

Temperature Dependent Robust Control of Hard Disk Drives
Using Parameter Varying Techniques

A DISSERTATION
SUBMITTED TO THE FACULTY OF THE GRADUATE SCHOOL
OF THE UNIVERSITY OF MINNESOTA
BY

Masanori Honda

IN PARTIAL FULFILLMENT OF THE REQUIREMENTS
FOR THE DEGREE OF
Doctor of Philosophy

Peter J. Seiler, advisor

October 2016

© Masanori Honda 2016

Acknowledgments

I am grateful to numerous people who have supported me over the last several years. First I would like to thank my adviser, Professor Peter Seiler, for giving me the opportunity to work on control theory research after my Master's in thermochemistry. I greatly appreciated the efforts he made to make time for all of his students even when he was very busy. He gives everyone a chance to work with him and is always open to new ideas. I would also like to thank Professor Jane Davidson for giving me an opportunity at her solar energy lab. Without her guidance, I would've never had the courage to transfer to a new department after my Master's.

I would also like to thank Seagate and its liaisons, Raye Sosseh, Chiyun Xia, and Bin Huang. Being able to apply robust control research onto a real system with such high performance has been an exciting research for me. Without the help of my liaisons, I would have not been able to become familiar with the hard disk drives nearly as fast. Furthermore without their sponsorship through the MINT organization, none of this would have been possible.

Finally I'd like to thank my wife, Kate, for her support at home. Kate has been extremely patient and understanding with me in the final months of completing my thesis. Not only has she been a great partner during this time, she has also been a wonderful mother to our daughter, Naomi.

To my lovely wife, Kate.

Abstract

A hard disk drive (HDD) is a device that stores digital data by writing and reading magnetic signals onto a disk using a magnetic transducer. The data is organized into circular tracks that are less than 100 nm wide. Modern HDDs use a dual-state actuator to position the transducer onto these tracks. Since the tracks are extremely narrow, a high performance controller is required to reject disturbances from various sources and maintain the position of the transducer on a single track.

This thesis focuses on designing a robust controller for the HDD system. The controller must be robust to its external environment, such as changes in temperature, and provide good performance to thousands of drives. An adequate uncertainty model designed using first principles is not available for robust controller design. Thus a set of frequency response data (FRD) measured from a number of HDDs and at different temperature points is used to design the uncertainty model of the system. A basic method of averaging the set of FRD to create an uncertainty model is used to design the baseline controller through a standard D-K synthesis method. A numerical algorithm is then developed to create an optimal uncertainty model for the system using the experimental FRD. Using this algorithm, a temperature dependent model is designed for the purpose of designing a temperature dependent robust controller. Finally a temperature dependent controller is designed to increase the performance of the HDDs compared to the baseline controller, and the theoretical validation for the method is given.

Contents

List of Tables	viii
List of Figures	ix
Chapter 1 Introduction	1
1.1 Dissertation Overview	2
1.2 Dissertation Contribution	4
Chapter 2 Modern Hard Disk Drives	6
2.1 Brief History	6
2.2 Mechanism	9
2.3 Disturbances	11
2.4 Control System for HDD System	13
Chapter 3 Baseline Control Design	16
3.1 Introduction	16
3.2 Uncertainty Modeling	16
3.2.1 Nominal Modeling	18
3.2.2 Uncertainty Weights	20

3.3	Interconnected Model Design for D-K Synthesis	22
3.4	D-K Synthesis	24
3.5	Controller Performance	26
3.6	Controller Implementation	27
3.7	Conclusion	28
Chapter 4 Optimal Uncertainty Modeling		29
4.1	Introduction	29
4.1.1	Problem Statement	30
4.2	Numerical Algorithm	31
4.3	Practical Issues	35
4.3.1	Incorporating Prior Knowledge	35
4.3.2	Limiting Magnitude of Nominal Model Derived From Multi- multiplicative Uncertainty Set	35
4.3.3	Application to MIMO Systems	36
4.4	Simple Numerical Example	38
4.5	Application of Optimization Method on HDD Data	41
4.5.1	Designing MISO Models	43
4.6	Conclusion	44
Chapter 5 Temperature Dependent Modeling		45
5.1	Introduction	45
5.2	Problem Formulation	47
5.3	Temperature Dependent Modeling	48
5.3.1	Benefits of Temperature Dependent Modeling For HDD System	48

5.3.2	Method	51
5.3.3	Constructing Temperature Dependent Models	56
5.3.4	Comparison with Temperature Independent Model	57
5.4	Conclusion	61
Chapter 6 Temperature Dependent Robust Controller		62
6.1	Introduction	62
6.2	Problem Formulation	64
6.3	Temperature Dependent Robust Controller	65
6.3.1	Method	65
6.4	Results	68
6.4.1	Temperature Dependent Nominal Models	68
6.4.2	Uncertainty Weights	72
6.4.3	Actuator Weights	73
6.4.4	Performance Weight	74
6.4.5	Discretization	74
6.4.6	Deriving a D-Scale	75
6.4.7	Conditioning State-Space Matrices For Numerical Reasons	76
6.4.8	Gain Scheduled Controller	77
6.4.9	Experimental Results	84
6.5	Special H_∞ Problems	85
6.5.1	Disturbance Feedforward and Full Information Problem	85
6.5.2	Equivalence of DF and FI Problem	86
6.5.3	Equivalent Controllers for DF and FI Problems	87

6.5.4	H_∞ Synthesis For DF and FI Systems	88
6.6	Interpolation of HDD Controller	90
6.6.1	HDD Problem as a DF System	90
6.6.2	Interpolation of H_∞ Controller	94
6.7	Separation Theory	96
6.8	Conclusion and Future Work	98
Chapter 7	Conclusion and Future Work	99
7.1	Recommendations For Future Work	100
	References	102
	Appendix A MATLAB Functions	111

List of Tables

5.1	The temperature of the HDD for index l	49
5.2	The isolated frequencies and the second order system for each H_n . . .	55
5.3	The area under the optimal weights from normalized frequency of 350 to 600 for temperature specific models and temperature independent model.	60
6.1	The temperature of the HDD for index l	65
6.2	The robust performance achieved by D-K synthesis at each temperature point.	75
6.3	The input and output channel dimensions of the interconnected model.	92

List of Figures

2.1	Picture of a hard disk drive with the front cover removed	7
2.2	Components of hard disk drive actuator system.	8
2.3	The diagram of the tracks on a magnetic media.	9
2.4	Diagram of a servo burst signal when the magnetic head is at the center of the track.	10
2.5	Diagram of a servo burst signal when the magnetic head is off the center of the track.	11
2.6	Shape of a real track.	12
2.7	Block diagram of a dual stage acutator with a controller.	13
2.8	Block diagram of the HDD system using the PQ method.	14
2.9	Block diagram of the HDD system using the decoupled master slave method.	15
3.1	The Bode plot of a set of experimental frequency response data for VCM and MA.	17
3.2	LTI uncertainty model of the HDD system.	18
3.3	The average of the experimental data and the LTI nominal model for VCM and MA	19
3.4	The optimal uncertainty weight and the resulting uncertainty weight for VCM and MA used for baseline control system.	21

3.5	The interconnected model used to design a baseline robust controller, $K(s)$	22
3.6	The inverse of the baseline performance weight.	23
3.7	Robust control problem	24
3.8	The block diagram for linear fractional transformation of the interconnected model, P , and controller, K	24
3.9	Block diagram of the scaled control problem.	25
3.10	Bode plot of the robust baseline controller designed using D-K synthesis for two different performance weights.	25
3.11	Bode plot of the open-loop response and closed-loop sensitivity created from two different baseline robust controllers.	26
3.12	Bode plot of the open-loop response and closed-loop sensitivity derived using the experimental data and baseline robust controllers.	27
3.13	The closed-loop sensitivity of the baseline robust controller measured off a real HDD.	28
4.1	Bode plot of the simple case models, its mean, and its optimal frequency response data.	39
4.2	The optimal multiplicative uncertainty weight required to cover all of the sampled models using the optimal FRD and mean.	39
4.3	The complex plot of the sampled data points with the smallest circle required to cover the data using the LMI method and the averaging method at $\omega = 5$	40
4.4	Bode plot of the average of the experimental HDD FRD and the optimal FRD based on the LMI method.	41
4.5	The optimal additive uncertainty weight required to cover the data using the average of the HDD data and the optimal nominal FRD.	42
4.6	The block diagram of the HDD system using MISO state-space model.	43

