FINAL REPORT:
SYNTHESIS OF PHTHALOCYANINE DOPED SOL-GEL MATERIALS
(JPL CONTRACT 959234)

PRINCIPAL INVESTIGATOR: BRUCE DUNN
DEPARTMENT OF MATERIALS SCIENCE AND ENGINEERING
UNIVERSITY OF CALIFORNIA, LOS ANGELES

The synthesis of sol-gel silica materials doped with three different types of metallophthalocyanines has been studied. Homogeneous materials of good optical quality were prepared and the first optical limiting measurements of dyes in sol-gel hosts were carried out. The properties of these solid state limiters are similar to limiters based on phthalocyanine (Pc) in solution.

Sol-gel silica materials containing copper, tin and germanium phthalocyanines were investigated. The initial step in all cases was to prepare silica sols by the sonogel method using tetramethoxy silane (TMOS), HCl and distilled water. Thereafter, the synthesis depended upon the specific Pc and its solubility characteristics. Copper phthalocyanine tetrasulfonic acid tetra sodium salt (CuPc4S) is soluble in water and various doping levels ($1 \times 10^{-4}M$ to $1 \times 10^{-5}M$) were added to the sol. The group IV Pc's, $(SnPc(OSi(n-hexyl)_3)_2$ and $GePc(OSi(n-hexyl)_3)_2$, are insoluble in water and the process was changed accordingly. In these cases, the compounds were dissolved in THF and then added to the sol. The Pc concentration in the sol was $2 \times 10^{-5}M$. The samples were then aged and dried in the standard method of making xerogel monoliths.

Comparative nanosecond optical limiting experiments were performed on silica xerogels that were doped with the different metallophthalocyanines. The ratio of the net excited state absorption cross section (σ_e) to the ground state cross section (σ_g) is an important figure of merit that is used to characterize these materials. By this standard the SnPc sample exhibits the best limiting for the Pc doped sol-gel materials. Its cross section ratio of 19 compares favorably with the value of 22 that was measured in toluene. The GePc materials appear to not be as useful as those containing SnPc. The GePc doped solids exhibit a higher onset energy (2.5 mJ) and lower cross section ratio, 7. The CuPc4S sol-gel material has a still lower cross section ratio, 4, however, the tetrasulfonate groups make the dye soluble in water which greatly facilitates its incorporation into the sol-gel matrix.

The nonlinear transmission of CuPc4S in a pH 2 buffer solution and in a silica xerogel were compared. It is evident that the CuPc4S preserves its optical limiting behavior in the sol-gel matrix, indicating that the fundamental excited state absorption process is essentially the same for a molecule in solution or in the solid state. Although the spectroscopic details of energy level lifetimes are unknown, the significance is that passive optical limiting has been achieved in the solid state via incorporation of a dye into an inorganic host. The only compromise occurs at the extremely high energy regime where photobleaching is observed. This is a result of the limited mobility of the dye molecules in the solid silica host relative to a liquid host. The effects of photodegradation in the xerogel are additive, whereas the solution provides a supply of fresh molecules that are free to enter the active volume between pulses.

In this program we supplied over 20 samples to JPL. The work was presented at the SPIE meeting in San Diego (July, 1992) and the paper was published in the conference proceedings:

P.D. Fuqua, K. Mansour, D. Alvarez, Jr. S.R. Marder, J.W. Perry and
B. Dunn, "Synthesis and Nonlinear Optical Properties of Sol-Gel Materials Containing Phthalocyanines," Proc. Sol-Gel Optics II, Vol. 1758, ed. by J.D. Mackenzie (SPIE, Bellingham, WA, 1992) pp. 499-506.

UCLA
School
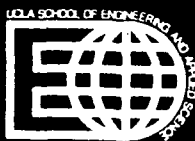# of
Engineering
and
Applied
Science

Final Report:

Part I: System Identification Using Gram-Schmidt Orthonormalization

Part II: Disturbance Rejection Using H_2 and H_∞ Control Design

Professor D. L. Mingori and K. F. Zimmermann
Mechanical, Aerospace, and Nuclear Eng. Dept.
University of California, Los Angeles
Los Angeles, CA 90024

Final Report[1]: Parts I & II
Part I: System Identification Using Gram–Schmidt Orthonormalization
Part II: Disturbance Rejection Using H_2 and H_∞ Control Design

Report No. 959234

D.L. Mingori
K.F. Zimmermann

Mechanical, Aerospace and Nuclear Engineering Department
University of California, Los Angeles, CA 90024

October 1, 1993

Contents

Introduction and Overview

This report contains the results of research supported by the Jet Propulsion Laboratory (JPL) contract no. 959234, conducted by Prof. D. L. Mingori and K. F. Zimmermann. The report is in two parts. Part one is titled "System Identification Using Gram-Schmidt Orthonormalization", and is concerned with the identification of linear, time-invariant systems, such as large flexible space structures. Part two is titled "Disturbance Rejection Using H_2 and H_∞ Control Design", and is concerned with the design of time-invariant controllers for narrow and wide band disturbance rejection. During the period covered by this contract, the focus of the research was expanded beyond the original focus of system identification to include the subject of disturbance rejection. This change was initiated with the approval and support of the JPL Technical Manager, Dr. Mark Milman.

Part I

As stated above, part one of the report is titled "System Identification Using Gram-Schmidt Orthonormalization." System identification is the fitting of a chosen model structure to some data from the system. Some common models for linear, time-invariant systems are the Auto-Regressive Moving-Average (ARMA) model, the Auto-Regressive (AR) model and the Moving-Average (MA) model. This report focuses on these model types.

In picking a model, the basis for the model must also be chosen. Often the basis that is chosen is not orthogonal. When the basis is not orthogonal, the model may be more sensitive to errors, and hence less "robust". Part one of this report contains a general procedure to orthonormalize any basis when fitting a model to frequency domain data.

The examples show that system identification using Gram–Schmidt or-

thonormalization has a great deal of potential. However, more work should be done in order to fully explore the usefulness of this procedure.

Part II

Part two of this report is titled "Disturbance Rejection Using H_2 and H_∞ Control Design." To begin this section one needs to have a model of the system. This model could be found using the results of part one of this report. Then using H_2 and H_∞ control design techniques one can find a time-invariant controller that satisfies the design criteria. This report shows examples of both wide and narrow band disturbance rejection controllers.

In this report we also explore the benefits of using H_2 and H_∞ control design techniques for multiple feedback control. It was found that for a given level of performance, single feedback controllers have higher gain than multiple feedback controllers. The lower gain of the multiple feedback controllers make them less sensitive to noise, and hence more "robust".

4

Part I

Chapter 1

System Identification Using Gram–Schmidt Orthonormalization

Abstract

Linear time–invariant plants are often modeled using Auto–Regressive Moving–Average (ARMA) models or Auto–Regressive (AR) models. Typically, these models are formed using basis functions corresponding to the z, s or δ operator raised to powers. For example, a function $A(p)$ could be expressed in terms of a truncated power series:

$$A(p) = \sum_{k=0}^{n} a_k p^k$$

where p represents either z, s or δ operator, and the a_k's are scalars. These bases may not be orthogonal. This is particularly apparent for s and δ. By replacing the basis $\{1, p, \ldots, p^n\}$ by an orthogonal basis, significant improvement can be achieved in the identification of a system. Presented in this report is a method of system identification using orthogonal bases found by Gram–Schmidt orthonormalization.

6

1.1 Introduction

System Identification is the process of developing mathematical models using data from a system. The mathematical models are formed using combinations of functions. Two of the popular model types for linear time–invariant systems are the Auto–Regressive Moving–Average (ARMA) model and the Auto–Regressive (AR) model. A more complete discussion of these model types can be found in [8] chapter 4. This report focuses on using Gram–Schmidt orthonormalization in conjunction with these model types.

Begin with the system,

$$y(p) = G(p)u(p) + H(p)e(p) \tag{1.1}$$

where $y(p)$, $u(p)$ and $e(p)$ are the output, input and the noise respectively; p represents either the z, s or δ operator; $G(p)$ and $H(p)$ are the plant and noise transfer functions. Assume that $y(p)$ and $u(p)$ are known, while $G(p)$, $H(p)$ and $e(p)$ are unknown. Consider a model in the form:

$$y(p) = \hat{G}(p,\theta)u(p) + \hat{H}(p,\theta)\hat{e}(p,\theta) \tag{1.2}$$

where $\hat{G}(p,\theta)$, $\hat{H}(p,\theta)$ and $\hat{e}(p,\theta)$ are estimates of $G(p)$, $H(p)$ and $e(p)$, and θ is a vector containing the parameters of the model. Let $\hat{e}(p,\theta) \triangleq y(p) - \hat{y}(p,\theta)$, and divide 1.2 through by $\hat{H}(p,\theta)$ to produce,

$$\hat{y}(p,\theta) = \hat{H}(p,\theta)^{-1}\hat{G}(p,\theta)u(p) + (1 - \hat{H}(p,\theta)^{-1})y(p). \tag{1.3}$$

Define $B(p,\theta) \triangleq \hat{H}(p,\theta)^{-1}\hat{G}(p,\theta)$, and $A(p,\theta) \triangleq \hat{H}(p,\theta)^{-1}$. Substitution using the definitions gives the following model,

$$\hat{y}(p,\theta) = B(p,\theta)u(p) + (1 - A(p,\theta)^{-1})y(p) \tag{1.4}$$

where the transfer functions $A(p,\theta)$ and $B(p,\theta)$ can be described by the following relationships,

$$A(p,\theta) = 1 + \sum_{k=1}^{n_a} a_k p^k \tag{1.5}$$

$$B(p,\theta) = \sum_{k=0}^{n_b} b_k p^k. \tag{1.6}$$

For an ARMA model, $n_a \geq n_b \geq 0$, and for an AR model, $n_a \geq 0$ and $B(p, \boldsymbol{\theta}) \equiv 0$.

Notice that in the frequency domain p (i.e. z, s or δ) is a function of the frequency ω, namely

$$z = e^{j\omega T} \qquad (1.7)$$

$$s = j\omega \qquad (1.8)$$

$$\delta = \frac{e^{j\omega T} - 1}{T}. \qquad (1.9)$$

From here on we will assume that p is a function of the frequency ω ($p \equiv p(\omega)$).

Notice that the transfer functions $A(p, \boldsymbol{\theta})$ and $B(p, \boldsymbol{\theta})$ are represented by a series of functions. These functions, $\{p^0, p^1, \ldots, p^n\}$, form a basis for the series. Instead of using p to powers to form the basis of the series, a sequence of orthogonal functions could be used instead, such that

$$A(p, \boldsymbol{\theta}) = \sum_{k=0}^{n_\alpha} \alpha_k \tilde{\phi}_k(\omega) \qquad (1.10)$$

$$B(p, \boldsymbol{\theta}) = \sum_{k=0}^{n_\beta} \beta_k \tilde{\phi}_k(\omega) \qquad (1.11)$$

where $\tilde{\phi}_k(\omega)$ is an orthogonal function. An orthogonal basis is considered to be a good choice for a basis, because it is typically less sensitive to errors, or more "robust". In this report the terms "function" and "vector" are used somewhat interchangeably, since a discrete function (a function evaluated at several values of its argument) can be used to form a vector.

This report discusses the properties of orthogonal functions, a method of generating orthogonal functions via Gram–Schmidt orthonormalization, and a method for using them in ARMA and AR models for the purpose of system identification.

1.2 Theory

1.2.1 Notation

p Represents either the z, δ or s operator.

z The Fourier series operator.

δ The Delta transform operator.

s The Laplace transform operator.

j The imaginary number, $j = \sqrt{-1}$.

ω Frequency in rad/sec.

T The sampling interval.

θ A vector containing the parameters of the model.

$w(x)$ A weighting function.

$w(\mathbf{x})$ A discrete weighting function (i.e. a vector containing the weights).

$W(\mathbf{x})$ A diagonal matrix containing the vector $w(\mathbf{x})$ along the main diagonal.

$\hat{\phi}_n(x)$ The nth orthonormal function in an orthonormal function sequence.

$\hat{\phi}_n(\mathbf{x})$ The nth discrete orthonormal function (i.e. an orthonormal vector) in an orthogonal function sequence.

$\mathbf{T}$ The transition matrix from the non–orthogonal basis to the orthonormal basis.

1.2.2 Orthogonal Functions

A function (or vector) sequence $\{\phi_0, \phi_1, \phi_2, \ldots\}$ is said to be orthogonal if it obeys the following condition,

$$\langle \phi_m, \phi_n \rangle = \left\{ \begin{array}{ll} 0 & , \; m \neq n \\ h_m & , \; m = n \end{array} \right. \tag{1.12}$$

where $\langle \cdot, \cdot \rangle$ is the inner product, and h_m is a real finite non-zero scalar. If $h_m \equiv 1$ then the sequence is called orthonormal. ϕ_k is used to denote both $\phi_k(x)$ and $\phi_k(\mathbf{x})$, where x is a real scalar and $\mathbf{x}$ is a real vector. $\phi_k(\mathbf{x})$ is a discrete function produced by evaluating $\phi_k(x)$ for $x = x_i$, where x_i are the elements of $\mathbf{x}$.

Continuous Functions

For continuous functions, the inner product is defined as follows,

$$\langle \phi_m(x), \phi_n(x) \rangle \triangleq \int_a^b w(x)\phi_m^*(x)\phi_n(x)\, dx \qquad (1.13)$$

where $w(x)$ is a weighting function, and $(\cdot)^*$ represents the complex conjugate. The weighting function $w(x)$ is real, non-negative and integrable on the interval (a, b), and $w(x) > 0$ on a sufficiently large subset of the interval such that the following is true,

$$0 < \int_a^b w(x)\, dx < \infty.$$

Remark: In this treatment, only finite intervals will be dealt with (i.e. $|a|, |b| < \infty$).

Discrete Functions

For discrete functions, the inner product is defined as follows,

$$\langle \phi_m(\mathbf{x}), \phi_n(\mathbf{x}) \rangle \triangleq \phi_m(\mathbf{x})^H W(\mathbf{x})\phi_n(\mathbf{x}) \qquad (1.14)$$

where

$$\mathbf{x} = \begin{bmatrix} x_1 & x_2 & \cdots & x_N \end{bmatrix}^T$$
$$\phi_m(\mathbf{x}) = \begin{bmatrix} \phi_m(x_1) & \phi_m(x_2) & \cdots & \phi_m(x_N) \end{bmatrix}^T$$

and $W(\mathbf{x})$ is a diagonal matrix containing the weights, $[\cdot]^H$ represents the complex conjugate transpose of a matrix. The weights, $w(\mathbf{x}) = diag(W(\mathbf{x}))$, are real, non-negative and $0 < \sum_{k=1}^N w(x_k) < \infty$.

1.2.3   Examples

A couple of examples of common orthogonal functions are shown in order to familiarize the reader with these concepts.

The Exponential Function

Let the orthogonal function be given by $\tilde{\phi}_n(\omega) = e^{jn\omega T}$, the weighting function by $w(\omega) = 1$, and the interval by $(a,b) = (0, \frac{2\pi}{T})$. Then

$$
\begin{aligned}
\int_0^{\frac{2\pi}{T}} e^{jm\omega T} e^{-jn\omega T}\, d\omega &= \int_0^{\frac{2\pi}{T}} (\cos(m\omega T) + j\sin(m\omega T))(\cos(n\omega T) - j\sin(n\omega T))\, d\omega \\
&= \int_0^{\frac{2\pi}{T}} (\cos(m\omega T)\cos(n\omega T) + \sin(m\omega T)\sin(n\omega T))\, d\omega \\
&\quad + j\int_0^{\frac{2\pi}{T}} (-\cos(m\omega T)\sin(n\omega T) + \sin(m\omega T)\cos(n\omega T))\, d\omega \\
&= \int_0^{\frac{2\pi}{T}} \cos((m-n)\omega T)\, d\omega + j\int_0^{\frac{2\pi}{T}} \sin((m-n)\omega T)\, d\omega \\
&= \begin{cases} 0 & , m \neq n \\ \frac{2\pi}{T} & , m = n \end{cases}.
\end{aligned}
$$

Hence the complex function $e^{jn\omega T}$ is orthogonal on the interval $(0, \frac{2\pi}{T})$ using the weighting function $w(\omega) = 1$.

The exponential function also forms a discrete orthogonal function when the frequencies are linearly spaced (i.e. $\omega = \omega_\ell = \frac{2\pi\ell}{TN}$, $\ell = 0, 1, \ldots, N-1$), and the weighting is uniform (i.e. $W(\omega) = \mathbf{I}$, $\mathbf{I}$ is the identity matrix and $\omega = \begin{bmatrix} \omega_0 & \omega_1 & \cdots & \omega_{N-1} \end{bmatrix}^T$). This fact becomes important when performing system identification in the z domain. A proof of this is shown in Appendix 1.9.1.

The Chebyshev Polynomials of the First Kind

First the orthogonal function $\tilde{\phi}_n(\theta) = \cos(n\theta)$ with the weighting function $w(\theta) = 1$ on the interval $(a,b) = (0, \pi)$ will be developed as follows,

$$
\begin{aligned}
\int_0^{\pi} \cos(m\theta)\cos(n\theta)\, d\theta &= \frac{1}{2}\int_0^{\pi} (\cos((m+n)\theta) + (\cos((m-n)\theta)\, d\theta \\
&= \begin{cases} 0 & , m \neq n \\ \pi & , m = n = 0 \\ \frac{\pi}{2} & , m = n \neq 0 \end{cases}.
\end{aligned}
$$

So, $\cos(n\theta)$ is an orthogonal function on the interval $(0, \pi)$ using a uniform weighting. Now, make the change of variable $x = \cos(\theta)$. The previous

development with this change of variables produces,

$$\int_{-1}^{1} (1-x^2)^{-\frac{1}{2}} T_m(x) T_n(x) \, dx = \begin{cases} 0 & , m \neq n \\ \pi & , m = n = 0 \\ \frac{\pi}{2} & , m = n \neq 0 \end{cases}$$

where the weighting function is given by $w(x) = (1 - x^2)^{-\frac{1}{2}}$ and $T_n(x) = \cos(n \cos^{-1}(x))$ are the Chebyshev polynomials of the 1st kind. In polynomial form, the Chebyshev polynomials are given by the following,

$$\begin{aligned} T_0(x) &= 1 \\ T_1(x) &= x \\ T_2(x) &= 2x^2 - 1 \\ T_3(x) &= 4x^3 - 3x \\ T_{n+1}(x) &= 2xT_n(x) - T_{n-1}(x). \end{aligned}$$

1.2.4 Gram-Schmidt Orthonormalization

The Gram–Schmidt orthonormalization procedure is outlined as follows. Begin with a set of linearly independent functions (or vectors) $\{\phi_0, \phi_1, \ldots, \phi_n\}$. Then form the corresponding orthonormal functions (or vectors) as follows:

$$\begin{aligned} \tilde{\phi}_0 &= \phi_0 \\ \hat{\phi}_0 &= \frac{\tilde{\phi}_0}{(\langle \tilde{\phi}_0, \tilde{\phi}_0 \rangle)^{\frac{1}{2}}} \\ \tilde{\phi}_{m+1} &= \phi_{m+1} - \sum_{k=0}^{m} \langle \phi_{m+1}, \hat{\phi}_k \rangle \hat{\phi}_k \\ \hat{\phi}_{m+1} &= \frac{\tilde{\phi}_{m+1}}{(\langle \tilde{\phi}_{m+1}, \tilde{\phi}_{m+1} \rangle)^{\frac{1}{2}}} \end{aligned}$$

The resulting sequence $\{\hat{\phi}_0, \hat{\phi}_1, \ldots, \hat{\phi}_n\}$ is orthonormal.

