE
      DO 30 I = 1, N
      UAR(I) = ATAN2 ( Y1(I), X1(I) ) * 180.0 / 3.141592654
      IF ( UAR(I) .GT. 180.0 ) GO TO 40
      IF ( UAR(I) .LT. -180.0 ) UAR(I) = UAR(I) + 360.0
      GO TO 60
40    UAR(I) = UAR(I) - 360.0
60    CONTINUE
      UMA(I) = SQRT ( X1(I)**2 + Y1(I)**2 )
      UMA(I) = 20.0 * LOG10 (UMA(I))
30    CONTINUE
      RETURN
      END
```

A complete program listing of the subroutine FFT which implements the mixed radix Fourier transformation can be found in the Singleton's paper[16].