4.7	The FRD of the MISO optimal HDD FRD and MISO HDD model. . .	44
5.1	The experiential FRD of HDD data at 16.3°C and 59.6°C for the VCM.	47
5.2	The experiential FRD of HDD data at 16.3°C and 59.6°C for the MA.	48
5.3	The optimal uncertainty weight for optimal MA FRD for temperature independent and dependent systems.	50
5.4	The FRD of HDD data and the optimal FRD at 30.6°C	53
5.5	The optimal FRD and the same FRD with isolated frequencies at 30.6°C.	53
5.6	The isolated optimal nominal FRD, and the second order fit.	54
5.7	The optimal FRD with the first mode divided out with isolated frequencies for the second mode of interest.	54
5.8	The optimal FRD and the state-space system constructed from series of second order systems.	56
5.9	The linear fit of an entry of the A matrix based on the state space system designed at each temperature point.	56
5.10	Temperature dependent model and the temperature specific model at 30.6°C.	57
5.11	The temperature dependent and independent model of the MA used for the control design.	58
5.12	The optimal uncertainty weight for temperature independent MA model, $G_{T_{ind}}(s)$, and temperature dependent MA model, $G_{T_{dep}}(s, T)$	59
6.1	Uncertain temperature dependent robust synthesis problem.	65
6.2	The interconnected model used to design a robust controller, $K(s)$. .	66
6.3	The temperature dependent and independent nominal model of the VCM used for the temperature dependent and independent controller design.	69

6.4	The temperature dependent and independent nominal model of the MA used for the temperature dependent and independent controller design.	70
6.5	The temperature dependent model of the VCM at 16.3° and at 59.6°C.	71
6.6	The temperature dependent model of the MA at 16.3° and at 59.6°C.	71
6.7	The VCM uncertainty weight for temperature dependent and independent system.	73
6.8	The MA uncertainty weight for temperature dependent and independent system.	73
6.9	The actuator weights used for VCM and MA for temperature dependent and independent controller design.	74
6.10	The inverse of the performance weight for temperature dependent and independent system.	75
6.11	Bode plot of the D-scale used to design a temperature dependent controller.	76
6.12	The Bode plot of the temperature specific controller designed at 30.6°C and linearly interpolated controller at 30.6°C.	78
6.13	The open-loop and closed-loop sensitivity of the controller designed at 30.6°C and gain scheduled controller at 30.6°C.	79
6.14	The upper bound on the robust performance of the temperature independent, specific, and dependent controllers using the temperature dependent scaled interconnected model.	80
6.15	The nominal sensitivity of the HDD system using the temperature dependent model, temperature independent controller and temperature dependent controller.	81
6.16	The open-loop and closed-loop sensitivity of the temperature independent controller and temperature dependent controller at 30.6°C using the experimental FRD.	82

6.17	The average of the FRD of the closed-loop sensitivity based on the temperature independent controller and temperature dependent controller.	83
6.18	The closed-loop sensitivity measured from a HDD using temperature independent controller and temperature dependent controller interpolated at 30°C.	84
6.19	H_∞ problem	86
6.20	Transformation of DF to FI system.	87
6.21	DF controller	88
6.22	Simplified HDD interconnected model block diagram.	91
6.23	Robust synthesis problem with D_{22} loop-shifted.	92
6.24	Robust controller with D_{22} loop-shifted.	92

Chapter 1

Introduction

Since their introduction to the public in 1956, hard disk drives (HDDs) continue to be a popular data storage device for personal computers and data centers [1]. In 2014, a total of 564 million units of hard disk drives were sold internationally, which was equivalent to 540 exabytes, or 540 million terabytes, of storage being sold that year. It is projected that by 2020, the total annual shipment of HDD storage will grow to 2,000 exabytes [2]. With increasing demand for more data storage, HDD manufacturers continue to improve every component of their products.

HDDs have been a source of research problems in the fields of materials, signal processing, and control systems as manufacturers continue to push their limits. In the field of material science, there are projects based on improving magnetic media to increase the storage density of HDDs [3]. In signal processing, there has been work in designing better magnetic signals to fit more data onto the media [4]. Within the HDD is a servo mechanical system, which consists of a recording media rotating at a constant rate and an actuator system that is used to position the magnetic head on the media to read or write data. In this dissertation, we consider the problem of designing a robust controller for the actuator system inside an HDD. To aid in this research, a set of experimental frequency response data was provided to the Controls Lab in Aerospace Engineering and Mechanics department at University of Minnesota by Seagate.

The robust control design for the HDD's actuator system is an interesting problem for several reasons:

1. The data on the recording media are organized as circular tracks, and to increase the storage density, these tracks have gotten narrower over time. Current HDDs use recording media with tracks that are less than 100nm wide [5]. Therefore the controller needs to be able to reject disturbances above 2000 Hz due to the increased storage density of the HDD [6]. The elastic response of the actuators have to be modeled to achieve a bandwidth of such high frequency.
2. To achieve high bandwidth, the modern HDDs use a dual stage actuator system. Since the only position measurement made by the HDD is from the signal on the recording media, it is a two-input one-output system.
3. The high bandwidth requirement of the controller requires the high frequency elastic response of the system to be modeled. Experimental frequency response data is easily obtainable and is commonly used to design robust control systems.
4. Modern HDD systems require a complex network of different control systems to achieve a specific performance. In this dissertation, the scope of the project is to design a robust controller for the purpose of rejecting high frequency and non-repeatable disturbances.
5. HDDs are sold in large volumes, and therefore the control system for the dual-stage actuator must be robust for thousands of drives per product. Due to manufacturing tolerances, the high frequency dynamics are slightly different between drives. Another factor that can change the dynamics of the system is temperature. The controller must be robustly stabilizing to these different factors and still provide a high bandwidth.
6. Different controllers can be tested on HDDs without significant monetary consequences due to its durability and low cost. With the help of engineers from Seagate, it was possible to validate the control synthesis methods that are described in this dissertation.

1.1 Dissertation Overview

In this dissertation, methods used to design a temperature dependent controller are described to increase the performance of HDDs under varying temperature conditions.

Chapter 2 gives an overview on the relevant HDD components and their improvements over time. In the last 60 years, since the invention of the first HDD, every

component of the HDD has been improved over time. These different improvements have made it possible for HDDs to increase its storage capacity as it shrank in size. The key components for this dissertation are the voice coil motor, the micro actuator, the magnetic head and the servo pattern written on the recording media. The details on how the HDD measures the position of the magnetic head through the magnetic patterns on the media will be explained. Since the system is a two-input one-output system, the different control schemes that are applicable to the system are explained as well.

In **Chapter 3**, a temperature independent controller is designed for the system. Although the method used in this chapter is not novel, the controller designed in this chapter is used as a baseline system, which the performance of other controllers can be compared against. A basic method of fitting a state-space system to the mean of experimental frequency response data is used to design the uncertainty model of the HDD system. The method used to design the performance and actuator weights are described in this chapter as well. Two different performance weights are designed to create two different controllers. Once an interconnected model of the HDD system with uncertainty is created, a conventional D-K synthesis is used to design the LTI robust controllers for the system. The controllers are ultimately validated by implementing them onto a HDD system.

In **Chapter 4**, a method to design an optimal nominal model based on experimental frequency response data is described. The developed algorithm finds the frequency response data which minimizes the dynamic uncertainty of the model. The D-K synthesis method, described in Chapter 3, requires an uncertainty model to design a robust controller. The method described in this chapter will derive a nominal model and weight that is optimal at each frequency point based on a set of experimental frequency response data. By optimizing the uncertainty model, conservatism in the control design can be reduced to increase the performance of the system. Furthermore, various model types applicable to the HDD system are discussed in the chapter.