1.2.5 The Weighted Least Squares Problem

The system identification procedure in this report uses the weighted least squares solution. Some simple examples will be done in order to illustrate the characteristics of weighted least squares solutions using orthogonal functions.

The Least Squares Problem

Suppose it is desired to approximate a function $f(x)$ using a finite orthogonal function sequence $\{\tilde{\phi}_0(x), \tilde{\phi}_1(x), \ldots, \tilde{\phi}_n(x)\}$. Let the following relationship exist,

$$f(x) \approx \sum_{k=0}^{n} b_k \tilde{\phi}_k(x). \tag{1.15}$$

By multiplying through by $w(x)\tilde{\phi}_m^*(x)$ $(0 \leq m \leq n)$, and integrating, b_m is found as,

$$b_m = \frac{\int_a^b \tilde{\phi}_m^*(x) f(x) dx}{h_m}. \tag{1.16}$$

This corresponds to the weighted least squares solution to the problem, where the weighting is given by $w(x)$.

The Role of the Weighting Function $w(x)$

The weighting function $w(x)$ ends up being the weighting on the least squares problem of equation 1.15. In this section a couple of examples are done in order to illustrate the importance of the weighting function $w(x)$.

Example Using Chebyshev Polynomials of the 1st Kind   The polynomial $f(x) = (x-1)^3$ is approximated using the first three Chebyshev polynomials of the 1st kind (i.e. $T_0(x) = 1$, $T_1(x) = x$ and $T_2(x) = 2x^2 - 1$) on the interval $(a, b) = (-1, 1)$. The Chebyshev polynomials of the 1st kind have a weighting function of $w(x) = (1 - x^2)^{-\frac{1}{2}}$ as shown in figure 1.1. The result is the following,

$$f(x) \approx -\frac{5}{2} + \frac{15}{4}x - \frac{3}{2}(2x^2 - 1).$$

Plots of the approximation and the approximation error are shown in figures 1.2 and 1.3.

Example Using Chebyshev Polynomials of the 2nd Kind   Again the polynomial $f(x) = (x-1)^3$ is approximated, but this time using the first three Chebyshev polynomials of the 2nd kind (i.e. $T_0(x) = 1$, $T_1(x) = 2x$ and $T_2(x) = 4x^2 - 1$) on the interval $(a, b) = (-1, 1)$. The Chebyshev polynomials

13

of the 2nd kind have a weighting function of $w(x) = (1 - x^2)^{\frac{1}{2}}$ as shown in figure 1.1. The result is the following,
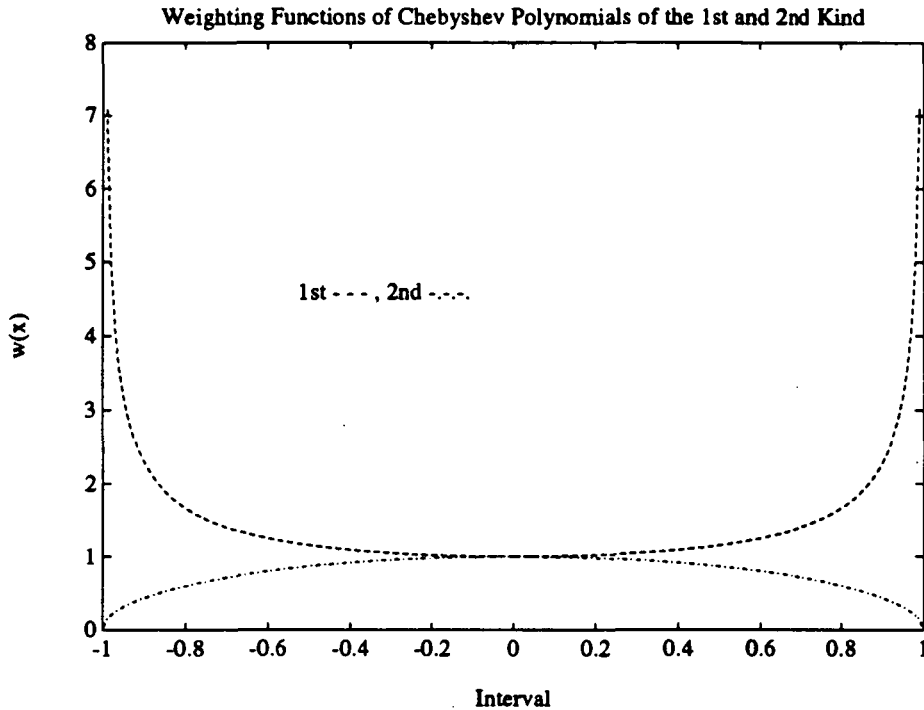

Figure 1.1: Plot of the Weighting Functions for Chebyshev Polynomials of the 1st and 2nd Kind

$$f(x) \approx -\frac{7}{4} + \frac{7}{2}(2x) - \frac{3}{4}(4x^2 - 1).$$

Plots of the approximation and the approximation error are shown in figures 1.2 and 1.3 for comparison with the results using Chebyshev polynomials of the 1st kind.

Notice that the error in the middle part of the interval is less using the Chebyshev polynomials of the 2nd kind (see figure 1.3). This can easily be explained by inspection of the corresponding weight functions (see figure 1.1). Since the Chebyshev polynomials of the 2nd kind place more relative weighting on the middle of the interval this method is more accurate there,
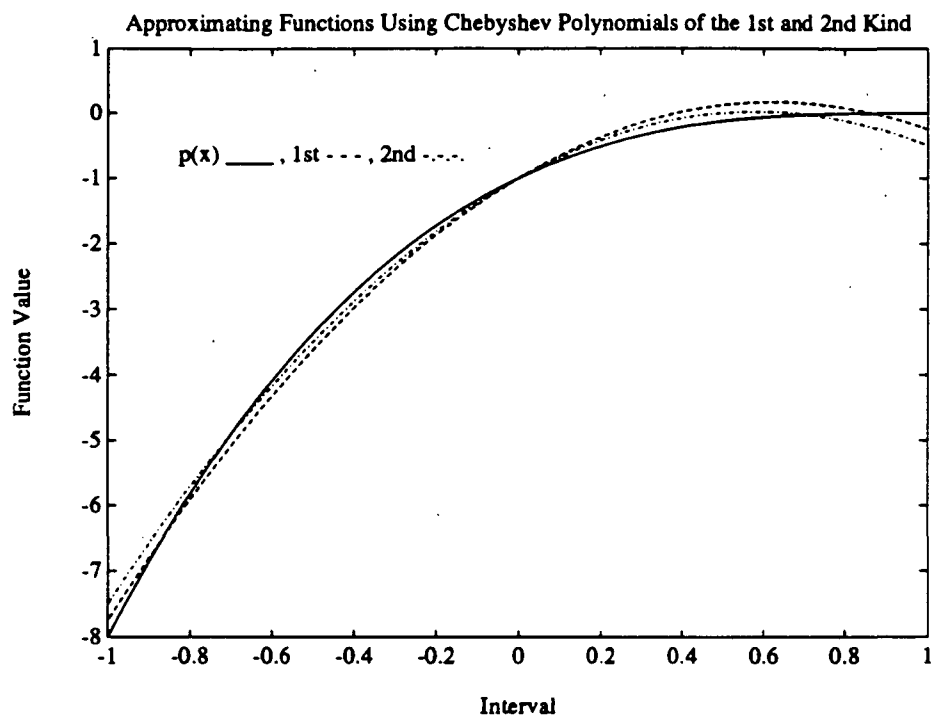
14

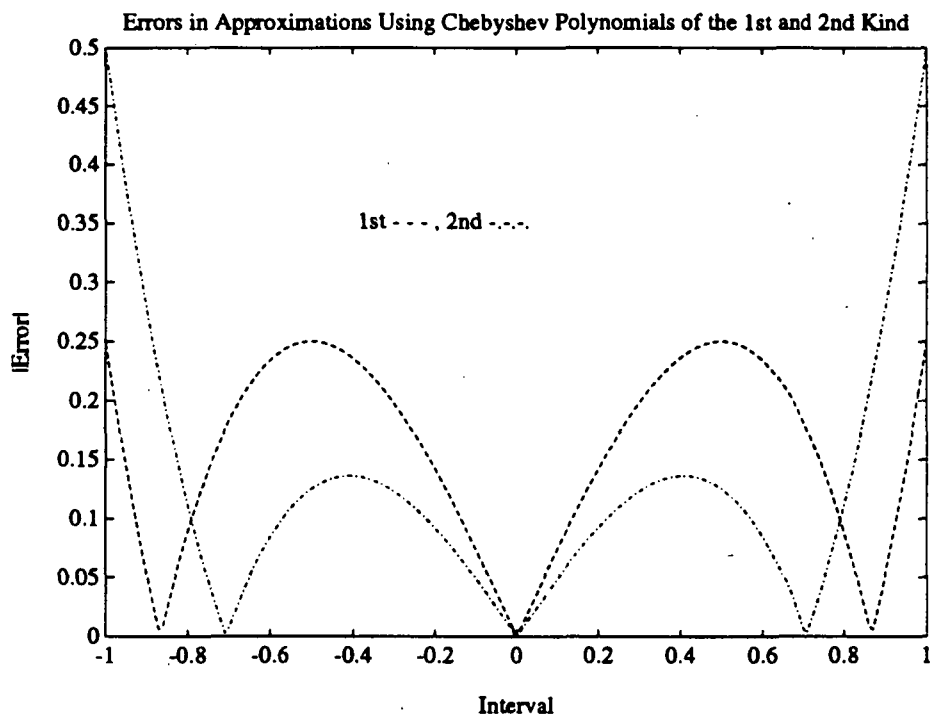Figure 1.2: Plot of the Approximating Functions Using Chebyshev Polynomials of the 1st and 2nd Kind

Figure 1.3: Plot of the Approximation Errors Using Chebyshev Polynomials
of the 1st and 2nd Kind

while the relative weighting on the middle of the interval is much less for the Chebyshev polynomials of the 1st kind.

1.3 Problem Statement

The advantage of using orthogonal functions in system identification is that the approximation of the system data using a sequence of orthogonal functions is numerically more robust than an approximation using non–orthogonal functions. A paradigm of this is the approximation of a vector by a basis. Typically, the "best" basis to use is an orthogonal basis. Following is a development to illustrate where the difficulties arise.

1.3.1 Fitting an Auto–Regressive Moving–Average Model to Frequency Domain Data

An Auto–Regressive Moving–Average (ARMA) model in the s domain may be expressed as:

$$\hat{y}(s,\boldsymbol{\theta}) = \sum_{k=0}^{n} b_k s^k u(s) + \sum_{k=1}^{n} a_k s^k y(s)$$

where $n_a = n_b = n$ for simplicity. Let $s = j\omega$, and assume that $u(j\omega) = 1$. Then $y(j\omega) = \mathcal{G}(\omega)$, where $\mathcal{G}(\omega)$ is the frequency domain data. If we wish to minimize the least squares cost function $J = (y(j\omega) - \hat{y}(j\omega,\boldsymbol{\theta}))^H (y(j\omega) - \hat{y}(j\omega,\boldsymbol{\theta}))$ the following least squares problem must be solved,

$$0 = \begin{bmatrix} \Phi_a & \Phi_b \end{bmatrix} \begin{bmatrix} -1 \\ \boldsymbol{\theta} \end{bmatrix}$$

where 0 is an appropriately sized zero matrix (a vector in this case), and

$$\Phi_a = [\mathcal{G}(\omega)] \, \Phi_b$$

$$[\mathcal{G}(\omega)] = \begin{bmatrix} \mathcal{G}(\omega_0) & 0 & \cdots & 0 \\ 0 & \mathcal{G}(\omega_1) & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \mathcal{G}(\omega_{N-1}) \end{bmatrix}$$

17

$$\Phi_b = \begin{bmatrix} 1 & j\omega_0 & \cdots & (j\omega_0)^n \\ 1 & j\omega_1 & \cdots & (j\omega_1)^n \\ \vdots & \vdots & & \vdots \\ 1 & j\omega_{N-1} & \cdots & (j\omega_{N-1})^n \end{bmatrix}$$

$$\theta = \begin{bmatrix} a_1 & a_2 & \cdots & a_n b_0 & b_1 & \cdots & b_n \end{bmatrix}^T.$$

The regressor matrix Φ_b contains the discrete basis functions for $B(j\omega, \theta)$. Let us take a closer look at these basis functions. It can be shown that column vectors which make up the regressor matrix Φ_b are not close to being orthogonal, and therefore, Φ_b is not well conditioned. To illustrate this point a plot of the first few normalized column vectors is shown in figure 1.4. (In
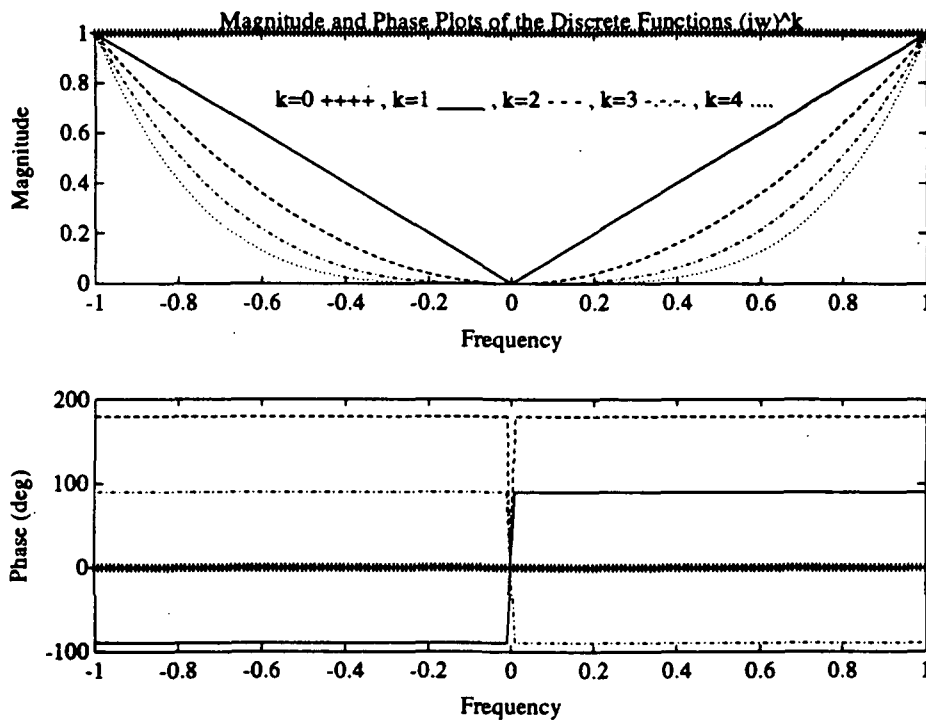

Figure 1.4: The First Few Column Vectors of the Matrix Φ_b

order to improve the numerics of the s domain formulation, the frequencies are often normalized such that ω is replaced by $\hat{\omega} = \frac{\omega}{\omega_{max}}$). If a function

18

which is orthogonal on the interval (ω_0, ω_{N-1}) where used instead, the new regressor matrix $\hat{\Phi}_b$ would be much better conditioned, and therefore the results would be numerically more robust.

The first few Gram–Schmidt orthonormalized column vectors using the weighting $w(\omega) = \frac{1}{(1-\omega^2)^{\frac{1}{2}}}$, where $\omega = \omega_\ell = -1 + \frac{\ell}{200}$, $\ell = 1, 2, \cdots, 199$, are shown in figure 1.5. This weighting corresponds to the weighting used by the
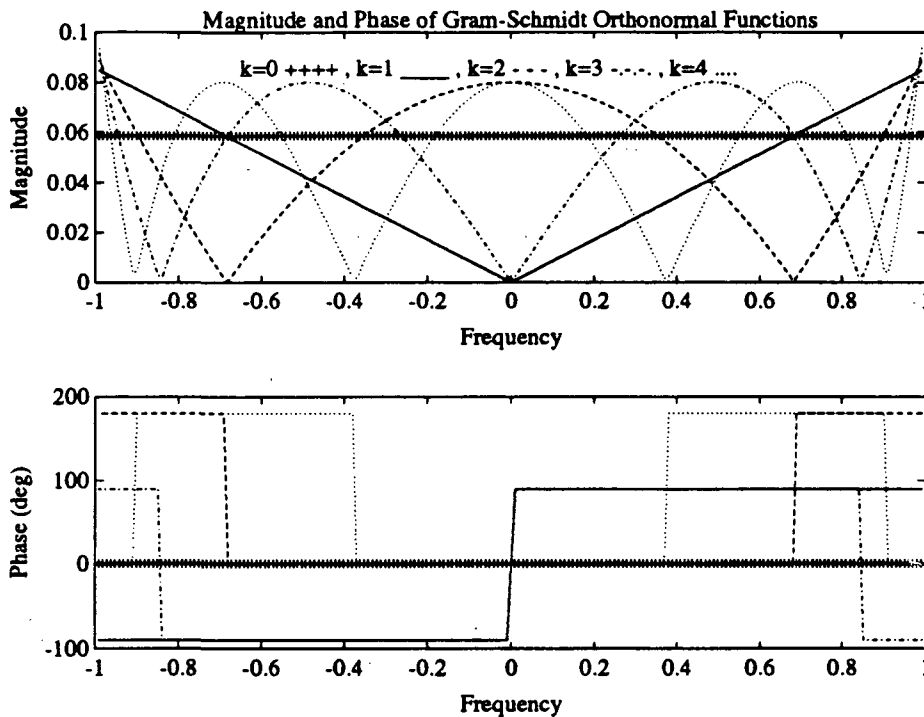

Figure 1.5: The First Few Column Vectors of the Matrix $\hat{\Phi}_b$

Chebyshev polynomials of the 1st kind. For the purpose of comparison, a plot of the magnitudes of the first few Gram–Schmidt orthonormal vectors and the first few normalized Chebyshev vectors are shown in figure 1.6. Notice the slight difference. The vectors formed using the Chebyshev polynomials of the 1st kind are not orthogonal, because we are taking ω at linearly spaced discrete points and the Chebyshev polynomials are not orthogonal as discrete functions for ω linearly spaced.
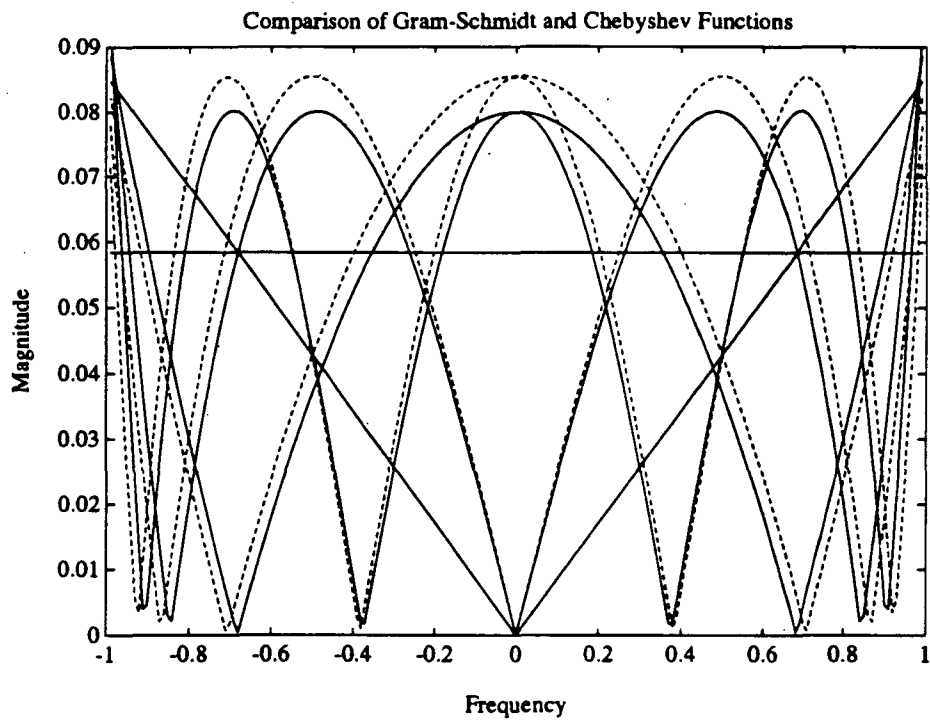
19

Figure 1.6: Comparison of Gram–Schmidt (solid line) and Chebyshev Functions (dashed line)

20

1.4 Background

The use of particular orthogonal functions has been proposed by researchers in [6], [1]. Following is a brief discussion of other work which directly relates to the present work.

1.4.1 The z Domain

When performing system identification using frequency data in the z domain, the substitution $z = e^{j\omega T}$ is used. As shown in section 1.2.3, the functions $1, e^{j\omega T}, \ldots, e^{jn\omega T}$ are orthogonal for the weighting $w(\omega) = 1$ on the interval $(0, 2\pi)$. Hence, the z domain already uses a sum of orthogonal functions to approximate the function. Although, it is only orthogonal when using uniform weighting and the complete range $(0, 2\pi)$ of linearly spaced points.

1.4.2 The s Domain

When performing system identification using frequency data in the s domain, the substitution $s = j\omega$ is used. In section 1.3.1, it was shown that the polynomials $1, j\omega, \ldots, (j\omega)^n$ are not orthogonal. Dailey and Lukich [6] recommend the use of Chebyshev polynomials of the 1st kind in this situation. The following transfer function relationship between the polynomials of $j\omega$ and the Chebyshev polynomials of ω can be made,

$$
\frac{B(j\omega, \boldsymbol{\theta})}{A(j\omega, \boldsymbol{\theta})} = \frac{\sum_{\substack{k=0 \\ k\,even}}^{n} (-1)^{\frac{k}{2}} b_k \omega^k + j \sum_{\substack{k=0 \\ k\,odd}}^{n} (-1)^{\frac{k-1}{2}} b_k \omega^k}{\sum_{\substack{k=0 \\ k\,even}}^{n} (-1)^{\frac{k}{2}} a_k \omega^k + j \sum_{\substack{k=0 \\ k\,odd}}^{n} (-1)^{\frac{k-1}{2}} a_k \omega^k}
$$

$$
= \frac{\sum_{\substack{k=0 \\ k\,even}}^{n} \beta_k T_k(\omega) + j \sum_{\substack{k=0 \\ k\,odd}}^{n} \beta_k T_k(\omega)}{\sum_{\substack{k=0 \\ k\,even}}^{n} \alpha_k T_k(\omega) + j \sum_{\substack{k=0 \\ k\,odd}}^{n} \alpha_k T_k(\omega)}.
$$

Identification may be performed using the Chebyshev polynomials, and then converting the result from the Chebyshev domain to the s (or $j\omega$) domain.

This method shows promise, but it may be possible to improve on it. Note that the Chebyshev polynomials of the 1st kind are orthogonal on the continuous interval $(-1, 1)$ with a weighting of $w(\omega) = (1 - \omega^2)^{-\frac{1}{2}}$. Hence in

all cases of system identification the Chebyshev polynomials are only approximately orthogonal, because the data, $\mathcal{G}(\omega)$, is only known at a finite number a frequencies, and $\sum_{k=0}^{N-1} w(\omega_k)T_m(\omega_k)T_n(\omega_k) \neq 0$, for all $m \neq n$. Also, if a weighting other than $w(\omega) = (1 - \omega^2)^{-\frac{1}{2}}$ is used the Chebyshev polynomials of the 1st kind are not orthogonal.

1.4.3 The δ Domain

Recall that $\delta \triangleq \frac{z-1}{T}$. Therefore in the frequency domain, $\delta = \frac{e^{j\omega T}-1}{T}$. Bayard [1], recommends the use of the Chebyshev polynomials of the 1st kind in the δ domain when the following condition is met: The sampling rate is very fast (or T very small) when compared to the frequency of the highest mode of the system. When this condition is met the δ domain is very similar to the s domain, hence the Chebyshev polynomials are approximately orthogonal in the δ domain.

The Gram–Schmidt procedure outlined in this report allows one to find an orthonormalization of any domain, and with any weighting.

1.5 System Identification Using Gram–Schmidt Orthonormalization

The first step in system identification is to obtain data, and in this procedure we begin with frequency data. The procedure of obtaining data falls under the heading "experiment design". Ljung [8] devotes a chapter to this subject. Here we will briefly mention some of the more popular methods of obtaining frequency domain data.

Perhaps the most obvious method is a sine dwell or sine sweep method. For the sine dwell method, the system is forced with a sine wave at constant frequency and magnitude until the transients die out, and then the phase and magnitude of the output are found. This is then done for additional frequencies until an adequate frequency record is found. The sine sweep method is similar except that the forcing frequency is not constant, but adjusted slowly.

Another method is to use time domain data from the input and output. This data is transformed into the frequency domain using the Discrete Fourier

Transform (DFT), and the output frequency data is divided by the input frequency data in order to find the system frequency data. Care must be taken to insure that the input used is "rich" enough in the frequency domain so that the results are accurate. Also, There are several other issues that must be dealt with here. Two of the most important issues are "aliasing" which is related to the high frequency modes of the system and sampling rate, and "windowing" which is important if the input data record does not contain an integer number of periods of the input signal.

1.5.1　The General Procedure

1. *Obtain frequency data $\mathcal{G}(\omega)$.*

2. *Choose the Model Type.* We will assume the most general model here, the ARMA model,

$$\hat{y}(p, \theta) = -\sum_{k=1}^{n_a} a_k p^k \mathcal{G}(\omega) + \sum_{k=0}^{n_b} b_k p^k. \tag{1.17}$$

where we assume that $u(p) = 1$ and $y(p) = \mathcal{G}(\omega)$. In order to keep this development general, the frequency dependence of p will be shown using the notation $p = p(\omega)$. For simplicity, let $n_a = n_b = n$. This leads to the following equation for the weighted linear least squares problem,

$$0 = W(\omega)^{\frac{1}{2}} \begin{bmatrix} \Phi_a & \Phi_b \end{bmatrix} \begin{bmatrix} 1 \\ \theta \end{bmatrix} \tag{1.18}$$

where 0 is an appropriately sized matrix of zeros (in this case a column vector of size $(N - 1) \times 1$),

$$W(\omega)^{\frac{1}{2}} = \begin{bmatrix} w(\omega_0)^{\frac{1}{2}} & 0 & \cdots & 0 \\ 0 & w(\omega_1)^{\frac{1}{2}} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & w(\omega_{N-1})^{\frac{1}{2}} \end{bmatrix}$$

$$\Phi_a = \begin{bmatrix} \mathcal{G}(\omega_0) & \mathcal{G}(\omega_0)p(\omega_0) & \cdots & \mathcal{G}(\omega_0)p(\omega_0)^n \\ \mathcal{G}(\omega_1) & \mathcal{G}(\omega_1)p(\omega_1) & \cdots & \mathcal{G}(\omega_1)p(\omega_1)^n \\ \vdots & \vdots & & \vdots \\ \mathcal{G}(\omega_{N-1}) & \mathcal{G}(\omega_{N-1})p(\omega_{N-1}) & \cdots & \mathcal{G}(\omega_{N-1})p(\omega_{N-1})^n \end{bmatrix}$$

23

$$\Phi_b = \begin{bmatrix} 1 & p(\omega_0) & \cdots & p(\omega_0)^n \\ 1 & p(\omega_1) & \cdots & p(\omega_1)^n \\ \vdots & \vdots & & \vdots \\ 1 & p(\omega_{N-1}) & \cdots & p(\omega_{N-1})^n \end{bmatrix}.$$

$$\theta = \begin{bmatrix} -a_1 & -a_2 & \cdots & -a_n & b_0 & b_1 & \cdots & b_n \end{bmatrix}^T$$

3. *Orthonormalize the Regressor Matrix.* At this point we have a few choices of how to proceed. Orthogonalization could be done with respect to $W(\omega)^{\frac{1}{2}}\Phi_a$, $W(\omega)^{\frac{1}{2}}\Phi_b$ or $W(\omega)^{\frac{1}{2}}\begin{bmatrix} \Phi_a & \Phi_b \end{bmatrix}$. We will choose the following method: The orthonomalization will be done with respect to $W(\omega)^{\frac{1}{2}}\Phi_b$ such that $\hat{\Phi}_b = \Phi_b \mathbf{T}$ and $\hat{\Phi}_b^H W(\omega)\hat{\Phi}_b = \mathbf{I}$. Therefore, our least squares problem can be written as follows,

$$0 = W(\omega)^{\frac{1}{2}} \begin{bmatrix} \Phi_a \mathbf{T} & \Phi_b \mathbf{T} \end{bmatrix} \begin{bmatrix} \mathbf{T}^{-1} & \mathbf{0} \\ \mathbf{0} & \mathbf{T}^{-1} \end{bmatrix} \begin{bmatrix} -1 \\ \theta \end{bmatrix} \quad (1.19)$$

$$= W(\omega)^{\frac{1}{2}} \begin{bmatrix} \bar{\Phi}_a & \hat{\Phi}_b \end{bmatrix} \begin{bmatrix} c \\ \hat{\theta} \end{bmatrix} \quad (1.20)$$

where $\hat{\Phi}_b$ is orthonormal, $\bar{\Phi}_a$ is not orthonormal, c is a known constant (because $\mathbf{T}^{-1}$ is a lower triangular matrix), and $\hat{\theta}$ contains the parameters of the orthonormalized model.

4. *Solve the Linear Least Squares Problem.* Using Matlab notation, the weighted least squares equation is the following,

$$c W(\omega)^{\frac{1}{2}} \bar{\Phi}_a(:,1) = W(\omega)^{\frac{1}{2}} \begin{bmatrix} \bar{\Phi}_a(:,2:n_a) & \hat{\Phi}_b \end{bmatrix} \hat{\theta} \quad (1.21)$$

where $\bar{\Phi}_a(:,1)$ is the first column of $\bar{\Phi}_a$ and $\bar{\Phi}_a(:,2:n_a)$ is the rest of $\bar{\Phi}_a$.

5. *Convert the Model Back into the Original Domain (find θ).* The model parameters in the original domain can easily be found as,

$$\theta = \begin{bmatrix} \mathbf{T}(2:n+1,2:n+1) & \mathbf{0} \\ \mathbf{0} & \mathbf{T} \end{bmatrix} \hat{\theta}. \quad (1.22)$$

Remark 1: Although θ is real, $\hat{\theta}$ may not be real. In order to insure that **T** is real, and hence θ is real, the frequency data $\mathcal{G}(\omega)$ and the weighting $w(\omega)$ should be chosen to be symmetric about $\omega = 0$.

Remark 2: It should be pointed out that orthonormalizing with respect to $W(\omega)^{\frac{1}{2}}\Phi_b$ essentially finds an orthonormal basis made up of polynomials of $p(\omega)$, i.e.

$$\hat{\phi}_\ell(\omega) = \sum_{k=0}^{\ell} \alpha_k p^k(\omega)$$

where α_k is a scalar (and real if Remark 1 is followed).

1.6 Examples

The following examples use a single–input single–output state space model generated from a finite element model of the Jet Propulsion Laboratory (JPL), Control and Structure Interaction (CSI), phase B testbed. The model was obtained courtesy of JPL's CSI group, and in particular Dr. Sam Sirlin. The model has 33 detectable modes, in the frequency range from 0 to 128Hz.

For all the tests, the assumed order of the model began at 33 and was decreased until a stable minimum order model was identified. By "minimum order" we mean a model which does not contain any pole zero cancellations. Also, for all the tests 513 linearly spaced frequencies points in the range $(-\frac{\pi}{T}, \frac{\pi}{T})$ symmetrically positioned about $\omega = 0$ were used (i.e. $\omega = \omega_\ell = (-\pi + \frac{2\pi\ell}{512})/T$, $\ell = 0, 1, \ldots, 512$). Once $A(p, \theta)$ and $B(p, \theta)$ are found, they are used to form the model

$$y(p, \theta) = \frac{B(p, \theta)}{A(p, \theta)} u(p). \tag{1.23}$$

This model is then compared to the original data.

1.6.1 z Domain Identification

The first identification was done using a uniform weighting $(W(\omega) = \mathbf{I})$. Both the z domain and orthonormalized identified models are 33rd order.

25

The results are shown in figure 1.7. The magnitude of the system data $\|\mathcal{G}(\omega)\|_2$ is shown as well as the errors in the two identified models. The errors are defined as $\mathcal{G}(\omega) - y(e^{j\omega T}, \theta)$. Notice that the identified models are
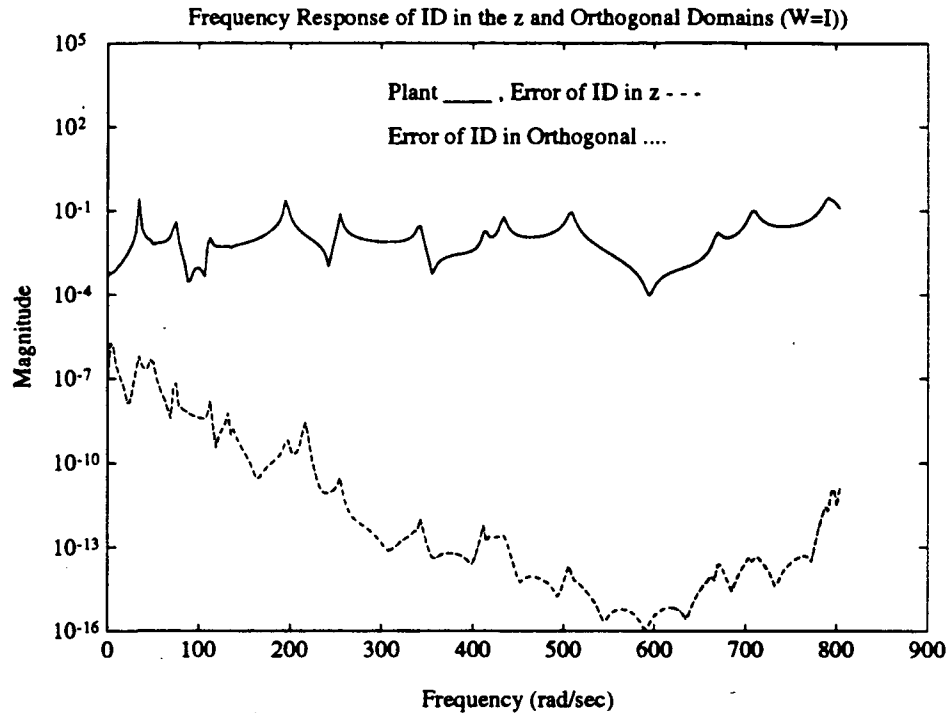

Figure 1.7: Frequency response magnitude plot of the plant, the error in the z domain identified model and the error in the orthonormalized identified model. The identification used uniform weighting.

almost indistinguishable. This result should be expected since the z domain is orthogonal for unit weighting (although it is not exactly orthogonal here since we are not using the proper ω in order to make it orthogonal).