In **Chapter 5**, a method used to design a temperature dependent model for the HDD system is described. As the temperature of the HDD system increases, the metal of the actuator softens, and its natural frequency decreases. In Chapter 3, a single LTI model of the HDD system is designed using all of the experimental data as a single set. In this chapter, the set of experimental frequency response data is separated

into multiple sets based on the temperature at which the data was collected. By designing a model at each temperature point while keeping the states consistent, a temperature dependent model was designed. A temperature dependent model reduces the uncertainty of the system at the frequencies of the first several high frequency modes and increases the performance of the controller designed in Chapter 6.

In **Chapter 6**, a method to design a temperature dependent controller for the HDD system is described. A gain scheduled controller was designed using the temperature dependent model from Chapter 5 to increase the performance of the system under varying temperature conditions. Modern HDDs have a temperature sensor built-in, and thus a gain-scheduled controller can be implemented. A modified D-K synthesis method was developed to design a gain-scheduling controller for a system with slow varying parameter. Furthermore the theory behind the controller synthesis method is described in detail. The goal was to increase the bandwidth of the closed-loop sensitivity without increasing its peak or low frequency gain compared to the temperature independent controller designed in Chapter 3. The improvement of the controller was ultimately validated by implementing it into a real HDD system.

Chapter 7 concludes the dissertation, and recommends possible future work. The appendix includes some of the core codes that were used to design the controllers.

1.2 Dissertation Contribution

Based on a research that focuses on designing robust controllers for HDD system, this dissertation contributes to the field of robust control systems in two ways.

1. **Practical Application to HDD:** Throughout the dissertation, an emphasis is placed on the application of robust control research to the HDD system. Using experimental HDD data, the methods that are described in the dissertation are designed to be applicable for real HDD systems. The practical issues that occur for HDD systems are resolved in several chapters. With collaboration from Seagate, it was possible to implement the designed control systems onto a HDD and validate their performance. In Chapter 3, an established D-K synthesis method was applied to the HDD problem using the tools available in the Robust Control Toolbox in MATLAB. Although the application of D-K synthesis to such a system is not novel, the described method provides insight

into control design methods for HDD systems. Chapters 4, 5, and 6 provide detailed descriptions on applying the newly developed tools for HDD system. In Chapter 4, a detailed method on designing an optimal uncertainty model of the HDD system is described. In Chapter 5 and Chapter 6, a step by step description of how to design a temperature dependent model and temperature dependent controller for the HDD systems are described, respectively.

2. **Development of Theoretical Tools to Improve HDD Control Systems:**

For the purpose of improving the HDD control system, several numerical tools were developed. The first tool, described in Chapter 4, provides a method for designing an optimal uncertainty model based off a set of experimental frequency response data. The linear matrix inequalities used to find the optimal nominal model is based off the same underlying theory that is used for the MATLAB function `ucover` [7]. This tool designs an uncertainty model of the HDD system based on minimizing the uncertainty set of the system and thus reduces the conservatism from the model and increases the performance of the system. It also provides a standardized method for designing an uncertainty model from a given set of experimental frequency response data. The second tool, described in Chapter 5, provides a method for designing a parameter dependent model based on a set of experimental frequency response data for a slow varying parameter. The third tool, described in Chapter 6, provides a method for designing a gain-scheduled controller for a system with slow varying parameter. The H_∞ theory is also looked at in depth to provide the necessary conditions to design a linearly interpolatable D-K synthesis controller.

Chapter 2

Modern Hard Disk Drives

2.1 Brief History

The hard disk drive (HDD) is a device that is commonly used to store digital data. It works by spinning a magnetic disk while a magnetic transducer, commonly called a head, reads/writes data from/to circular tracks on the disk. The first computer with a HDD to hit the market was the Random Access Memory Accounting Machine (RAMAC), introduced by IBM in 1956 [8]. The RAMAC's 5 MB storage capability and the ability to access digital data in real time rendered the punch card systems obsolete at the time. It was a large system that contained 50 disks, each 24 inches in diameter, and had a data storage density of about 2 kilobits/in². A modern HDD, shown in Figure 2.1, uses disks that are less than 3.25 inches in diameter and has a storage density of over 0.25 Terabits/in² [9] [10] [11]. Along with the increase in storage density, the price for HDDs has drastically decreased as well. The cost of RAMAC was approximately \$32,000 per MB in 1956, compared to a modern HDD that costs less than \$0.10 per GB [12]. The fundamental mechanism of a HDD has not changed since the invention of the RAMAC, however the components of the HDD have been significantly improved over the years.

One of the main components that has been improved is the magnetic disk, or the recording media, which is used to store the digital data. Over time, the base material and magnetic coating of the media was improved to increase its storage capabilities. Initial HDDs used magnesium substrate as the media material, while current HDDs use aluminum alloy or glass depending on the size of the media and how fast the



Figure 2.1: Picture of a hard disk drive with the front cover removed

media is set to spin within the HDD. RAMAC used media that was painted with ferrite, but as the media material changed and the disk sizes were reduced, the thin-film sputtering has become the primary method of coating the media [13] [8]. As physical limitations on the thin-film sputtering method is reached, new technologies continue to be developed to improve the media. One such technology is the bit-patterned media where the magnetic coating is done such that there exists extremely small islands of magnetic material that can carry 1 bit of information [13] [14]. The improvements that were made to the media have increased the number of tracks that it can hold on one surface, which is typically measured as tracks per inch (TPI). Since the digital data is directly stored onto the media, improving the track per inch (TPI) capability of the media is directly related to improving the storage density of HDDs [9]. The improvements made onto the media have increased its TPI from 20 to more than 340,000, however the increase in TPI has reduced the width of the tracks to be less than 75 nm [6] [5]. As the tracks became extremely narrow, the actuator system and its control system had to improve as well.

The actuator used in the HDD had to improve along with the media technology to be able to place the magnetic head onto the center of a desired track. The overall system that is used to maintain the position of the magnetic head is called the servomechanical system, and is called the servo system for short. Initially the RAMAC utilized linear actuators, made from aircraft cables and pulleys, to move a single magnetic head vertically and radially across the disks. The first major development was the comb structure, which allowed the actuator to hold a magnetic head for each surface of the media, and therefore require the actuator to move in only one dimen-

sion. The linear actuators were then replaced by rotary actuators to increase stability against linear shocks and vibration [15]. Modern HDDs utilize a dual stage actuator, as shown in Figure 2.2, where it uses a main rotary actuator named the voice coil motor (VCM) for the broad general motion, and a micro actuator (MA) near the magnetic head for a more precise but much smaller range of motion [16]. The VCM is named for its similarity to how a loudspeaker works, and the MA is comprised of two piezoelectric elements that contract and expand to tilt the tip of the actuator. Although only one VCM is used for the entire rack of disks, there is a MA for each surface of disks inside the HDD. The improvements made in the actuator system have allowed HDD systems to take advantage of the increase in TPI capabilities of the media [9].

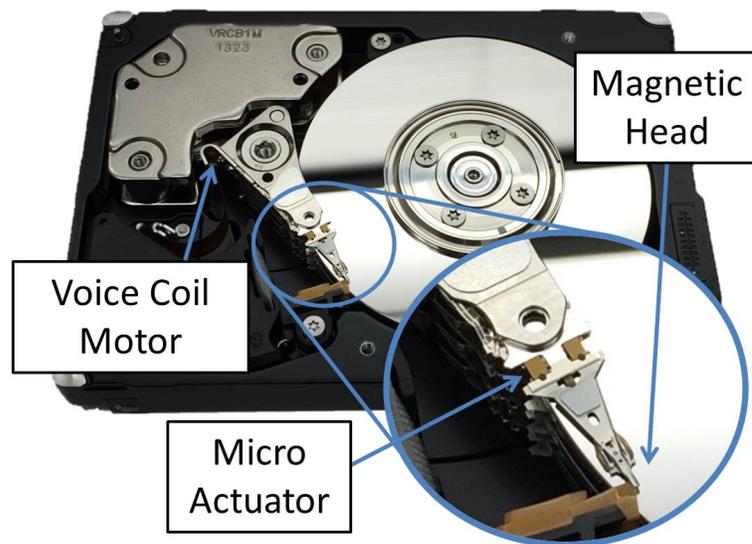


Figure 2.2: Components of hard disk drive actuator system.