Next, we will use the weighting $w(\omega) = \|\mathcal{G}(\omega)\|_2^2$. This is a reasonable weighting to use, because the weighting of $w(\omega) = \|\frac{1}{A(e^{j\omega T}, \theta)}\|_2^2$ gives us an unbiased model, but since we do not know $A(e^{j\omega T}, \theta)$ we use $\mathcal{G}(\omega)$ in its place. The models are both 32nd order. The results are shown in figure 1.8. The difference between the models is noticeable, but it is not apparent that one model is superior to the other.
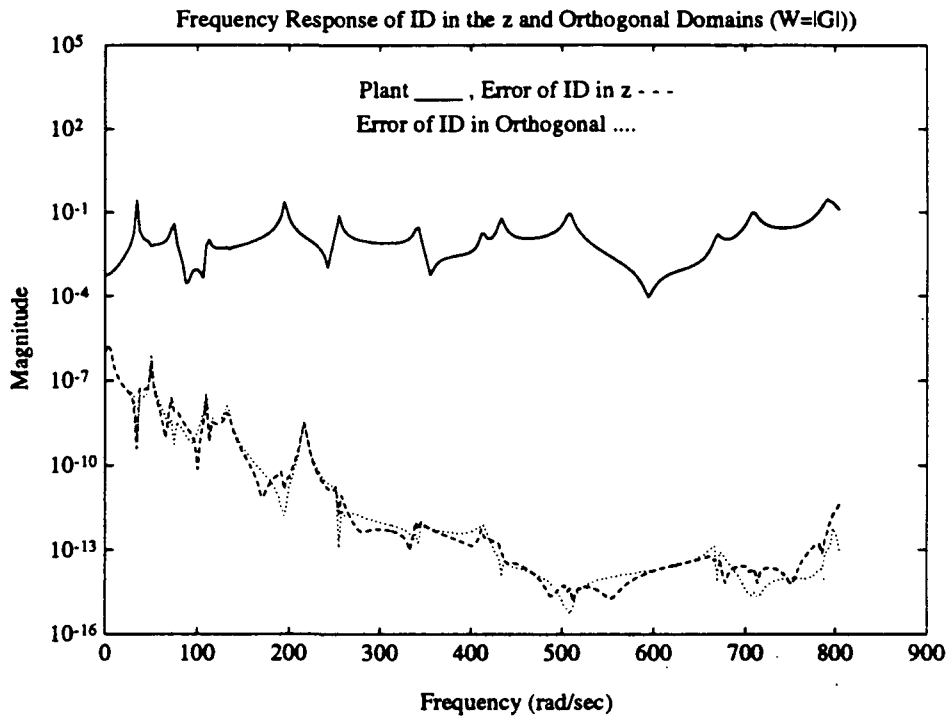
26

Figure 1.8: Frequency response magnitude plot of the plant, the error in the z domain identified model and the error in the orthonormalized identified model. The identification used a weighting of $\|\mathcal{G}(\omega)\|_2^2$.

1.6.2   s Domain Identification

All the identifications in this section use unit weighting, and the frequencies are the same as those in the z domain identification.

First, identification was done using the s domain and the Gram–Schmidt orthonormalized s domain. The s domain model is 3rd order, and the orthonormalized s domain model is 28th order. As can be seen in figure 1.9, the s domain identification model is not good at all, while the orthonormalized s domain model is much better.

Frequency Response of ID in the s and Orthogonal Domains (W=I)

Plant _____ , ID in s - - -
ID in Orthogonal Domain ......
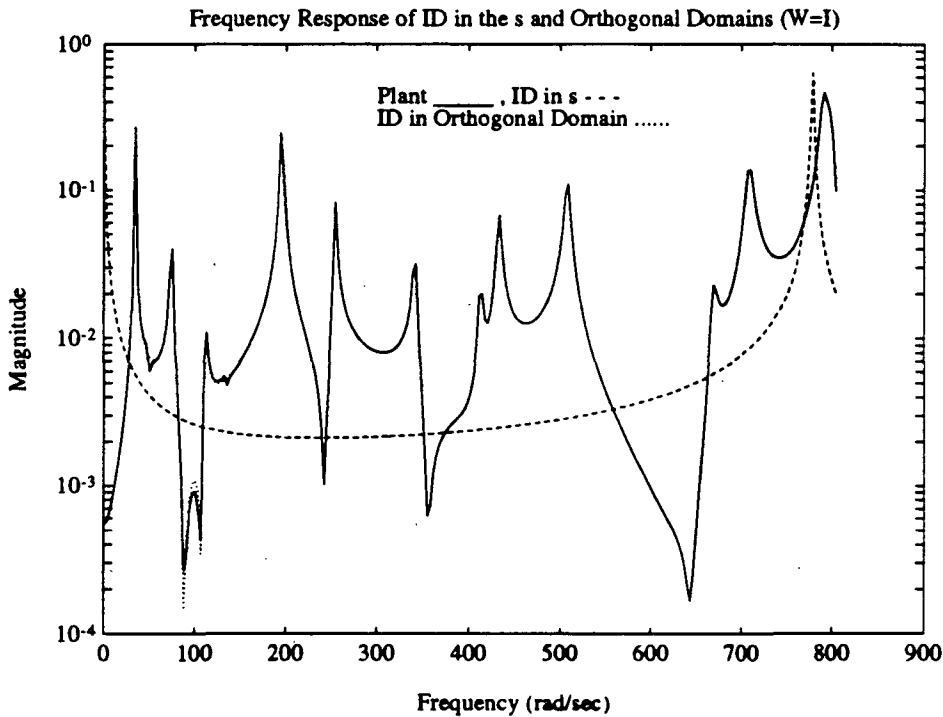
Figure 1.9: Frequency response magnitude plot of the plant, the s domain identified model and the orthonormalized s domain identified model. The identification used uniform weighting.

Next, identification was done using the scaled s domain and the Gram–Schmidt orthonormalized scaled s domain. The scaled s domain is used to improve the numerics of the s domain formulation. Scaling is done by

replacing ω by $\hat{\omega} = \frac{\omega}{\omega_{max}}$, where $\omega_{max} = max(|\omega|)$. The scaled s domain model is 24th order, and the orthonormalized scaled s domain model is 28th order. As can be seen in figure 1.10, the scaled s domain identification model is much improved over the unscaled s domain, but still not as good as the orthonormalized scaled s domain model, or the orthonormalized s domain model.
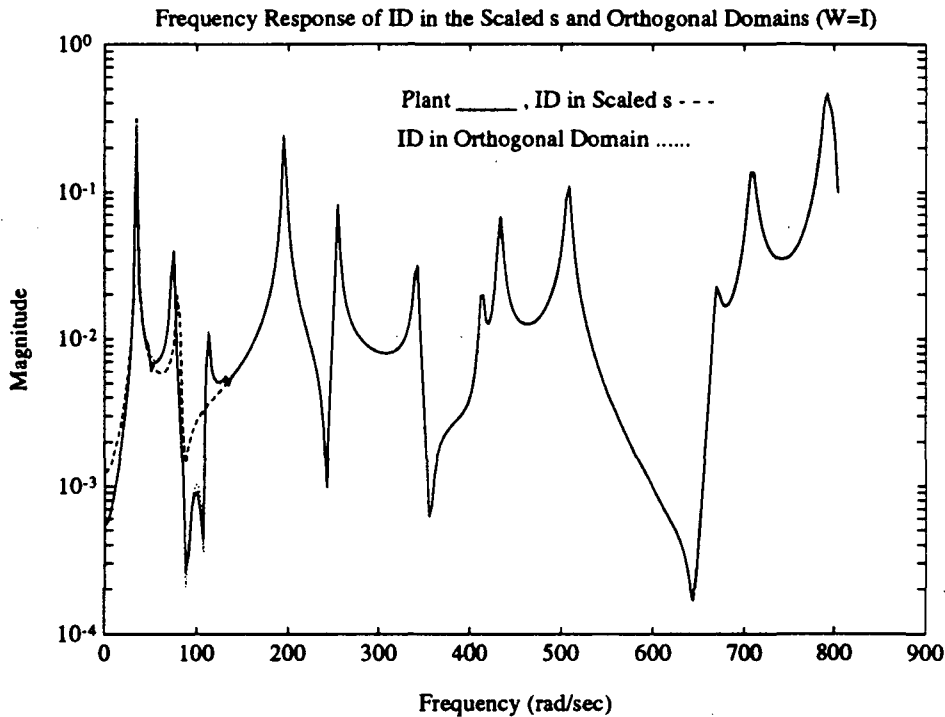

Figure 1.10: Frequency response magnitude plot of the plant, the scaled s domain identified model and the orthonormalized scaled s domain identified model. The identification used uniform weighting.

Figure 1.11 displays the error in the orthogonalized unscaled and scaled s domains for comparison. Both models are 28th order, and very similar. This result is not unexpected since the Gram–Schmidt orthonormalization procedure does its own scaling. Prescaling seems to have little effect on this method.
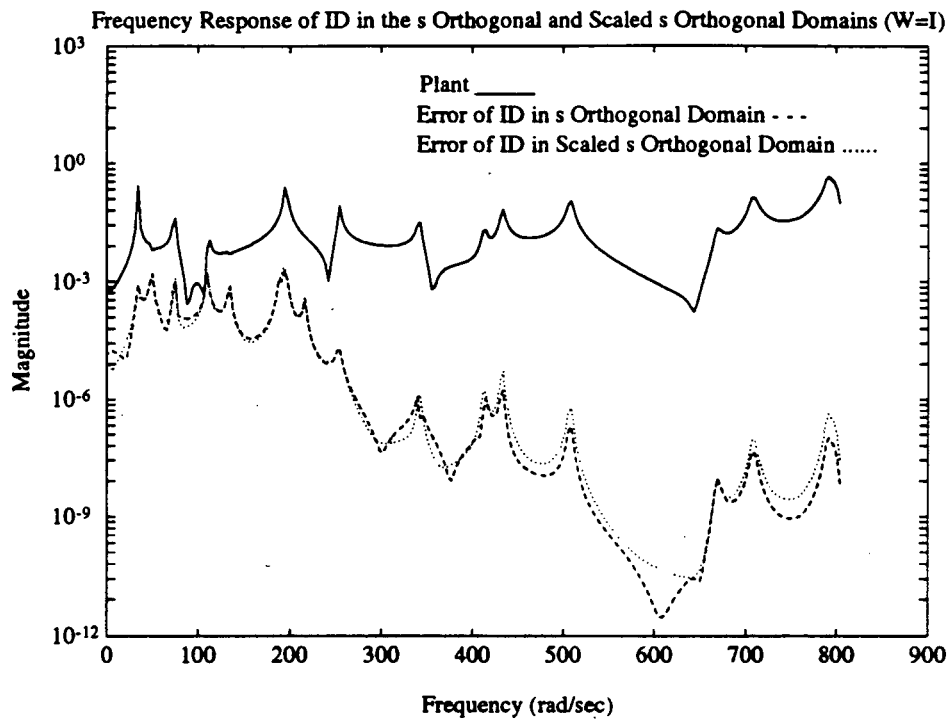
Figure 1.11: Frequency response magnitude plot of the plant, and the errors in the orthonormalized *s* domain identified model and the orthonormalized scaled *s* domain identified model. The identification used uniform weighting.

1.6.3 δ Domain Identification

All the identifications in this section use unit weighting, and the frequencies are the same as those in the z domain identification.

First, identification was done using the δ domain and the Gram–Schmidt orthonormalized δ domain. The highest order stable δ domain model is 4th order. As can be expected from a 4th order model approximating a 33rd order plant, the result was very poor. The orthonormalized δ domain did not produce a stable model for any order from 1 to 33. For the higher order attempts, the orthonormalized δ domain models fit the data well, but were unstable.

Next, identification was done using the scaled δ domain and the Gram–Schmidt orthonormalized scaled δ domain. Like the scaled s domain, the scaled δ domain is used to improve the numerics of the domain. Scaling is done by replacing $\nu \triangleq \frac{e^{j\omega T}-1}{T}$ by $\hat{\nu} = \frac{\nu}{\nu_{max}}$, where $\nu_{max} = max(|\nu|)$. The scaled δ domain model is 23rd order, and the orthonormalized scaled δ domain model is 22nd order. The results are shown in figure 1.12. The scaled δ domain identification model is much improved over the unscaled δ domain, and also better than the orthonormalized scaled δ domain model.

The Gram–Schmidt orthonormalization does not seem to be of benefit for this type of model in the δ domain. This result is surprising considering the success in the s domain. Part of the problem is that the higher order, better fitting models in the orthogonalized domain were unstable. So now we will compare the method once again, but this time allow unstable models. All the models are 33rd order and unstable.

Figure 1.13 displays the δ domain and the Gram–Schmidt orthonormalized δ domain models. The Gram–Schmidt orthonormalized δ domain model fits the data much better than the δ domain model. Figure 1.14 displays the scaled δ domain model and the Gram–Schmidt orthonormalized scaled δ domain model. These models are very similar. For a better comparison, the model errors are shown in figure 1.15. The orthonormalized scaled δ domain model is better in the low frequencies, but worse in the high frequencies compared to the scaled δ domain model. When the model order was permitted to increase the errors were much less, but the models were unstable.

Although system identification using Gram–Schmidt orthonormalization did not perform well for this model in the δ domain, it should be studied further. We will continue to research this problem with the intent of finding
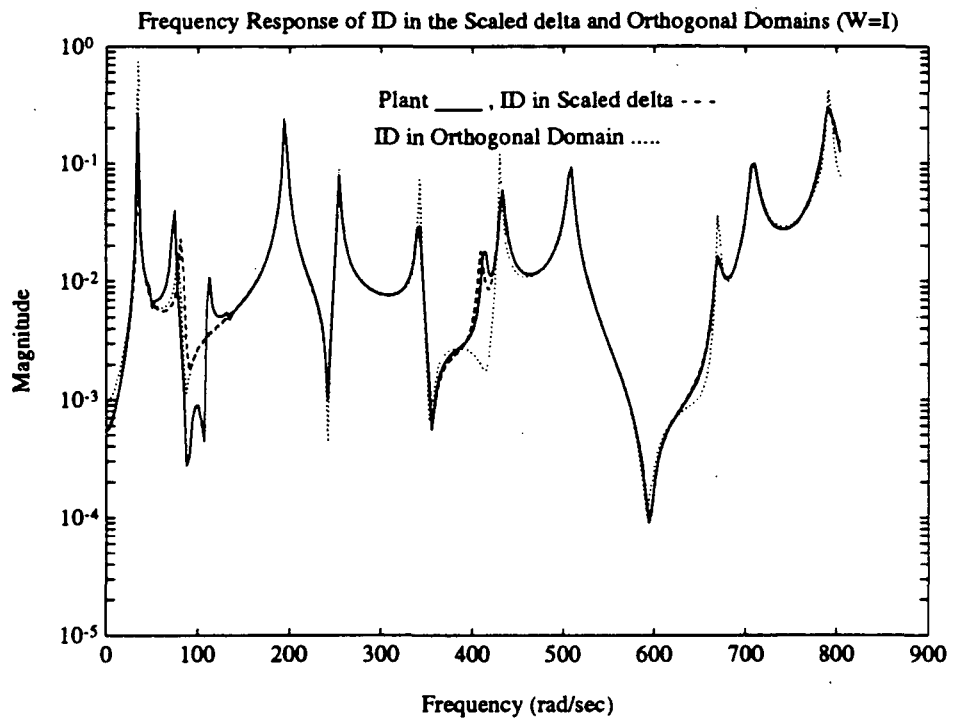
Figure 1.12: Frequency response magnitude plot of the plant, the scaled δ domain identified model and the orthonormalized scaled δ domain identified model. The identification used uniform weighting.
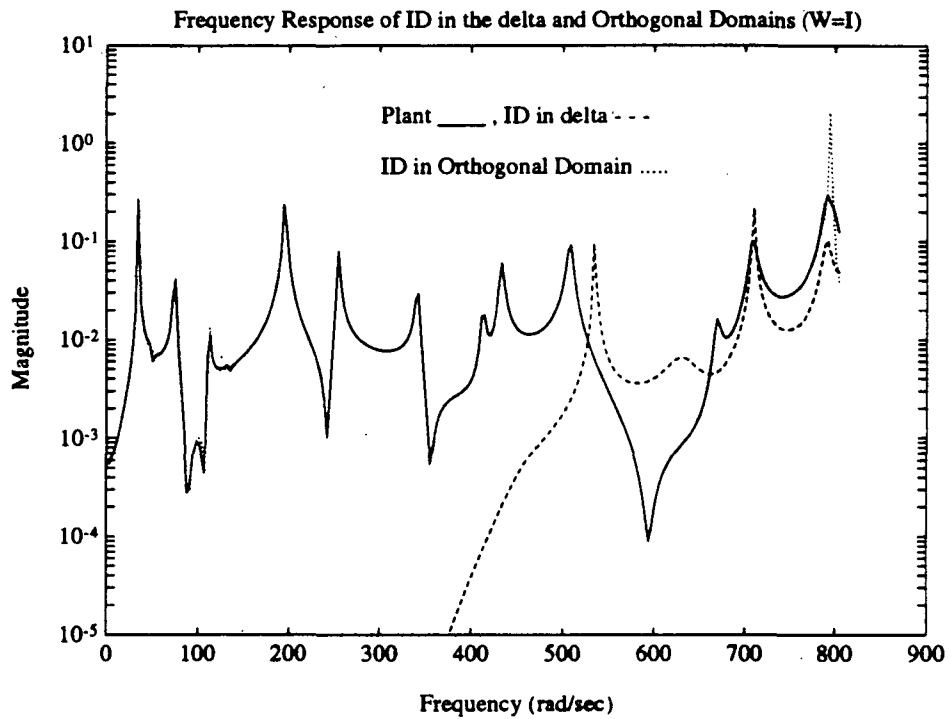
Figure 1.13: Frequency response magnitude plot of the plant, the δ domain identified model and the orthonormalized δ domain identified model, both 33rd order and unstable. The identification used uniform weighting.
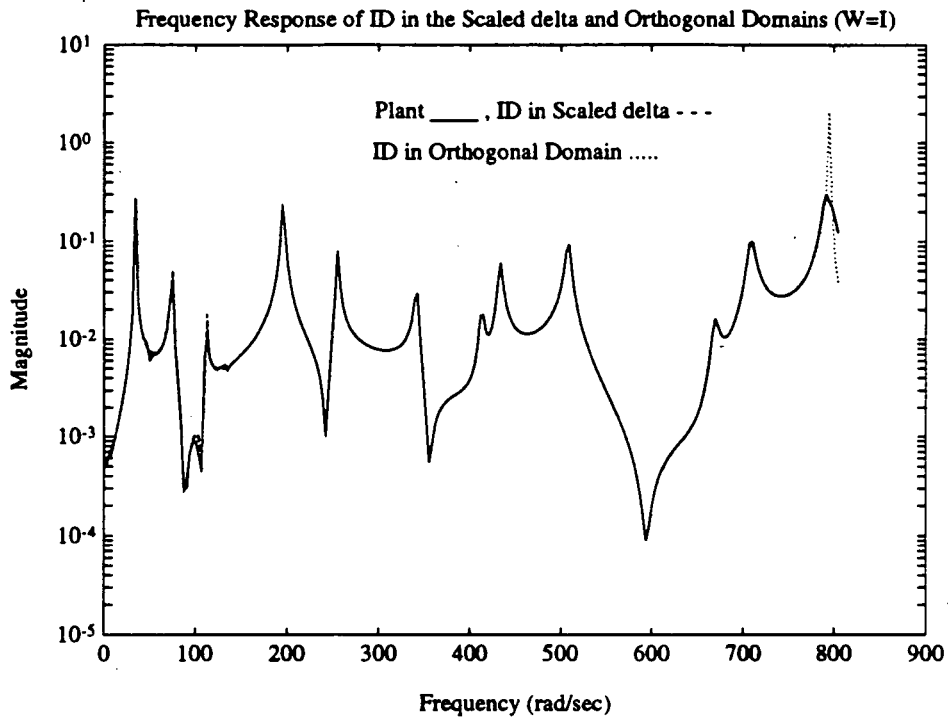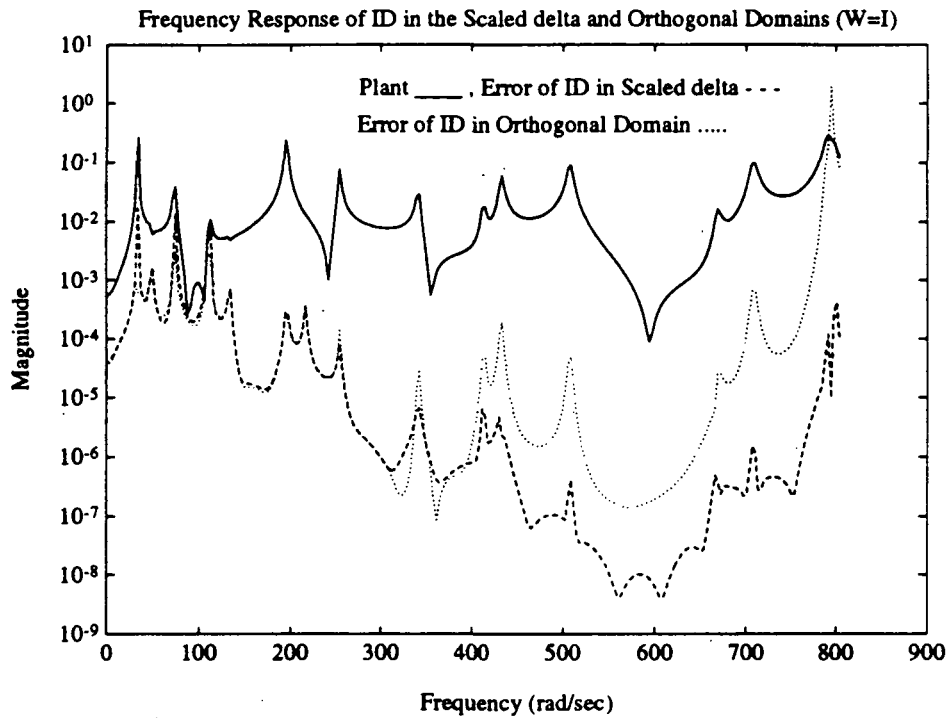
33