With the improvements made over time to the components of the HDD, its storage density has been making drastic improvements over time [3]. Historically, since the invention of the HDD, its storage density has been doubling almost every year [11]. There is some doubt that this rate of improvement will not be sustained as physical limitations on media technology are reached, but it is still likely that the storage density will continue doubling every 24-36 months [17]. Interested readers should read the references cited in this section for additional details.

2.2 Mechanism

As stated in the previous section, modern HDDs use a dual stage actuator comprised of the VCM and the MA, as shown in Figure 2.2. There is one VCM for all of the magnetic heads, while there are individual MAs for each surface. The VCM provides a general range of motion so that the magnetic heads can be positioned from inner to outermost radial position on the disk, while the MA provides a more accurate movement, but with a much smaller range compared to the VCM. The MA is comprised of two piezoelectric elements that contract and expand to tilt the position of the magnetic heads, and it can be seen as the two gold elements in Figure 2.2. Although there is a magnetic head for each surface of the disk, or two transducers per disk, only one magnetic head is used at a time to read/write data. Furthermore there are no sensors on the actuators to measure the position of the magnetic head, thus the positional information is obtained through signals encoded on the magnetic disks.

The magnetic signals are organized as circular tracks with certain radial width on the surface of the magnetic disks. Furthermore, these tracks are divided angularly into data and servo sections as shown in Figure 2.3. The data sections are the angular

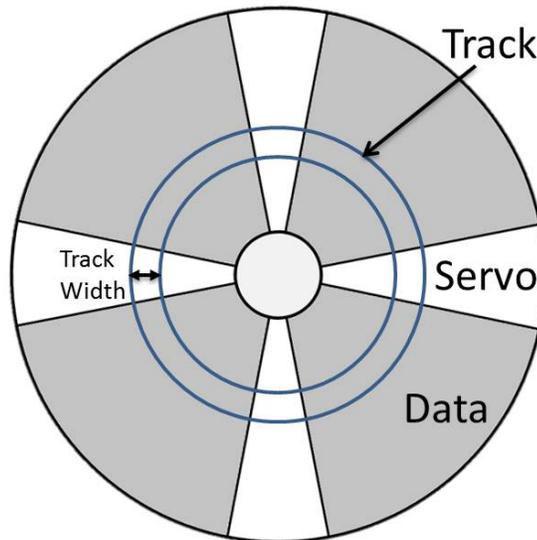


Figure 2.3: The magnetic signals are radially organized as tracks, and angularly organized as servo and data sections. Typically there are over 100 servo and data sectors [4]

components of the tracks that are used to store digital data, while the servo sections contain encoded signals used as markers for the HDD to determine the position of

the magnetic head. The magnetic signals on the servo sections are encoded during manufacturing and are not changed in a typical HDD usage. Typically 10% of the media area is used for the servo section [12]. Each servo section contains various components to allow the HDD to calibrate and calculate the position of the magnetic head. The primary components of the servo section used to determine the position of the head are the gray code and the servo burst signals [4]. The gray code contains the information on the angular and radial position of magnetic head relative to the entire surface of the disk. The servo burst signals are used to calculate the radial position of the magnetic head relative to the center of the track, otherwise known as position error signal (PES).

Typically the servo burst signals are arranged as shown in Figure 2.4 and 2.5. The servo burst signals labeled PS1 and PS2 are offset such that each of them take up one half the width of the track. If the magnetic head is located perfectly at the center of the track, as shown in Figure 2.4, the HDD will read two magnetic signals with equal amplitude. If the head is off-center, as shown in Figure 2.5, the signal strength from the two signals will be different [6]. Using the difference in the strength of the signals, the HDD is able to calculate the PES [4]. The PES can then be used as the input into the control system.

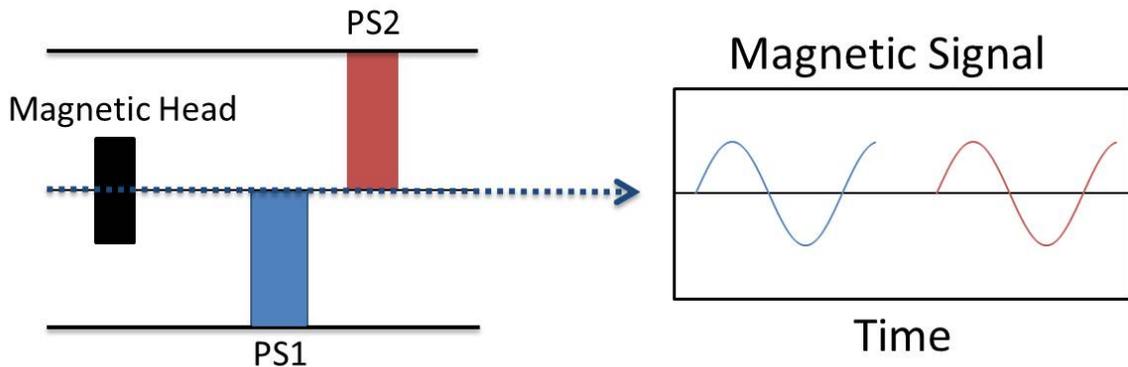


Figure 2.4: Servo burst signals read by the magnetic head when it is placed on the center of the track.

Lastly, there are two operating modes for the actuators, track seeking and track following. Track seeking occurs when the magnetic head needs to be moved from one track to another track. Track following occurs when the HDD needs to maintain the position of the magnetic head on one track to read/write data from/onto it. This thesis focuses on designing a robust controller for track following mode, where

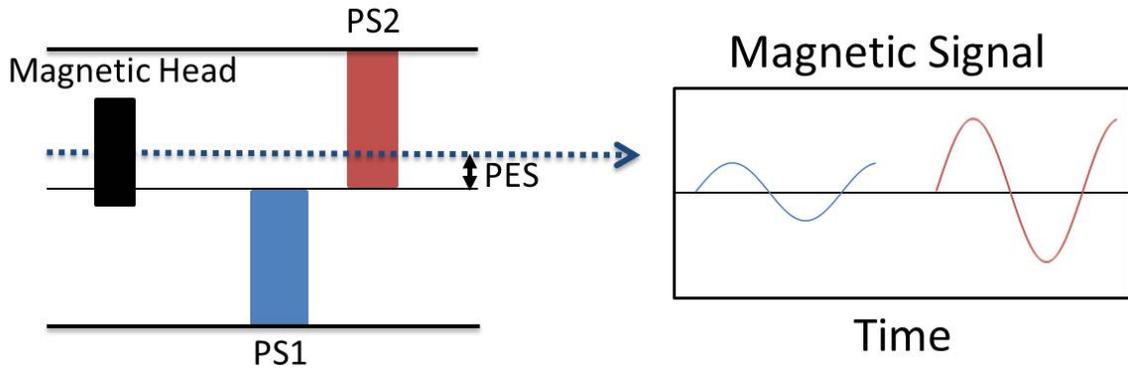


Figure 2.5: Servo burst signals read by the magnetic head when it is placed off center from the track.

various disturbances must be rejected by the controller to maintain the position of the magnetic head [9].

2.3 Disturbances

There are various disturbances that can affect the performance of the actuator system. These disturbances can cause the actuator to position on the wrong track. The event of being on the wrong track is known as track misregistration (TMR), and a control system is implemented in the HDD to minimize its occurrence [18]. There are several causes of disturbances that can affect the HDD system. The manufacturing tolerances can affect the shape of the track and cause what is known as a repeatable run out disturbance. The fluid movement caused by the rotation of the disks is a type of disturbance that is known as windage. There are also external disturbances that are caused by forces outside of the HDD.

One of the primary disturbances occur due to manufacturing tolerances that cause the magnetic tracks to be non-ideal in shape. Although in Figure 2.3, the track is shown as a perfect circle, in reality the tracks are wobbly due to manufacturing errors as shown in Figure 2.6. Since the actuator spins at a constant rate, this error repeats every cycle. Therefore this type of error or disturbance is known as repeatable run out (RRO) [6]. There are various ways the RRO can be compensated. One type of method relied on a feed forward controller to compensate the repeated error [6, 19]. Another type relied on disturbance estimation and learning gain, which could take some cycles of data for the RRO to be compensated [20,21]. Since RRO is a repeatable disturbance, it is also possible to measure and store the RRO correction value onto

the media itself [22]. Within a modern HDD, there are various systems that help to mitigate RRO, but not completely cancel it out. Any leftover disturbances from RRO must be compensated by the robust controller designed in this thesis.

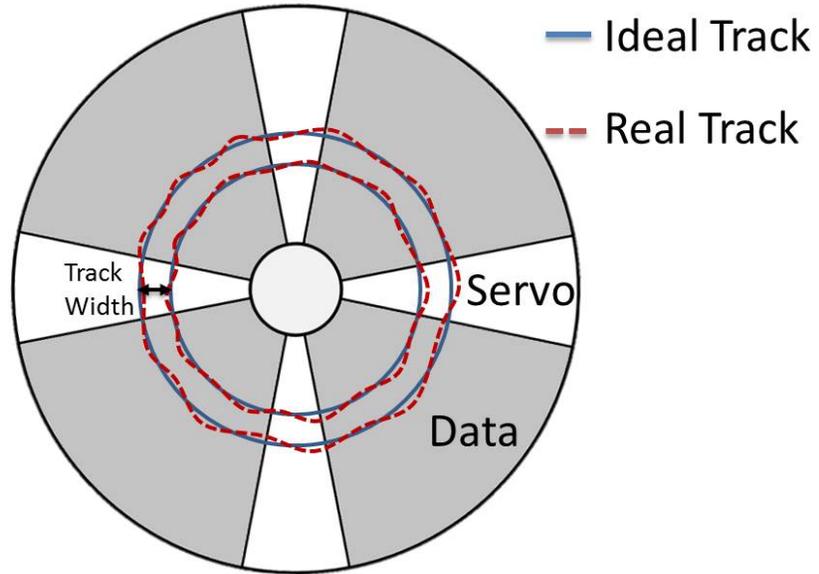


Figure 2.6: Real tracks are not written as perfect circles [6].

The spinning disks inside the HDD can cause a disturbance within the system. Due to boundary layer effect, as the disk spins, the fluid inside the system moves rapidly. The force caused by the movement of fluid inside the HDD is called windage [6]. The magnetic heads that are sandwiched by the magnetic disks will typically have different windage force compared to the magnetic heads that are exposed on one side. The spinning media can also have its own resonance, called disk platter resonance, that can disturb the position of the magnetic head [23].

Including external sources of disturbances, such as music from laptop speakers [24], the different disturbances have to be rejected to minimize the occurrence of TMR during track following. The bandwidth of the HDD control system has to increase along with the narrowing of the tracks to increase storage capacity. Unfortunately, the mechanical resonant modes of the HDD actuators prevents the bandwidth of the system to be increased arbitrarily [25]. Over the years, various control systems have been developed for the servo system to maintain the position of the magnetic heads.

2.4 Control System for HDD System

The disturbances into the HDD system can excite the high frequency dynamics of the actuators, which can vary from HDD to HDD due to manufacturing tolerances. A high performance robust controller for a HDD is required to maintain the position of the magnetic head on very narrow tracks. The temperature of the HDD can change the property of the metal causing the high frequency modes to shift in natural frequency and damping ratio. Frequency response data of hard disk drives that were collected for this thesis can be separated into various categories of different HDDs, heads, and temperatures. A robust controller is designed in this thesis such that the HDD can achieve high performance across different HDDs, and temperatures. It is assumed that a controller would be designed for each magnetic head within the HDD.

The concept of using a dual stage actuator with a piezoelectric system and a voice coil motor for the HDD was introduced in literature as early as 1991 [26]. The dual stage actuator was developed to increase the bandwidth of the actuator system to keep up with the improvement of the media as it reached a capacity of over 10 kTPI. A general controller configuration for the HDD system is shown in Figure 2.7. The dual stage HDD is modeled as a two-input, one-output system. The output from these actuators sum to provide the net position of the head [27]. During track following the reference input, r , is zero, and the disturbance, d , is the only input into the HDD system. The output, y , is the position of the magnetic head which is typically measured through the PES as described in Section 2.2. Alternatively, an extraneous sensor, such as the laser Doppler vibrometer, can be used to measure the head position during an experiment [28]. Finally, a one-input, two-output controller, K , is designed to actuate the VCM and MA.

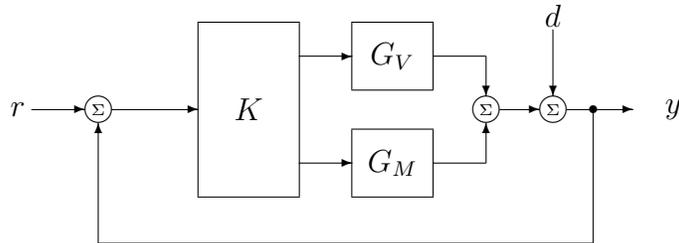


Figure 2.7: Control structure for HDD with dual stage actuators.

One common control configuration used for the dual stage actuator system is shown in Figure 2.8. This control system configuration reduces the MISO design problem

to two SISO design problems, however it can not guarantee the stability of the VCM feedback loop [29]. Early on in the control development for dual stage actuators, what is known as the PQ method was used to design a controller using transfer functions [6, 30]. The open loop transfer function of the system shown in Figure 2.8 can be expressed as

$$1 + \frac{G_M K_M}{G_V K_V} = 1 + PQ = 0, \quad (2.1)$$

where $Q = K_M/K_V$ and $P = G_M/G_V$. Using the PQ method a controller, Q , that can stabilize P is first designed. Then from Q , stable controllers K_V and K_M can be derived [30]. Using H_∞ or H_2 control design method, it is possible to simultaneously design both the controllers by creating a weighted interconnected model of the system as well [31].

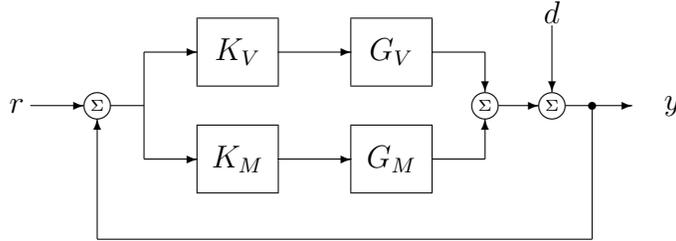


Figure 2.8: Control structure for HDD with dual stage actuators using the PQ method.

Another common control configuration for the HDD servo system is called the sensitivity decoupled controller, or the decoupled master slave configuration. In this configuration, an estimator of the MA is used to decouple the controllers for the VCM and the MA as shown in Figure 2.9 [32] [33] [34]. This allows the controller for the VCM and MA to be designed separately, and thus stabilize the VCM and MA control loops separately [35]. Due to reliance on an estimator of MA, which also has hysteresis characteristics [32], there is an added complexity to the system design. Typically for a decoupled master slave configuration, the VCM controller is designed first to achieve basic performance and stability. Then the estimator and the controller for the MA is designed to increase the bandwidth of the system [29, 34].