Figure 1.14: Frequency response magnitude plot of the plant, the scaled δ domain identified model and the orthonormalized scaled δ domain identified model, both 33rd order and unstable. The identification used uniform weighting.

34

Figure 1.15: Frequency response magnitude plot of the plant, and the errors in the scaled δ domain identified model and the orthonormalized scaled δ domain identified model, both 33rd order and unstable. The identification used uniform weighting.

the reasons for the difficulties and solutions to them.

1.7 Multiple–Input Multiple–Output Gram–Schmidt Orthonormal Identification

The MIMO system is the following,

$$\mathbf{y}(p) = \mathbf{G}(p)\mathbf{u}(p) + \mathbf{H}(p)\mathbf{e}(p) \tag{1.24}$$

where $\mathbf{y}(p)$, $\mathbf{u}(p)$ and $\mathbf{e}(p)$ are column vectors containing the output, input and noise respectively; $\mathbf{G}(p)$ and $\mathbf{H}(p)$ are the plant and noise matrix transfer functions. As in the single–input single–output case, we begin with the following model,

$$\mathbf{y}(p) = \hat{\mathbf{G}}(p,\boldsymbol{\theta})\mathbf{u}(p) + \hat{\mathbf{H}}(p,\boldsymbol{\theta})\hat{\mathbf{e}}(p,\boldsymbol{\theta}) \tag{1.25}$$

where $\hat{\mathbf{G}}(p,\boldsymbol{\theta})$, $\hat{\mathbf{H}}(p,\boldsymbol{\theta})$, and $\hat{\mathbf{e}}(p,\boldsymbol{\theta})$ are estimates of $\mathbf{G}(p)$, $\mathbf{H}(p)$, and $\mathbf{e}(p)$. Let $\hat{\mathbf{G}}(p,\boldsymbol{\theta}) = \frac{\mathbf{B}(p,\boldsymbol{\theta})}{A(p,\boldsymbol{\theta})}$, and assume that $\hat{\mathbf{H}}(p,\boldsymbol{\theta}) = \frac{\mathbf{I}}{A(p,\boldsymbol{\theta})}$, where $A(p,\boldsymbol{\theta})$ is a scalar transfer function, $\mathbf{B}(p,\boldsymbol{\theta})$ is a matrix transfer function, and $\mathbf{I}$ is an appropriately sized identity matrix. The assumption that $\hat{\mathbf{H}}(p,\boldsymbol{\theta}) = \frac{\mathbf{I}}{A(p,\boldsymbol{\theta})}$ greatly simplifies the result. Now, by multiplying through by $A(p,\boldsymbol{\theta})$, and substituting $\mathbf{e}(p,\boldsymbol{\theta}) = \mathbf{y}(p) - \hat{\mathbf{y}}(p,\boldsymbol{\theta})$ the following equation is found

$$\hat{\mathbf{y}}(p,\boldsymbol{\theta}) = \mathbf{B}(p,\boldsymbol{\theta})\mathbf{u}(p) + (1 - A(p,\boldsymbol{\theta})^{-1})\mathbf{y}(p) \tag{1.26}$$

where

$$\mathbf{B}(p,\boldsymbol{\theta}) = \begin{bmatrix} B_{11}(p,\boldsymbol{\theta}) & B_{12}(p,\boldsymbol{\theta}) & \cdots & B_{1n_i}(p,\boldsymbol{\theta}) \\ B_{21}(p,\boldsymbol{\theta}) & & & \vdots \\ \vdots & & & \\ B_{n_o1}(p,\boldsymbol{\theta}) & \cdots & & B_{n_on_i}(p,\boldsymbol{\theta}) \end{bmatrix}, \tag{1.27}$$

n_i is the number of inputs and n_o is the number of outputs. $A(p,\boldsymbol{\theta})$ and $B_{oi}(p,\boldsymbol{\theta})$ can be described by the following relationships,

$$A(p,\boldsymbol{\theta}) = 1 + \sum_{k=1}^{n_a} a_k p^k \tag{1.28}$$

$$B_{oi}(p,\boldsymbol{\theta}) = \sum_{k=0}^{n_b} b_{oi,k} p^k \tag{1.29}$$

where $i = 1, 2, \ldots, n_i$ and $o = 1, 2, \ldots, n_o$. To simplify the notation, we have made all the series which approximate the transfer functions $B_{oi}(p, \theta)$ the same length, n_b.

In order to set up the least squares problem, we need $n_i \times n_o$ frequency data records $\mathcal{G}_{oi}(\omega)$, where $\mathcal{G}_{oi}(\omega)$ is the frequency data from input i to output o. Setting up the weighted least squares problem associated with 1.18 results in the following equation,

$$0 = \mathbf{W}(\omega)^{\frac{1}{2}} \mathbf{\Phi} \begin{bmatrix} -1 \\ \boldsymbol{\theta} \end{bmatrix}, \tag{1.30}$$

where $\mathbf{W}(\omega)$ is size $(n_i n_o N \times n_i n_o N)$, and $\mathbf{\Phi}$ is size $(n_i n_o N \times (n_a + n_b n_o))$

$$\mathbf{W}(\omega)^{\frac{1}{2}} = \begin{bmatrix} W(\omega)^{\frac{1}{2}} & 0 & \cdots & 0 \\ 0 & W(\omega)^{\frac{1}{2}} & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & W(\omega)^{\frac{1}{2}} \end{bmatrix}$$

$$\mathbf{\Phi} = \begin{bmatrix} [\mathbf{\Phi}_a]_1 & [\mathbf{\Phi}_b] \\ [\mathbf{\Phi}_a]_2 & [\mathbf{\Phi}_b] \\ \vdots & \vdots \\ [\mathbf{\Phi}_a]_{n_o} & [\mathbf{\Phi}_b] \end{bmatrix}$$

$$[\mathbf{\Phi}_a]_o = \begin{bmatrix} \mathbf{\Phi}_{a,o1} \\ \mathbf{\Phi}_{a,o2} \\ \cdots \\ \mathbf{\Phi}_{a,on_i} \end{bmatrix}$$

$$\mathbf{\Phi}_{a,oi} = \begin{bmatrix} \mathcal{G}_{oi}(\omega_0) & \mathcal{G}_{oi}(\omega_0)p(\omega_0) & \cdots & \mathcal{G}_{oi}(\omega_0)p(\omega_0)^{n_a} \\ \mathcal{G}_{oi}(\omega_1) & \mathcal{G}_{oi}(\omega_1)p(\omega_1) & \cdots & \mathcal{G}_{oi}(\omega_1)p(\omega_1)^{n_a} \\ \vdots & \vdots & & \vdots \\ \mathcal{G}_{oi}(\omega_{N-1}) & \mathcal{G}_{oi}(\omega_{N-1})p(\omega_{N-1}) & \cdots & \mathcal{G}_{oi}(\omega_{N-1})p(\omega_{N-1})^{n_a} \end{bmatrix}$$

$$[\mathbf{\Phi}_b] = \begin{bmatrix} \mathbf{\Phi}_b & 0 & \cdots & 0 \\ 0 & \mathbf{\Phi}_b & \ddots & \vdots \\ \vdots & \ddots & \ddots & 0 \\ 0 & \cdots & 0 & \mathbf{\Phi}_b \end{bmatrix}$$

$$\Phi_b = \begin{bmatrix} 1 & p(\omega_0) & \cdots & p(\omega_0)^{n_b} \\ 1 & p(\omega_1) & \cdots & p(\omega_1)^{n_b} \\ \vdots & \vdots & & \vdots \\ 1 & p(\omega_{N-1}) & \cdots & p(\omega_{N-1})^{n_b} \end{bmatrix}$$

$$\theta = \begin{bmatrix} \mathbf{a} \\ \mathbf{b}_{11} \\ \mathbf{b}_{12} \\ \vdots \\ \mathbf{b}_{1n_i} \\ \mathbf{b}_{21} \\ \vdots \\ \mathbf{b}_{n_o n_i} \end{bmatrix}$$

$$\mathbf{a} = \begin{bmatrix} -a_1 & -a_2 & \cdots & -a_{n_a} \end{bmatrix}^T$$

$$\mathbf{b}_{oi} = \begin{bmatrix} b_{oi,0} & b_{oi,1} & \cdots & b_{oi,n_b} \end{bmatrix}^T$$

In the MIMO problem just formulated, it seems to make the most sense to orthonormalize with respect to $W(\omega)^{\frac{1}{2}}\Phi_b$. This would produce the following equations,

$$0 = \mathbf{W}(\omega)^{\frac{1}{2}} \begin{bmatrix} [\Phi_a]_1 & [\Phi_b] \\ [\Phi_a]_2 & [\Phi_b] \\ \vdots & \vdots \\ [\Phi_a]_{n_o} & [\Phi_b] \end{bmatrix} [\mathbf{T}][\mathbf{T}^{-1}] \begin{bmatrix} -1 \\ \theta \end{bmatrix}$$

$$0 = \mathbf{W}(\omega)^{\frac{1}{2}} \begin{bmatrix} [\bar{\bar{\Phi}}_a]_1 & [\hat{\Phi}_b] \\ [\bar{\bar{\Phi}}_a]_2 & [\hat{\Phi}_b] \\ \vdots & \vdots \\ [\bar{\bar{\Phi}}_a]_{n_o} & [\hat{\Phi}_b] \end{bmatrix} \dot{\theta} \begin{bmatrix} c \\ \hat{\theta} \end{bmatrix}$$

$$0 = \mathbf{W}(\omega)^{\frac{1}{2}} \hat{\Phi} \hat{\theta} \begin{bmatrix} c \\ \hat{\theta} \end{bmatrix}$$

38

where

$$[\mathbf{T}] = \begin{bmatrix} \mathbf{T} & \mathbf{0} & \cdots & \mathbf{0} \\ \mathbf{0} & \mathbf{T} & \ddots & \vdots \\ \vdots & \ddots & \ddots & \mathbf{0} \\ \mathbf{0} & \cdots & \mathbf{0} & \mathbf{T} \end{bmatrix}$$

$\left[\bar{\mathbf{\Phi}}_a\right]_o$ is not orthogonal and $[\hat{\mathbf{\Phi}}_b]$ is orthonormal. So, like the SISO case, we need to solve the least squares problem

$$c\bar{\mathbf{\Phi}}_a(:,1) = \begin{bmatrix} \bar{\mathbf{\Phi}}_a(:,2:n_a) & \hat{\mathbf{\Phi}}_b \end{bmatrix} \hat{\boldsymbol{\theta}}, \tag{1.31}$$

for $\hat{\boldsymbol{\theta}}$ and then convert to the original domain using $[\mathbf{T}]$.

1.8 Conclusions for Part I

This report outlines a general method for doing system identification of linear time–invariant systems based on Gram–Schmidt orthonormalization. The advantage of this method is that it allows one to orthonormalize in any domain with any weighting.

System identification using Gram–Schmidt orthonormalization achieved significant improvement in the identification of models in the s domain, as compared to the more standard methods. The use of Gram–Schmidt orthonormalization in the z domain showed no improvement, and in the δ domain we encountered problems in applying the method. More research is necessary to fully evaluate the benefits and difficulties associated with this method of system identification.

1.9 Appendix

1.9.1 Orthogonality of Discrete Exponential Function

Claim: The discrete function $e^{j\boldsymbol{\omega}T}$, where $\boldsymbol{\omega} = \begin{bmatrix} \omega_0 & \omega_1 & \cdots & \omega_{N-1} \end{bmatrix}^T$ and $\omega_\ell \triangleq \frac{2\pi\ell}{TN}$, is orthogonal with unit weighting, $w(\omega_\ell) = 1$.

Proof: The discrete function $e^{j\omega T}$ is orthogonal with unit weighting if the following holds,

$$S_{mk} = (e^{j\omega Tm})^H e^{j\omega Tk} = \begin{cases} h_m & , \ k = m \\ 0 & , \ k \neq m \end{cases}. \tag{1.32}$$

where h_m is a real scalar and $0 < h_m < \infty$. Changing from vector notation to summation notation,

$$S_{mk} = \sum_{\ell=0}^{N-1} e^{-j\omega_\ell Tm} e^{j\omega_\ell Tk} \tag{1.33}$$

$$= \sum_{\ell=0}^{N-1} e^{j\omega_\ell Tq} \tag{1.34}$$

where $q = k - m$. At this point, it is obvious that if $k = m$ ($q = 0$), then $S_{mk} = N+1$ ($h_m = N+1$). Now, we must show that if $k \neq m$, then $S_{mk} = 0$. This can be shown as follows: Expand 1.34

$$S_{mk} = 1 + e^{j\omega_1 Tq} + \cdots + e^{j\omega_{N-2} Tq} + e^{j\omega_{N-1} Tq},$$

then multiply by $e^{j\omega_1 Tq}$,

$$e^{j\omega_1 Tq} S_{mk} = e^{j\omega_1 Tq} + e^{j(\omega_1+\omega_1)Tq} + \cdots + e^{j(\omega_1+\omega_{N-2})Tq} + e^{j(\omega_1+\omega_{N-1})Tq}.$$

Notice that $\omega_1 + \omega_\ell = \omega_{\ell+1}$, and that $\omega_N = \omega_0 = 0 \ rad$, so that

$$e^{j\omega_1 Tq} S_{mk} = e^{j\omega_1 Tq} + e^{j\omega_2 Tq} + \cdots + e^{j\omega_{N-1} Tq} + 1 = S_{mk}.$$

Since $e^{j\omega_1 Tq} S_{mk} = S_{mk}$, and $e^{j\omega_1 Tq} \neq 1$ for $q \neq 0$, this implies that $S_{mk} = 0$ for $q \neq 0$.

Part II

Chapter 2

Disturbance Rejection Using H_2 and H_∞ Control Design

Abstract

Wide and narrow band disturbance rejection is explored using H_2 and H_∞ control design. This design method allows for multiple feedback design, therefore, the advantages of using primary and secondary feedback measurements verses only primary feedback measurements for disturbance rejection are also discussed. The primary measurement is a measurement of the signal which is to be controlled, and the secondary measurement is a measurement which is closely correlated to the disturbance.

2.1   Introduction

This report discusses research disturbance rejection using H_2 and H_∞ control design. First, the design procedure is developed. In this procedure both the inputs and the outputs of the closed loop system are weighted. Weighting the inputs corresponds to including a priori information in the system for control design. Weighting the outputs corresponds to enforcing the performance criteria. Examples are then used to illustrated the method.

   Also, the use of primary and secondary feedback measurements verses only primary feedback measurements is explored. Primary measurements correspond to outputs one wishes to control, while secondary measurements

correspond to outputs one does not care to control but which are highly correlated to the disturbance. The motivation for using secondary measurements is that the more information the controller has, the better the control should be. For instance, in disturbance rejection one obvious measurement would be the output of the sensor corresponding to the quantity which is being minimized. This is called the primary output. This may be a force sensor at the location where it is desired to have zero force. Another output which is desired is the disturbance itself. This is not usually a practical measurement, but perhaps it is possible to place a sensor "near by" the disturbance, so that this output is well correlated with the disturbance. This is called the secondary output. An example is included which illustrates the advantage of using a primary (controlled) output and a secondary (disturbance) output as feedback.

2.2 General Discussion

First, notation and the problem framework are defined.