There are various research groups that have worked on designing a robust controller for a HDD using the H_∞ and H_2 synthesis methods [36]. Initially, the models used to design the controllers were low order systems [37], however as the desired performance of the controllers increased, higher order uncertainty models became necessary. Since the high frequency modes of the HDD actuators are difficult to model analytically,

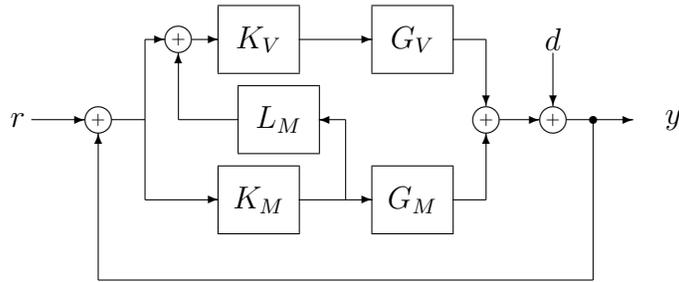


Figure 2.9: Control structure for HDD with dual stage actuators using the DMS method.

most uncertainty models for the HDD actuators are derived based on experimental frequency response data [28] [38] [39]. These uncertainty models are used to design a robust controller through either a H_∞ or H_2 synthesis methods [40] [41].

In addition to H_∞ and H_2 controller synthesis methods, there have been other modern control synthesis method applied to the HDD servo system. One such method is applying the PQ method using modern tools in combination with contact induced vibration data to design an effective controller [42]. Another method that has been developed to improve disturbance rejection is to filter specific narrow-band disturbances that affects the HDD system [43]. A switched controller design using H_2 controller synthesis has been used on HDD system to achieve high performance while avoiding saturation of MA during track seek operations [44]. Similarly adaptive controllers and model predictive controllers have been applied to HDD servo systems to improve disturbance rejection and improve tracking performance [25, 45].

The HDD system provides a stable platform for testing high bandwidth controllers using various methods. For this thesis a temperature dependent controller was developed and tested on a HDD using the configuration shown in Figure 2.8, where the controllers for VCM and MA are designed as individual SISO systems. In the next chapter, the controller design method that was used to create a baseline robust controller using D-K synthesis will be described.

Chapter 3

Baseline Control Design

3.1 Introduction

In this chapter a baseline method of designing a robust controller from experimental frequency data will be shown. The readers of this chapter should be able to recreate the steps taken to design a robust controller for the actuator system of a HDD using D-K synthesis. The relevant codes will be shown in the appendix. The objective is to design a robust controller that will stabilize the HDD and achieve good performance across numerous drives and temperature. The general steps taken to design a robust controller for HDDs are:

1. Design an uncertainty model for the actuators based on the experimental frequency response data.
2. Design a weighted interconnected model.
3. Use D-K synthesis to design a robust controller.
4. Check the performance of the controller by simulating its open-loop and closed-loop sensitivity using the experimental frequency response data.
5. Test the performance of the controller by implementing the controller on a HDD.

3.2 Uncertainty Modeling

In general a simple model of the voice coil motor (VCM) is a double integrator and a simple model of the micro actuator (MA) is a constant gain with a second

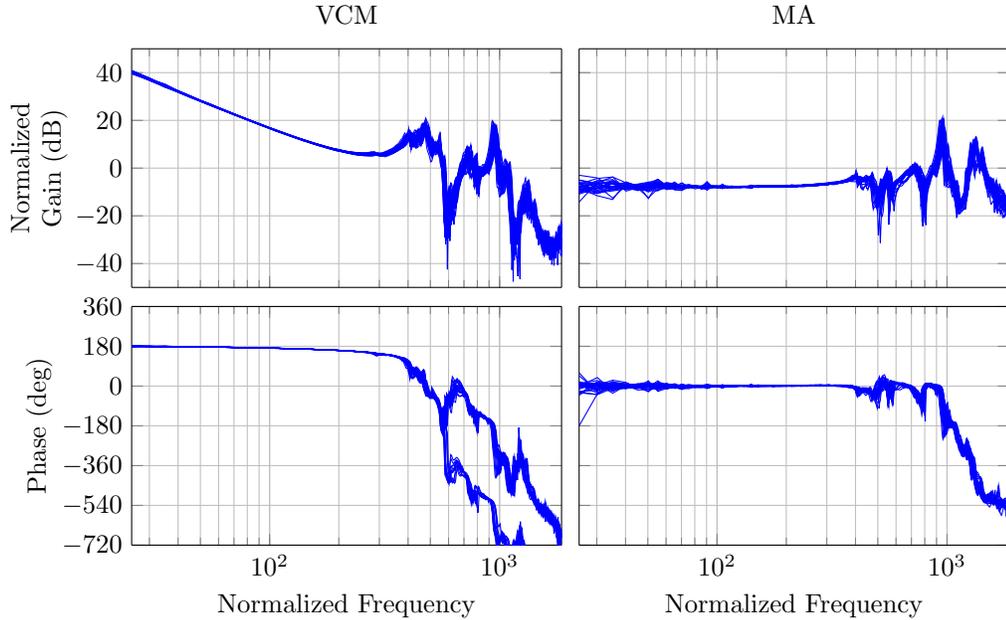


Figure 3.1: The Bode plot of a set of experimental frequency response data for VCM and MA.

order high frequency roll off [9]. Due to the high bandwidth requirement for modern HDDs, however, the high frequency modes of the actuators are important to model to design high performance controllers. Unfortunately, it is difficult to model the high frequency dynamics of the actuators based on first principles due to the complicated elastic response of the system. An alternative method is to create an uncertainty model of the actuators based off experimental frequency response data (FRD) of several different HDDs. The FRD of VCMs from controller voltage input to position error signal (PES) and the FRD of MAs from controller current input to PES were provided to the Controls Lab by Seagate. FRD of VCMs and MAs from 16 HDDs for a specific magnetic head from temperatures of 16°C to 60°C is shown in Figure 3.1. For proprietary reasons, the magnitude and frequency of the data have been normalized by some fixed values.

One definition for the nominal model of the dual-stage actuator is,

$$G_o = G_{V_o} + G_{M_o}, \quad (3.1)$$

where G_{V_o} and G_{M_o} are the nominal model of the VCM and MA, respectively. In this definition the nominal models for the VCM and MA are constructed as independent

SISO systems. For VCM, a multiplicative uncertain model is defined such that,

$$G_V = (1 + W_{V_L} \Delta_1 W_{V_R}) G_{V_0}, \quad (3.2)$$

where W_{V_L} and W_{V_R} are the dynamic uncertainty weights for the VCM, and Δ_1 is the uncertain linear time invariant dynamic with magnitude less than 1. In robust controls, this is one of the standard method to define dynamic uncertainty [46]. A similar uncertainty model is defined for the MA,

$$G_M = (1 + W_{M_L} \Delta_2 W_{M_R}) G_{M_0}. \quad (3.3)$$

The multiplicative uncertainty model for the HDD is defined such that,

$$G = (1 + W_{V_L} \Delta_1 W_{V_R}) G_{V_0} + (1 + W_{M_L} \Delta_2 W_{M_R}) G_{M_0} \quad (3.4)$$

The block diagram for Equation 3.4 is shown in Figure 3.2. The output, y , is the PES in this case.

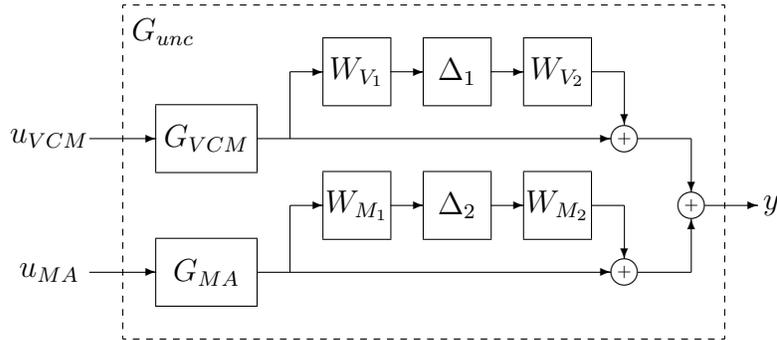


Figure 3.2: Baseline uncertainty model of the HDD system.

3.2.1 Nominal Modeling

One simple method to consolidate the experimental data is to calculate their mean. Let the k^{th} experimental data set ($k = 1, \dots, K$) of one of the actuators be $D_k(j\omega_i)$ defined on a common grid of frequency $\{\omega_f\}_{f=1,2,\dots,F}$. Then the mean of the experimental data is,

$$D_m(j\omega_i) = \sum_{k=1}^K \frac{D_k(j\omega_i)}{K} \quad (3.5)$$

Once the mean of the FRD is calculated, a state-space system can be fit onto it using a function such as `fitfrd` available in MATLAB's Robust Control Toolbox [47]. The state-space fitting method, however, does not guarantee that the VCM or MA models will have the right characteristics outside the frequency range of the experimental data. Therefore once the mean of the FRD is calculated, the known model components, e.g. the dominant double integrator characteristic of the VCM, are divided out from the mean,

$$T_M(j\omega_i) = \frac{D_m(j\omega_i)}{L(j\omega_i)} \quad (3.6)$$

where $T_M(j\omega_i)$ is the unknown model component of the actuator, and $L(j\omega_i)$ is the known model component of the actuator. Due to numerical issues during the control design process, the double integrator for the VCM is approximated by a second order system with poles close to the real-axis and no zeros [9]. A state-space system can be fit onto $T_M(j\omega_i)$ using `fitfrd` to derive $T_{fit}(s)$. Finally the nominal model can be constructed by multiplying $T_{fit}(s)$ and $L(s)$ together. Using `fitfrd`, a fourth order state-space system was fit for the VCM and an eighth order system was fit for the MA. The resulting nominal models for the VCM and MA are shown in Figure 3.3.

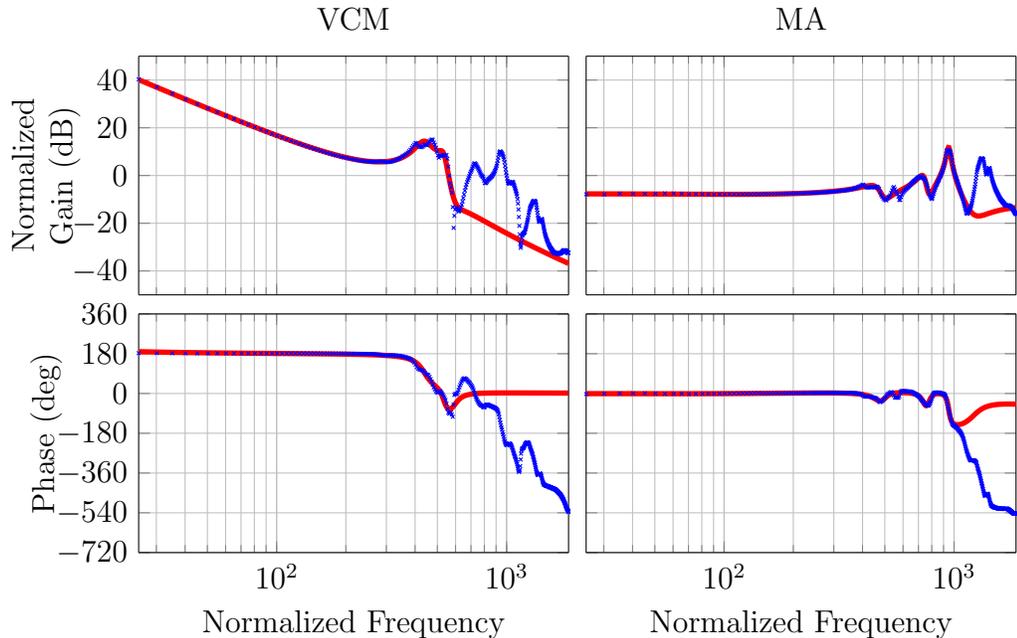


Figure 3.3: The average \bullet of the experimental data and the LTI nominal model --- for VCM and MA.

Although the high frequency modes are similar in complexity between the VCM and

the MA, a higher order fit was used for the MA compared to the VCM. The controller design will be setup such that MA is dominantly used in the high frequency range. Thus it is more efficient to use the extra state orders for the MA model.

3.2.2 Uncertainty Weights

Once the nominal models are created, the uncertainty weights must be designed to create an uncertainty model of the HDD. The purpose of the uncertainty weight is to capture the difference in the dynamics between the nominal model, and the set of experimental data which the model was designed from. The multiplicative uncertainty set, \mathcal{S}_M , for a SISO system is defined as [31],

$$\mathcal{S}_M := \{(1 + W_L \Delta W_R)G_0, \Delta \in \mathbb{RL}_\infty, |\Delta|_\infty \leq 1\} \quad (3.7)$$

The goal is to design optimal uncertainty weights $W_L(s)$ and $W_R(s)$ such that all $D_k(j\omega_i)$ exists within the set while minimizing the size of \mathcal{S}_M . The uncertainty weights for the VCM and MA are designed independently and thus the following steps are repeated for each component.

The initial step is to create an optimal uncertainty weight based on the nominal model, $G_0(j\omega_f)$ and the set of experimental FRD, $D_k(j\omega_f)$. A function called `ucover` on MATLAB can be used to derive optimal uncertainty weights, $W_{Lopt}(G_0, D_k, \omega_f)$ and $W_{Ropt}(G_0, D_k, \omega_f)$, for nominal model G_0 [7]. Since the system is SISO, the dynamics of $W_{Ropt}(G_0, D_k, \omega_f)$ and $W_{Lopt}(G_0, D_k, \omega_f)$ can be consolidated into one variable where, $W_{opt} = W_{Lopt}W_{Ropt}$. Once a state-space system of the weight, $W(s)$, is designed based on $W_{opt}(G_0, D_k, \omega_f)$, the dynamics of the $W(s)$ can be split based on engineering intuition. For the HDD, either W_R or W_L is kept as a constant to reduce the order of the uncertainty model.

The next step is to create a state-space system with a gain that over bounds $\|W_{opt}(G_0, D_k, \omega_f)\|$. There are two methods that can be used to create $W(s)$ based on $W_{opt}(G_0, D_k, \omega_f)$. The first method is to fit a non-minimum phase state-space system that over bounds $\|W_{opt}(G_0, D_k, \omega_f)\|$. A fitting function such as `fitmagfrd` on MATLAB is appropriate for such application. For practical purposes, it is useful to do an iterative fitting method to create the weights, the steps are listed below,

1. Fit a j^{th} order non-minimum phase system, $W^j(s)$, with gain greater than

$$\|W_{opt}(G_0, D_k, \omega_f)\|$$

2. Increase the gain of $W^j(s)$ by 10%.
3. Fit a $(j+1)^{th}$ order non-minimum phase system with gain greater than $\|W_{opt}(G_0, D_k, \omega_f)\|$, but with gain less than the modified $W^j(s)$.
4. Repeat step 1-3 until the desired state order is achieved

Typically the iterative method will start by fitting a first order system. This method prevents large notches from occurring in the state-space fit.