2.2.1 Notation and Problem Framework

Begin by describing the system in figure 2.1, where P is the open loop system, K is the controller, w is the exogenous input, u is the controlled input, z is the controlled output, and y is the sensor output which is fed back to the controller. P and K are matrix transfer functions, while w, u, z and y are vectors. Note that in general w contains all exogenous inputs (i.e. reference inputs, disturbances, sensor noise, process noise), and z contains all controlled outputs for the performance criteria. Therefore, z may include the controlled input u if one of the design criteria is to limit u in some way. P can be written as follows,

$$P = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix}$$

where P_{11} corresponds to the open loop transfer function matrix from w to z, and so forth. Thus, the system equations are the following,
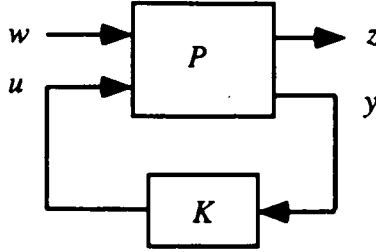
$$z \quad = \quad P_{11}w + P_{12}u$$

Figure 2.1: Closed Loop System

$$y = P_{21}w + P_{22}u$$
$$u = Ky.$$

The closed loop transfer function matrix T from the exogenous input w to the controlled output z is given by,

$$T = P_{11} + P_{12}K(I - P_{22}K)^{-1}P_{21}. \tag{2.1}$$

The H_2 control design finds the stabilizing controller K which minimizes the *2-norm* of T (i.e. $min_K\|T\|_2$).

The H_∞ control design finds a stabilizing controller K such that the ∞-*norm* of T is less than or equal to one (i.e. $\|T\|_\infty \le 1$).

In order to include a priori information concerning the exogenous inputs and to enforce the design criteria, input and output weightings are added to the system.

2.2.2 Input and Output Weightings

Weightings can be added to the system shown in figure 2.1. These weightings are transfer functions, and can be used to incorporate a priori knowledge into the system and to enforce design criteria. The placement of the weightings is shown in figure 2.2, where W_i is the input weighting matrix and W_o is the output weighting matrix. W_i and W_o are diagonal transfer function matrices. Now, the system equations are the following,

$$W_o^{-1}\hat{z} = P_{11}W_i\hat{w} + P_{12}u$$
$$y = P_{21}W_i\hat{w} + P_{22}u$$
$$u = Ky.$$

44
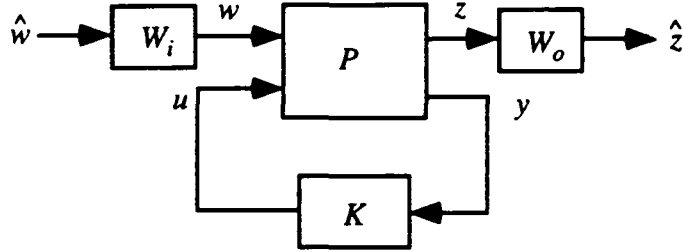
Figure 2.2: Closed Loop System with Weightings

Therefore, the weighted closed loop transfer function matrix $\hat{T}$ from the disturbance $\hat{w}$ to the weighted controlled output $\hat{z}$ is given by,

$$\hat{T} = W_o \left(P_{11} + P_{12}K(I - P_{22}K)^{-1}P_{21} \right) W_i = W_o T W_i. \qquad (2.2)$$

Following is a discussion of the roles of the input and output weightings.

Input Weighting Matrix

The input weighting matrix is used to incorporate a priori knowledge of the inputs into the system. For instance, if we know that the sensor noise is below some value, then we would use that value as our weighting for the input which corresponds to the sensor noise. For example, if sensor noise is given as in figure 2.3, then a transfer function is used which closely resembles this as the weighting on the sensor noise input, in order to include this information in the design. The reason to use these weightings is that the H_2 and H_∞ designs work to produce a controller K such that $\hat{T}$ is minimized in some respect. So, in a rough sense the H_2 and H_∞ designs assume that the exogenous inputs w are full spectrum with a magnitude of one, so that the outputs are given by the transfer functions in $\hat{T}$. Therefore, the input weightings are used to incorporate what we know about the inputs into the system. Examples are given in section 2.3 of this report.

Output Weighting Matrix

The output weighting matrix would be used to enforce design criteria on the closed loop system. For instance, in the case of disturbance rejection where the system output z is the plant output, and it is desired to minimize the

45
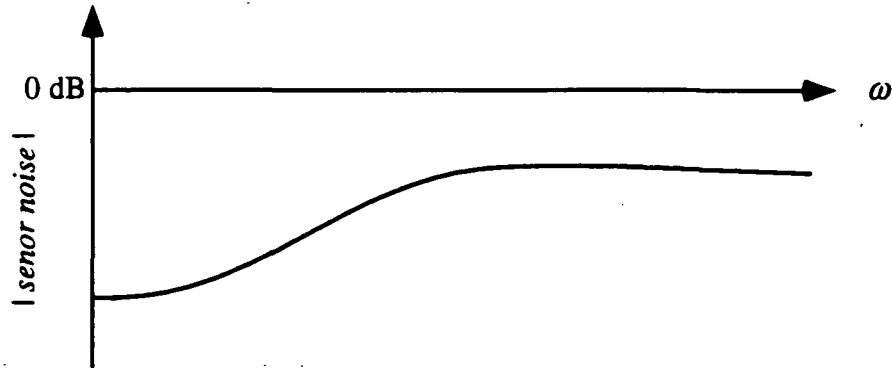
Figure 2.3: Example of Sensor Noise Spectrum

output at high frequencies, but the system can tolerate larger outputs at low frequencies, then the weighting shown in figure 2.4 may be appropriate. This
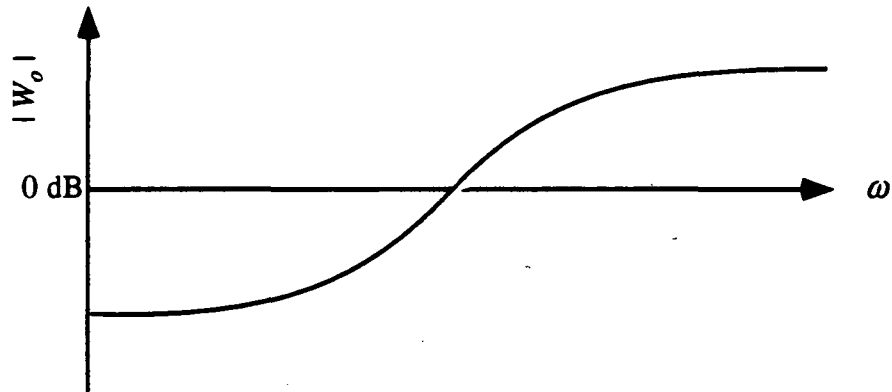

Figure 2.4: Example of Output Weighting Spectrum

weighting forces the unweighted closed loop transfer function T to be small where W_o is large, because the control design is done on the weighted system $\hat{T} = W_o T W_i$. Therefore, weighting the output corresponds to enforcing the design criteria.

2.3 Examples

The test model is a one mass model with one control input and two sensors. The model is developed and then used to illustrate the design method. The designs are for rejection of two types of noise, wide band and narrow band disturbances.

2.3.1 Test Model

The one mass model is shown in figure 2.5. The constants used in the model
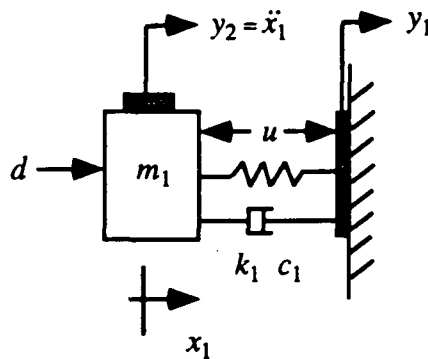
Figure 2.5: One Mass Model

are given in table 2.1. The damping coefficients correspond to 2% damping of the modes, and the outputs (y_1, y_2) correspond to the force at the wall and the acceleration of the mass. The sampling interval is $T = 0.001$. The

$$\begin{array}{|l|}
\hline
m_1 = 2.18\,kg \\
k_1 = 14\,N/mm \\
c_1 = 6.9880\,kg/s \\
\hline
\end{array}$$

Table 2.1: Values Used in One Mass Model

block diagram for this system is shown in figure 2.6. The dependence of the transfer function on s, the Laplace variable, is not stated explicitly to keep the notation cleaner. A bode plot of the plant transfer function from the disturbance d to the controlled output y_1 is shown in figure 2.7.

47
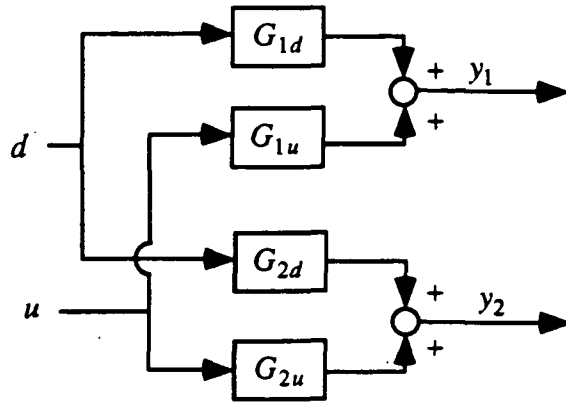
Figure 2.6: Open Loop System
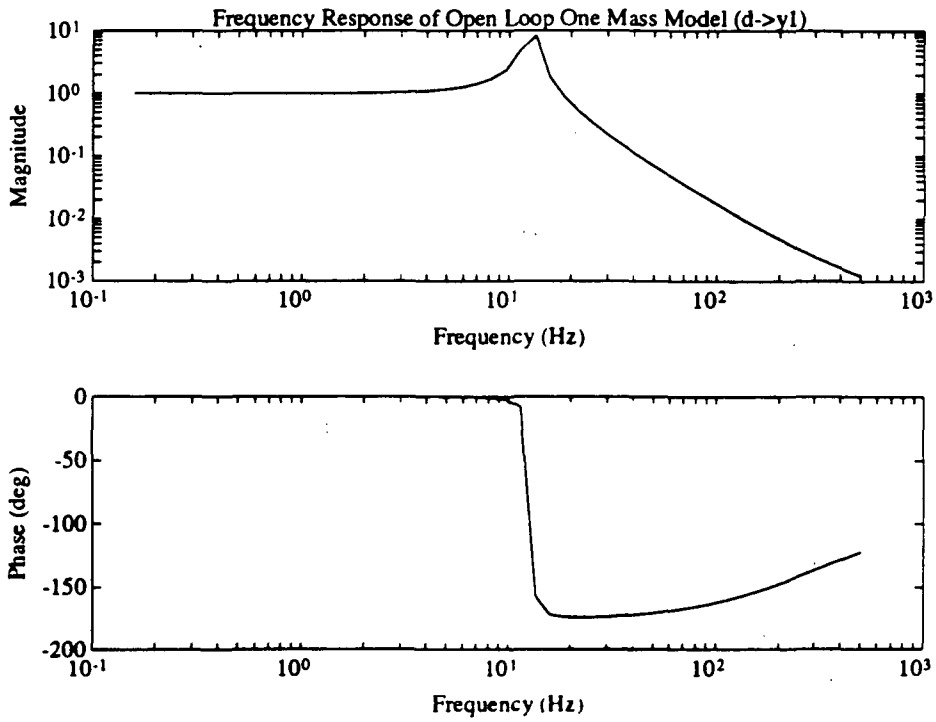
Figure 2.7: One Mass Model Bode Plot

The complete closed loop system block diagram is shown in figure 2.8. This system includes the input and output weightings, a delay in the response
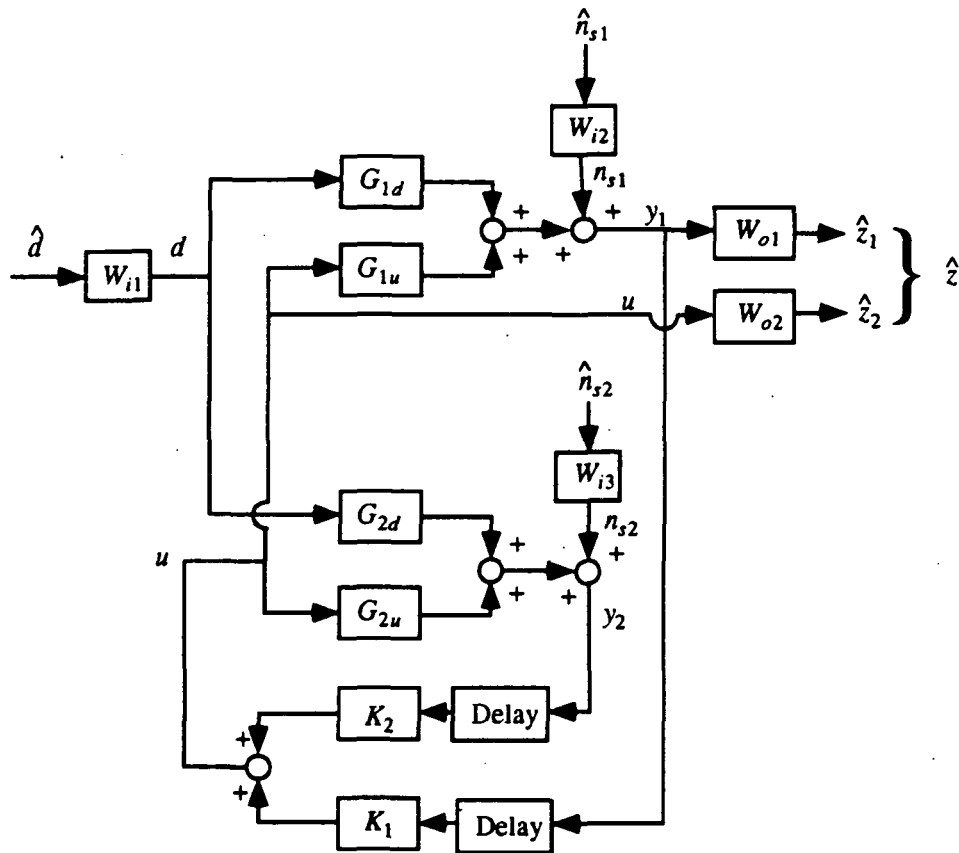

Figure 2.8: Closed Loop System Including Weightings

of the controller, and sensor noise. It does not include process noise at this point. The controlled outputs are y_1 and u. So, corresponding to the earlier system description P, K, $\hat{w}$, w, u, $\hat{z}$, z and y are given as follows

$$P_{11} = \begin{bmatrix} G_{1d} & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$P_{12} = \begin{bmatrix} G_{1u} \\ 1 \end{bmatrix}$$

$$P_{21} = \begin{bmatrix} G_{1d} & 1 & 0 \\ G_{2d} & 0 & 1 \end{bmatrix}$$

$$P_{22} = \begin{bmatrix} G_{1u} \\ G_{2u} \end{bmatrix}$$

$$K = \begin{bmatrix} D(T_d)K_1 & D(T_d)K_2 \end{bmatrix}$$

$$\hat{w} = \begin{bmatrix} \hat{d} \\ \hat{n}_{s1} \\ \hat{n}_{s2} \end{bmatrix}$$

$$w = W_i \begin{bmatrix} \hat{d} \\ \hat{n}_{s1} \\ \hat{n}_{s2} \end{bmatrix} = \begin{bmatrix} d \\ n_{s1} \\ n_{s2} \end{bmatrix}$$

$$u = u$$

$$\hat{z} = W_o \begin{bmatrix} y_1 \\ u \end{bmatrix}$$

$$z = \begin{bmatrix} y_1 \\ u \end{bmatrix}$$

$$y = \begin{bmatrix} y_1 \\ y_2 \end{bmatrix}$$

where $D(T_d)$ is a delay of time T_d, and W_i and W_o are diagonal transfer function matrices containing the appropriate weightings. By absorbing the weightings and the delays into the open loop plant P, a new open loop system $\hat{P}$ is formed given by the following

$$\hat{P}_{11} = \begin{bmatrix} W_{o1}G_{1d}W_{i1} & W_{o1}W_{i2} & 0 \\ 0 & 0 & 0 \end{bmatrix}$$

$$\hat{P}_{12} = \begin{bmatrix} W_{o1}G_{1u} \\ W_{o2} \end{bmatrix}$$

$$\hat{P}_{21} = \begin{bmatrix} D(T_d)G_{1d} & D(T_d) & 0 \\ D(T_d)G_{2d} & 0 & D(T_d) \end{bmatrix}$$

$$\hat{P}_{22} = \begin{bmatrix} D(T_d)G_{1u} \\ D(T_d)G_{2u} \end{bmatrix}$$

$$\hat{K} = \begin{bmatrix} K_1 & K_2 \end{bmatrix}.$$

2.3.2 H_2 and H_∞ Control Design Tests

Two different tests are run. The first is a demonstration of wide band disturbance rejection, and the second is a demonstration of narrow band disturbance rejection. The tests are run using the one mass model described previously. And, the delay, $D(T_d)$, is approximated using a second order Padé approximation, where $T_d = 0.001$. The tests are run using the Matlab Robust-Control Toolbox [3].

Wide Band Disturbance Rejection Using H_2 and H_∞ Control design

H_2 Control Design for Wide Band Disturbance Rejection The first step is to set up the input and output weightings. These are shown in figure 2.9. The input weightings were all chosen to be constant transfer functions
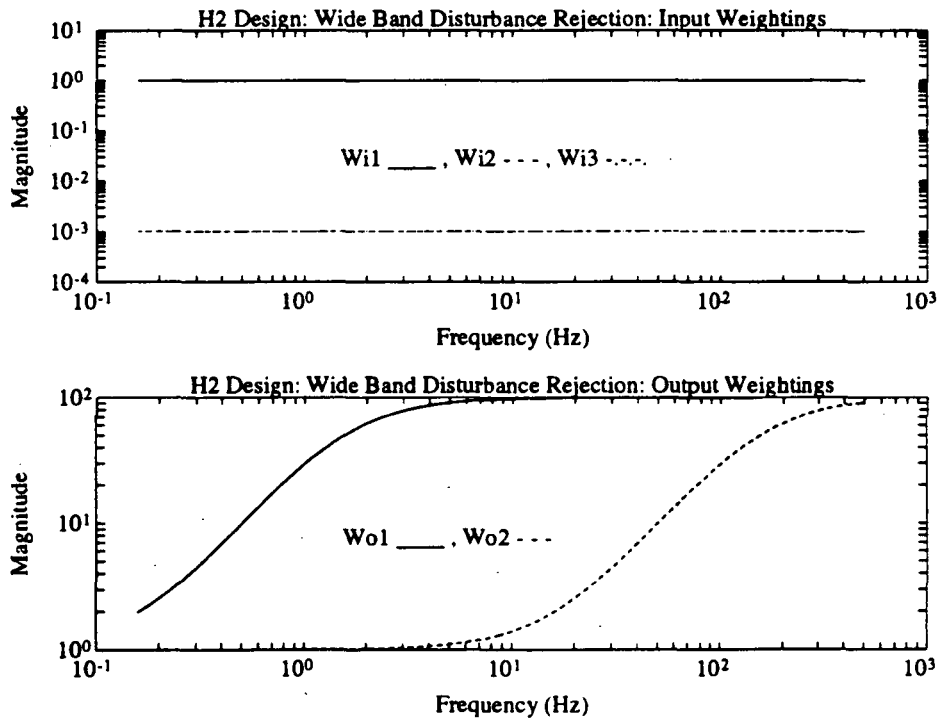
Figure 2.9: Input and Output Weightings for H_2 Design Wide Band Disturbance Rejection

(no dependence on frequency). The weighting on the disturbance $\hat{d}$ is 1, while the weighting on the two sensor noises n_{s1} and n_{s2} is equal and is 0.001. This corresponds to most of the noise in the system being attributed to the disturbance, and not the sensor noise. The output weightings correspond to a design criteria which demands better disturbance rejection at higher frequencies, and less control force at higher frequencies.

The closed loop and open loop frequency response is shown for comparison in figure 2.10. At high frequencies the disturbance rejection is quite good,
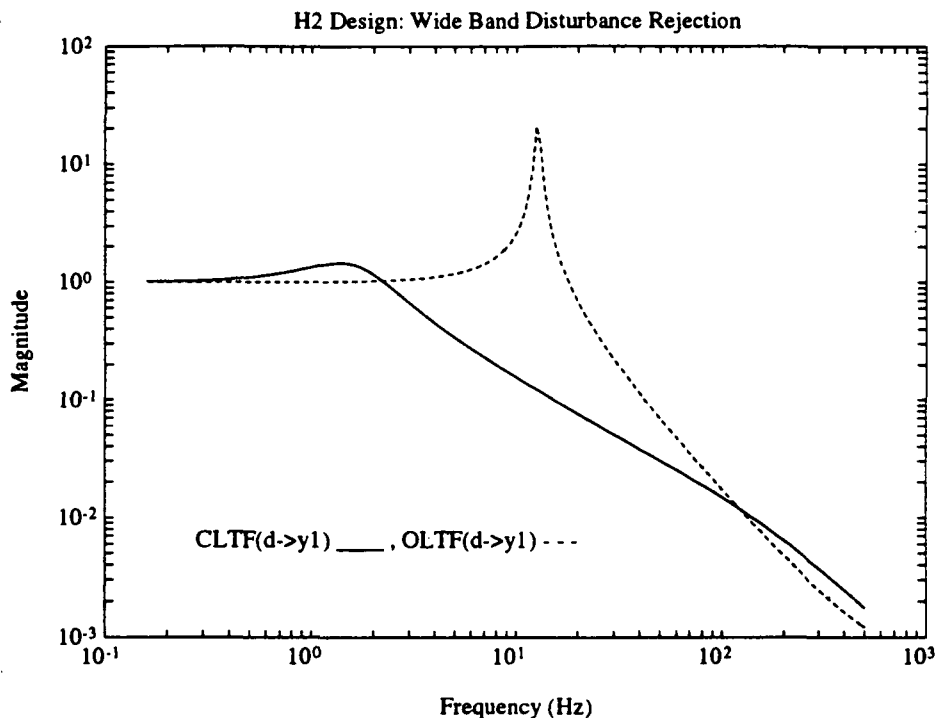
Figure 2.10: Comparison of Open Loop and Closed Loop Transfer Functions for H_2 Design Wide Band Disturbance Rejection

and not as good in low frequencies. The poor results at low frequencies can be explained by examining the model (figure 2.5). For a constant disturbance force d, the control force u would have to be infinite in order to have y_1 equal to zero. Since the control force u is part of the performance measure, by being included in z, solutions which rely on unrealistic control forces are

ruled out. The closed loop transfer functions from the sensor noises to the output y_1 were inspected to insure that they contribute very little to the output y_1 when compared to the effect of the disturbance d.

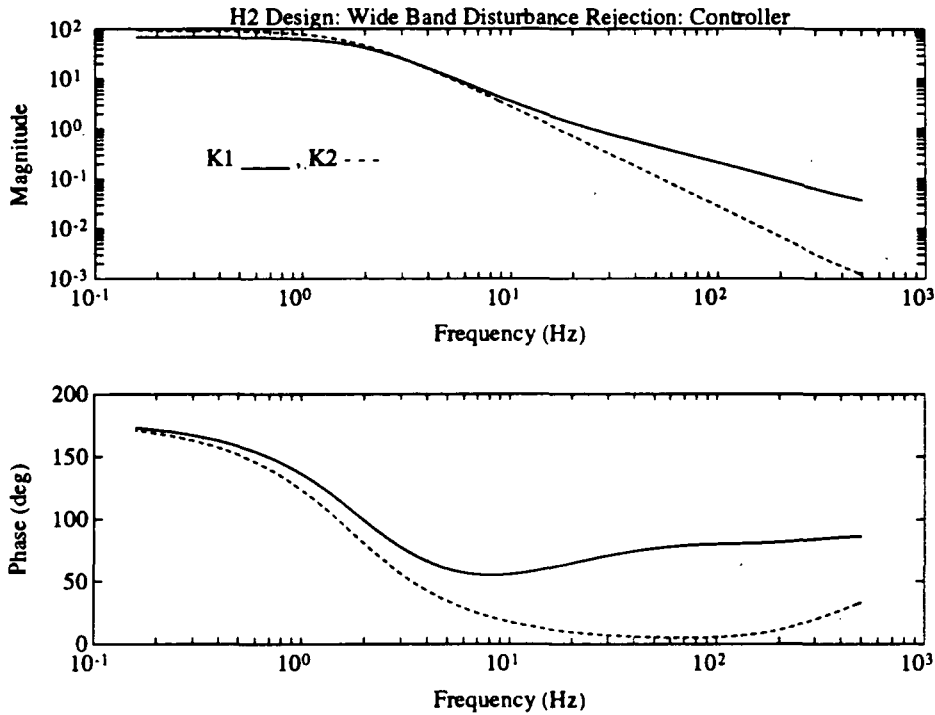A frequency response plot of the controller is shown in figure 2.11.

Figure 2.11: Controller Transfer Function for H_2 Design Wide Band Disturbance Rejection

H_∞ **Control Design for Wide Band Disturbance Rejection** Again, the first step is to set up the input and output weightings. The same input weightings are used from the H_2 design case, but the output weightings are scaled so that a H_∞ design solution can be found. The scaled output weights are shown in figure 2.12. It was possible to find a solution without changing the weighting on the controlled input u, but the weighting on y_1 had to be so relaxed that the solution was to use a very low gain controller. This corresponds to essentially disconnecting the controller. Also, the weighting
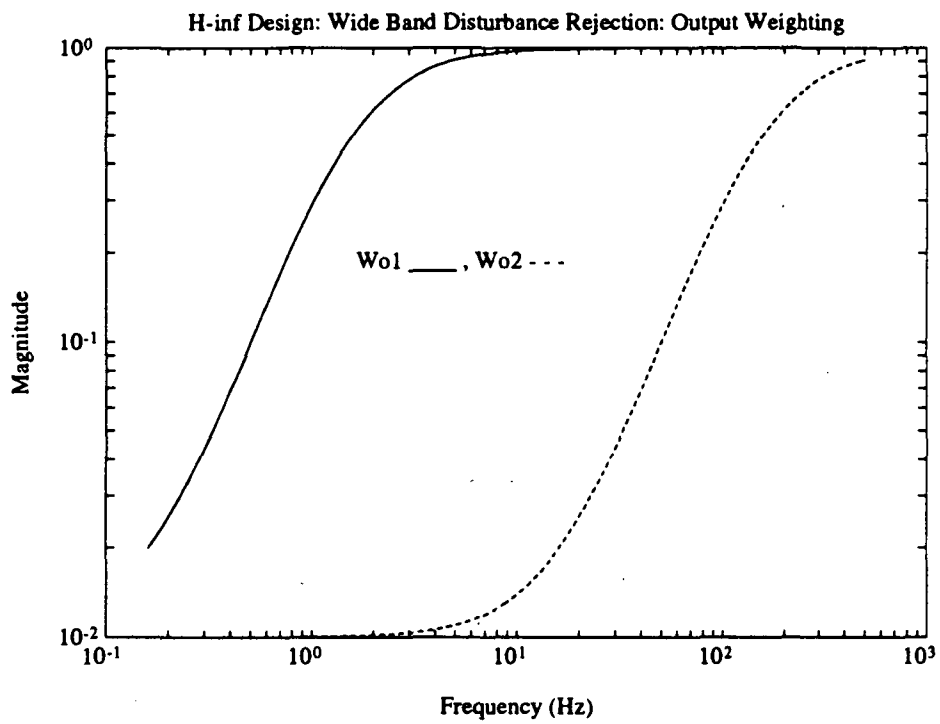
Figure 2.12: Output Weightings for H_∞ Design Wide Band Disturbance Rejection

on y_1 could be raised, but at the expense of having to decrease the controlled input weighting, which corresponds to allowing an even greater control force. The weighting used for W_{o2}, the weighting on u, allows the H_∞ design to have a much greater controlled input than the H_2 design. This is not desirable.

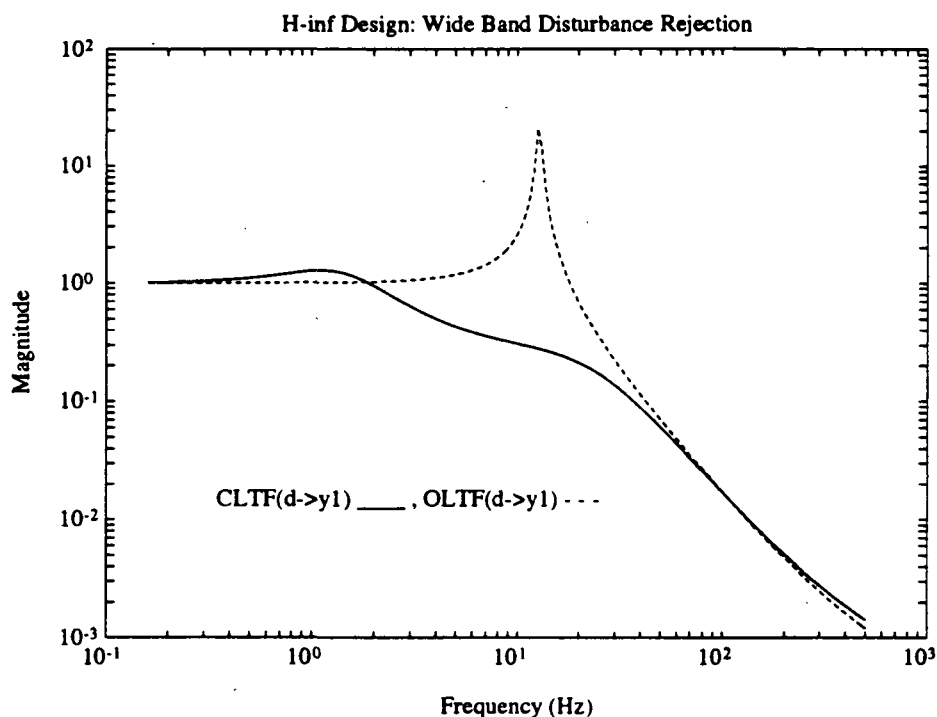The closed loop and open loop frequency response is shown for comparison in figure 2.13.

Figure 2.13: Comparison of Open Loop and Closed Loop Transfer Functions for H_∞ Design Wide Band Disturbance Rejection

A frequency response plot of the controller is shown in figure 2.14.

Narrow Band Disturbance Rejection Using H_2 and H_∞ Control design

H_2 Control Design for Narrow Band Disturbance Rejection The input and output weightings are the same as those for the H_2 design for wide
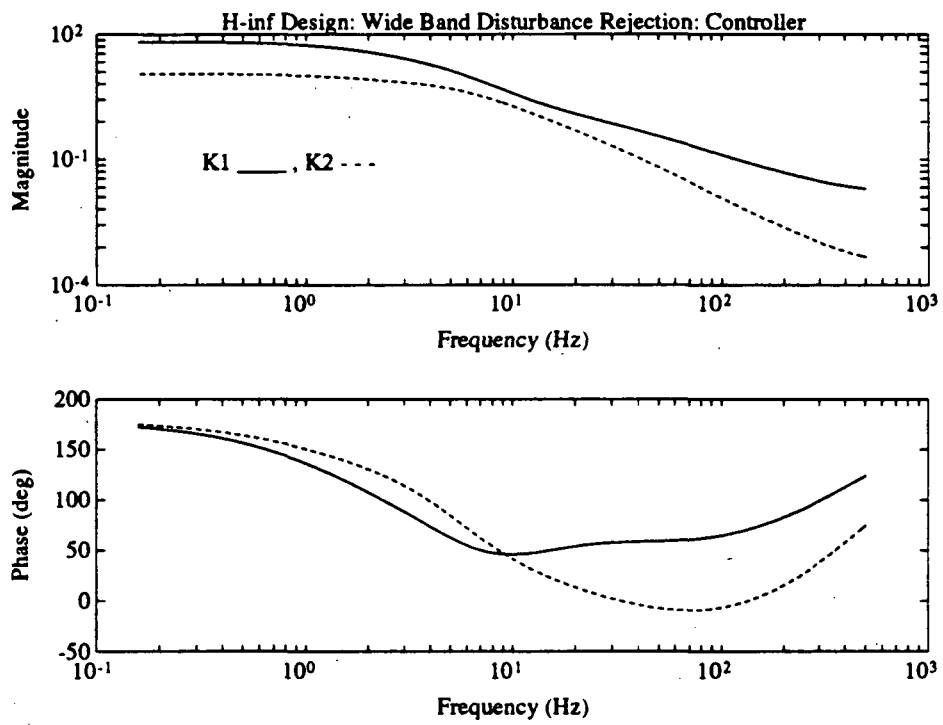
Figure 2.14: Controller Transfer Function for H_∞ Design Wide Band Disturbance Rejection

band disturbance rejection except for the input weighting on the disturbance $\hat{d}$. The input weighting on the disturbance $\hat{d}$ is formed to account for a narrow band disturbance at 10 Hz, and broad band low level noise. The weightings are shown in figure 2.15.
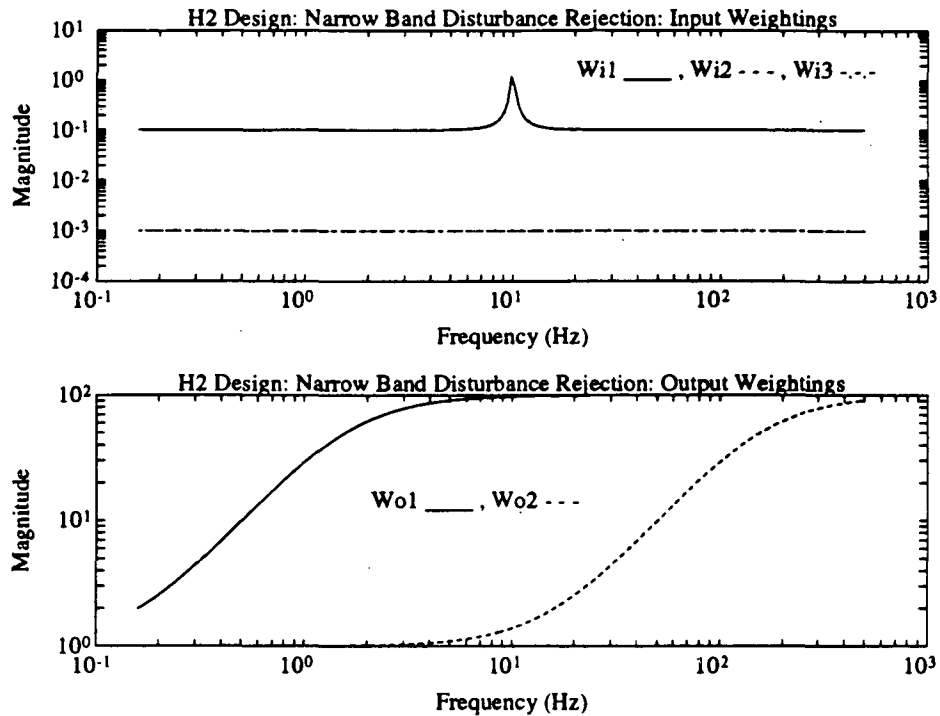

Figure 2.15: Input and Output Weightings for H_2 Design Narrow Band Disturbance Rejection

The closed loop and open loop frequency responses are shown for comparison in figure 2.16.

A frequency response plot of the controller is shown in figure 2.17. The weighting of the disturbance causes the controller to have an inverted notch at 10 Hz. This is difficult to see on the magnitude plot, but is more apparent on the phase plot. This result can be explained by the Internal Model Principle [9], where the dynamics of the disturbance is included in the controller, so that in the closed loop system the dynamics of the disturbance are canceled by those of the controller.
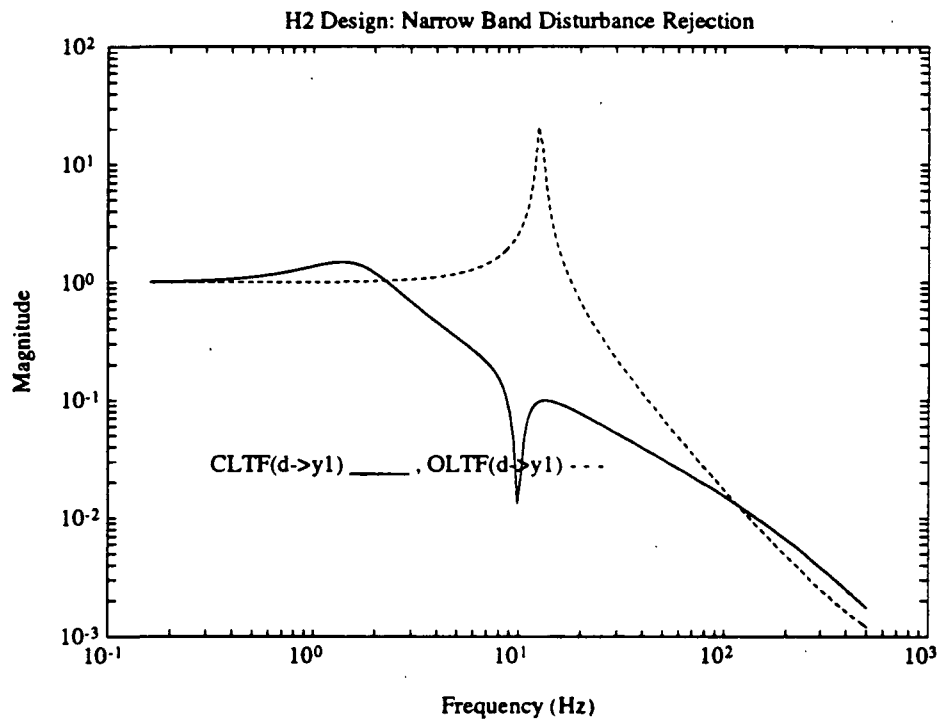
Figure 2.16: Comparison of Open Loop and Closed Loop Transfer Functions
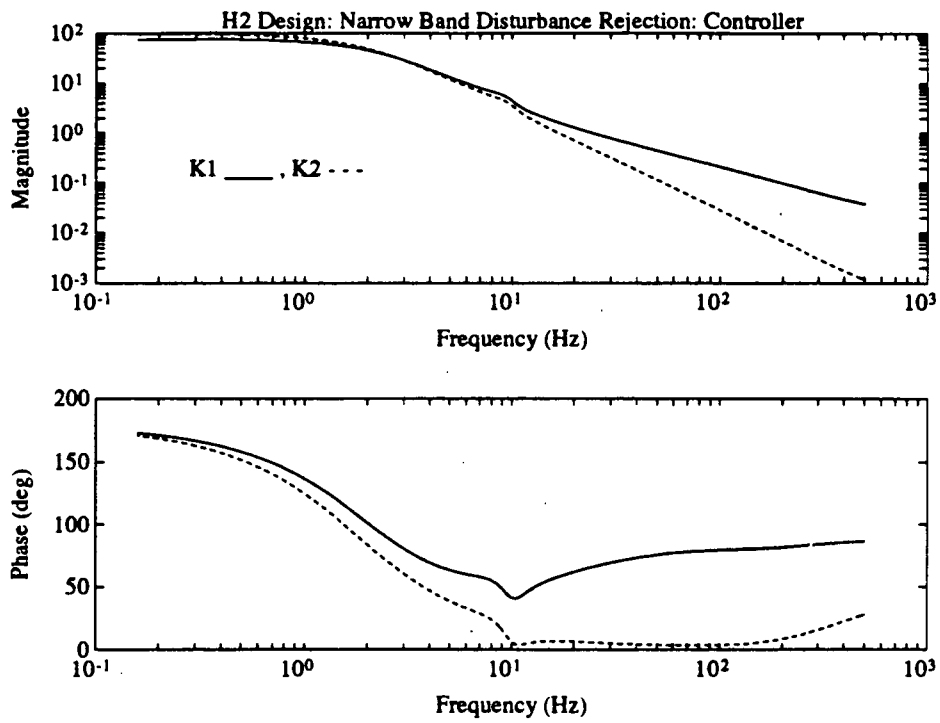for H_2 Design Narrow Band Disturbance Rejection

Figure 2.17: Controller Transfer Function for H_2 Design Narrow Band Disturbance Rejection

H_∞ **Control Design for Narrow Band Disturbance Rejection** The input weightings are the same as those for the H_2 design for narrow band disturbance rejection. Again, the output weightings are scaled so that a H_∞ design solution can be found. The scaled output weights are shown in figure 2.18. As with the wide band case, it was possible to find a solution without
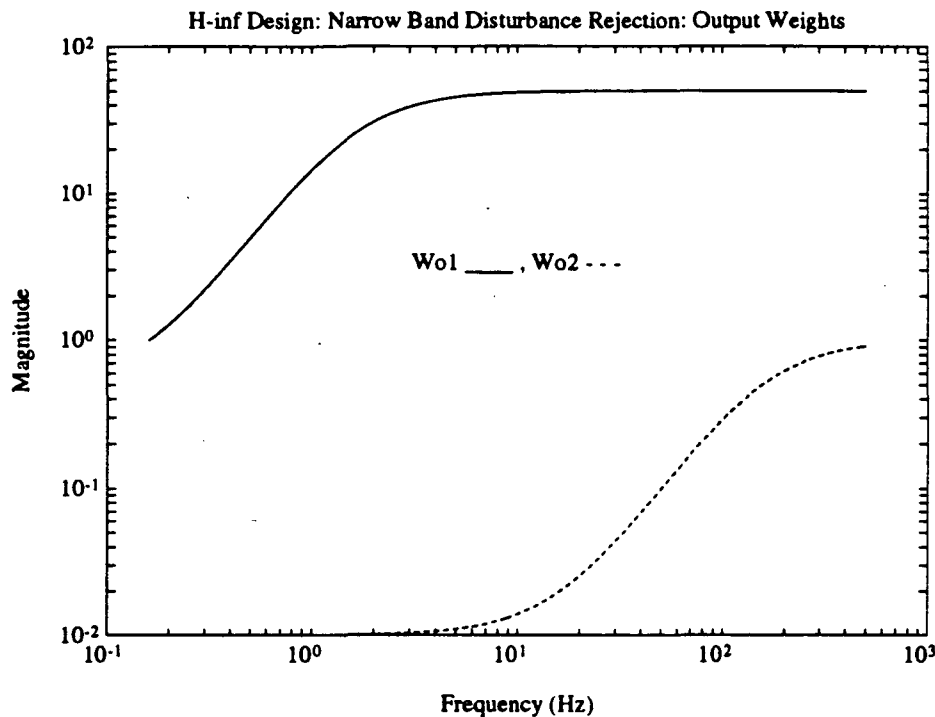

Figure 2.18: Output Weightings for H_∞ Design Narrow Band Disturbance Rejection

changing the weighting on the controlled input u, but the weighting on y_1 had to be so relaxed that the solution was to use a very low gain controller. Also, the weighting on y_1 could be raised, but at the expense of having to decrease the controlled input weighting. The result shown here is a compromise of these two situations.

The closed loop and open loop frequency response is shown for comparison in figure 2.19.

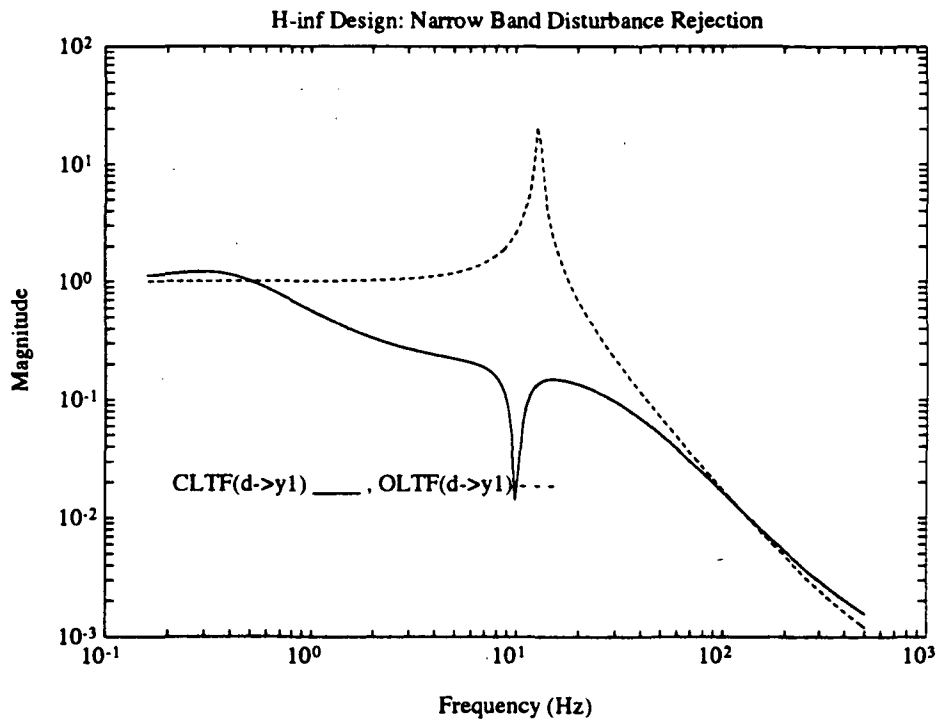A frequency response plot of the controller is shown in figure 2.20.

60

Figure 2.19: Comparison of Open Loop and Closed Loop Transfer Functions for H_∞ Design Narrow Band Disturbance Rejection
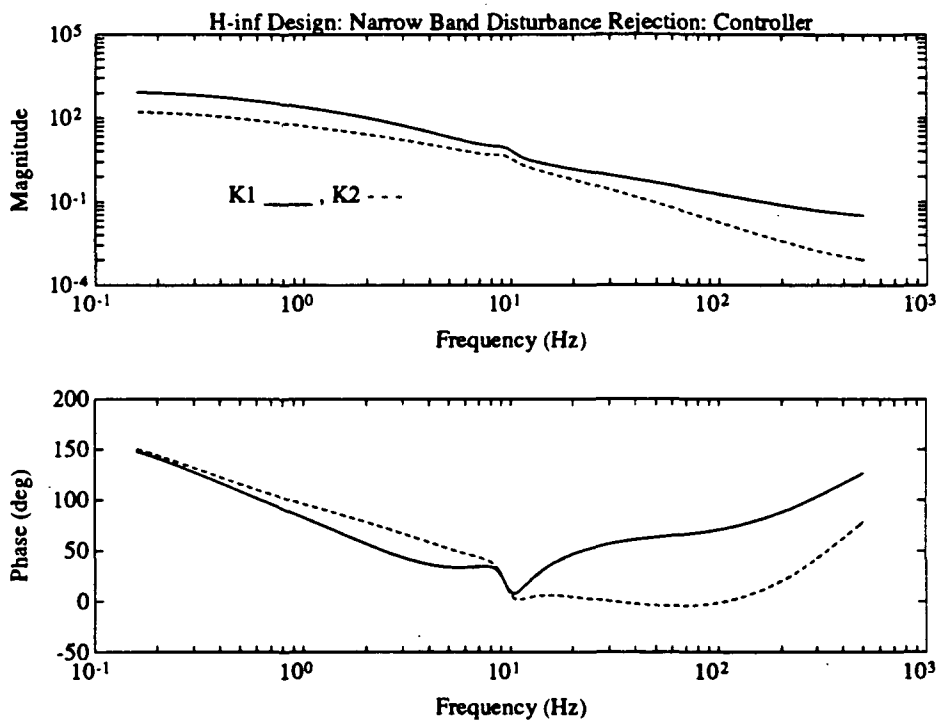
Figure 2.20: Controller Transfer Function for H_∞ Design Narrow Band Disturbance Rejection

2.4 Primary and Secondary Feedback vs Primary Feedback Control

This section explores the advantages of using primary and secondary feedback as opposed to only primary feedback. The primary feedback system is equivalent to setting K_2 equivalent to zero in figure 2.8. With the primary feedback system, the fed back quantity, y_1, is the same quantity which the controller is trying to make zero. This presents a fundamental problem since the controller would "stop working" if y_1 were actually driven to zero. For instance, in the case of the disturbance d being a single sine wave at frequency 10 Hz, the primary feedback system must have a greater gain in the feedback control in order to obtain the same level of performance as the primary plus secondary feedback system, because the two feedback system does not only rely on the measurement which the system is trying to drive to zero, y_1, but also a measurement correlated to the disturbance, namely y_2. Following is an example of this.

The following example uses the input and output weightings of the H_2 design narrow band disturbance rejection example of the previous section. First, the plot of the closed loop frequency response from the disturbance d to the output y_1 of both the primary and primary plus secondary feedback systems is shown in figure 2.21, to confirm that the performance levels are comparable. The control transfer functions are shown in figure 2.22. These controllers were designed for a narrow band disturbance acting at 10 Hz. The primary feedback controller is obviously at a much higher gain than the primary plus secondary feedback controller. The disadvantage of a high gain controller is that its performance is greatly effected by noise in the system.

2.5 Conclusions for Part II

A method for design of controllers for both wide and narrow band disturbance rejection has been developed. This procedure allows the designer to incorporate a priori knowledge concerning the exogenous inputs, and to enforce design criteria in a systematic manner, thereby removing much of the "guess work" present in classical design methods. These designs utilized the H_2 and H_∞ norms by using the Matlab Robust-Control Toolbox.
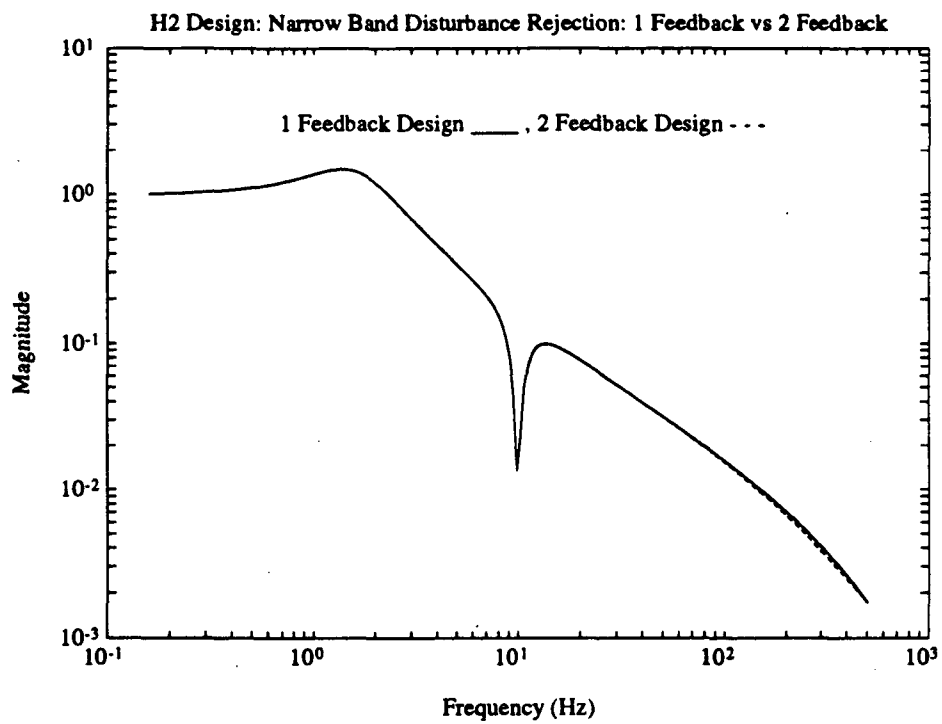
Figure 2.21: Comparison of Performance for the Primary (1) Feedback System and the Primary Plus Secondary (2) Feedback System
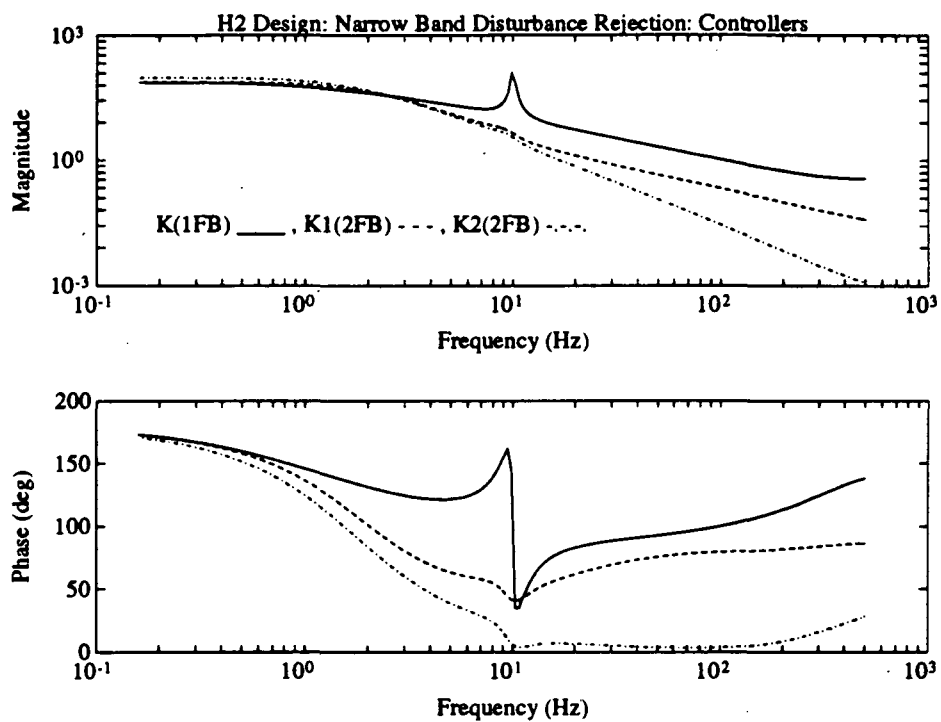
Figure 2.22: Comparison of Controllers for the Primary (1) Feedback System and the Primary Plus Secondary (2) Feedback System

65

Also, the advantages of using a primary plus secondary feedback in the closed loop system for disturbance rejection was explored. The primary feedback corresponds to the output which the control is attempting to control, while the secondary feedback corresponds to the output which is correlated to the disturbance. It was shown that a lower gain controller can be found when feeding back both primary and secondary outputs, as compared with feeding back only the primary output.

Bibliography

[1] D. S. Bayard. Multivariable state-space identification in the delta and shift operators: Algorithms and experimental results. In *American Control Conference*, June 1993.

[2] S. P. Boyd and C. H. Barratt. *Linear Controller Design: Limits of Performance*. Prentice Hall, 1991.

[3] R. Y. Chiang and M. G. Safonov. *Robust-Control Toolbox*. The Math Works Inc., 1988.

[4] T. S. Chihara. *An Introduction to Orthogonal Polynomials*. Gordon and Breach, 1978.

[5] R. L. Dailey. Lecture notes for the workshop on h_∞ and μ methods in robust control. 1990 American Control Conference, May 1990.

[6] R. L. Dailey and M. S. Lukich. Mimo transfer function curve fitting using chebyshev polynomials. In *SIAM 35th Anniversary Meeting*, Denver, CO, 1987.

[7] G. Freud. *Orthogonal Polynomials*. Akadémiai Kiadó, 1971.

[8] L. Ljung. *System Identification: Theory for the User*. Prentice Hall, 1987.

[9] R. H. Middleton and G. C. Goodwin. *Digital Control and Estimation: A Unified Approach*. Prentice Hall, 1990.

[10] Smart Structures and Intelligent Systems. *Active Vibration Isolation on an Experimental Flexible Structure*, 1993.