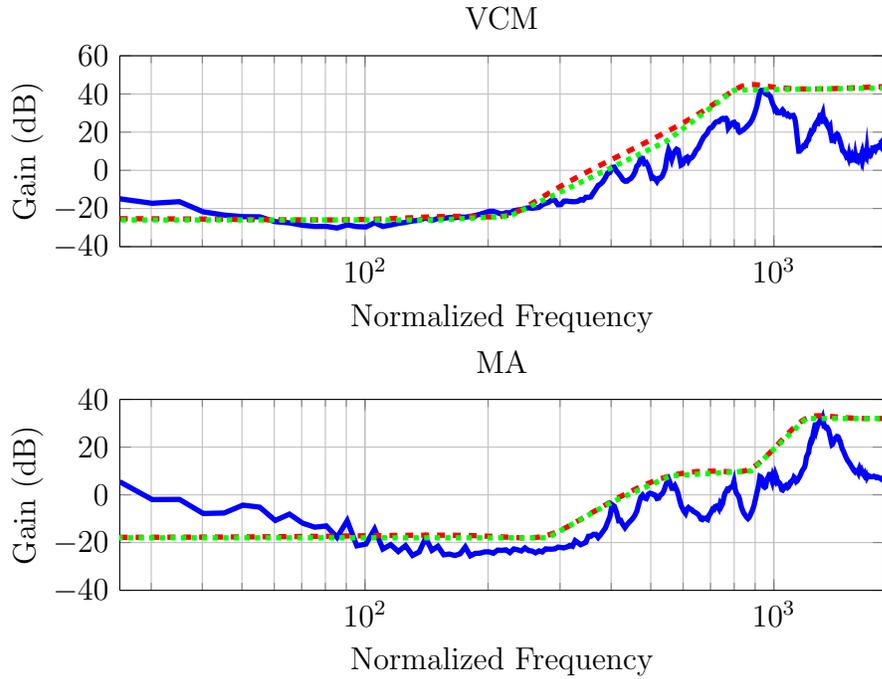


Figure 3.4: The optimal uncertainty weight --- from `ucover` and the resulting uncertainty weight - - - for VCM and MA. $W_g(j\omega_i)$... is used to design the uncertainty weight.

The second method is done by designing a FRD, $W_g(j\omega_i)$ with gain that over bounds $\|W_{opt}(G_0, D_k, \omega_f)\|$, and then fitting a non-minimum phase state-space system that over bounds the gain of $\|W_g(j\omega_i)\|$. The second method allows a user to easily adjust the shape of the uncertainty weight, while incorporating the optimal uncertainty weight, $W_{opt}(G_0, D_k, \omega_f)$. For example, it is known that there is measurement noise at the low frequency, which causes the optimal uncertainty weight to have higher gain at the low frequency. Using the second method allows the user to adjust the gain of the uncertainty weight based on engineering judgment. The $W(s)$ for both VCM and

MA were designed using the second method and are shown in Figure 3.4. $W_g(j\omega_i)$ is also shown in the figure. The VCM uncertainty weight was designed using a fourth order fit, and the MA uncertainty weight was designed using a sixth order fit.

Once the uncertainty weights are designed for both the VCM and MA, the final step is to put the components together as shown in Figure 3.2. Functions available on MATLAB such as `sysic` and `connect` can be used to construct the uncertainty model of the dual stage actuator. Typically the input and output from Δ is dealt as disturbance input and error outputs of the uncertainty system. However, in MATLAB's Robust Control Toolbox, there is a function `ultidyn` that creates uncertain LTI variables, which can be used as a regular transfer function object in MATLAB. With `ultidyn`, the uncertainty model can be directly constructed using Equation 3.4.

3.3 Interconnected Model Design for D-K Synthesis

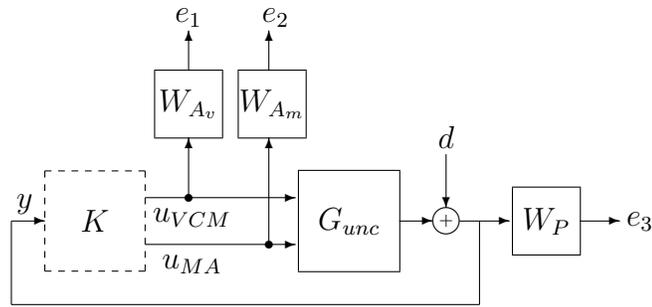


Figure 3.5: The interconnected model used to design a baseline robust controller, $K(s)$

Once the uncertainty model of the system is designed, the next step is to design the performance and actuator weights for the weighted interconnected model as shown in Figure 3.5. In the figure, W_{A_v} and W_{A_m} are the actuator weights for the VCM and MA, respectively, W_P is the performance weight, and K is the robust controller that is to be designed.

It is desirable to favor the use of MA for high frequency responses due to its precise but small range of motion. Thus the W_{A_v} is typically a constant gain while W_{A_m} is a low pass filter. The W_P is designed to shape the closed-loop sensitivity, and is the weight that is adjusted to affect the controller performance the most. The closed-loop sensitivity is the closed-loop system response from the disturbance input to the error output. In the frequency domain, it represents how much the closed-loop

system attenuates or increases the error caused by the disturbance input at different frequencies. The bandwidth is the frequency at which the gain of the sensitivity first crosses -3dB. A controller will be designed to maximize the bandwidth, minimize the gain below the bandwidth, and minimize the peak value of the sensitivity. A better controller allows the HDD to reduce the error between the position of the magnetic head and the center of the track under various disturbances.

A first order performance weight could have the form,

$$W_P(s) = \frac{\mu s + \sqrt{7}\varepsilon w_b}{s + \sqrt{7}w_b} \quad (3.8)$$

where w_b is the target bandwidth, μ is the target peak gain, and ε is the target DC gain of the closed-loop system.

In this chapter, two different performance weights, W_{p1} and W_{p2} will be used to highlight the effect it has on the closed-loop system. The inverse of the performance weights depicts the desired loop shape of the closed-loop sensitivity, and are shown in Figure 3.6. Both of the performance weights are designed to have very similar bandwidth. W_{p1} , is a first order system, and W_{p2} is a fourth order system. The inverse of W_{p2} is designed to have a much lower gain at the frequency below the bandwidth compared to the inverse of W_{p1} . Typically the performance weight should over bound the nominal sensitivity, however when the weight is designed too ambitiously, this will not occur.

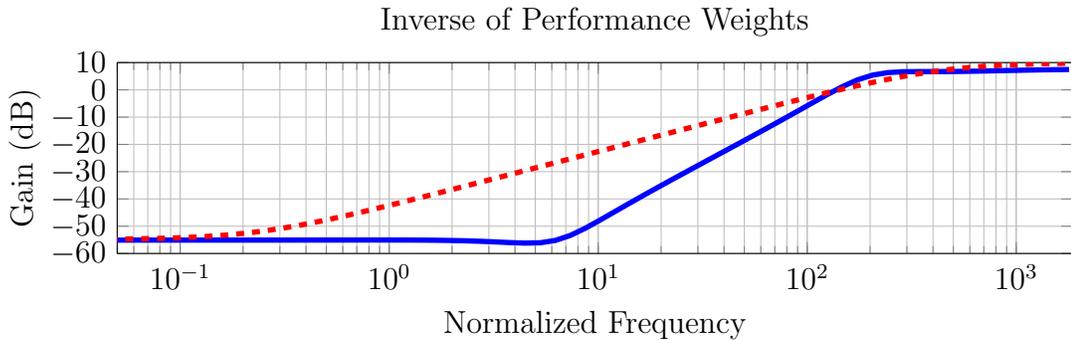


Figure 3.6: The inverse of the performance weight, W_{p1} — and W_{p2} - - .

Before the interconnected model can be constructed using either `connect` or `sysic` function in MATLAB, all of the components must be discretized. The HDD system is inherently discrete as the magnetic head obtains positional data from the servo

section in fixed time intervals. The nominal models are discretized using either zero order hold or first order hold, and the weights are discretized using the matched pole-zero method. If there is a time delay, it is discretized using a bilinear approximation method. Once all of the components are discretized and the interconnected model is constructed, the robust synthesis problem as, shown in Figure 3.7, is setup.

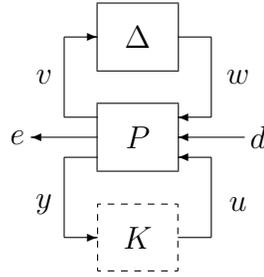


Figure 3.7: Robust control problem

3.4 D-K Synthesis

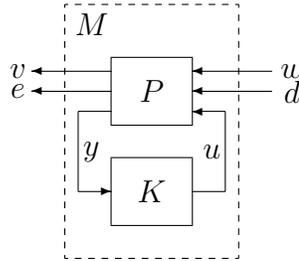


Figure 3.8: The block diagram for LFT of the interconnected model, P , and controller, K .

D-K synthesis is a method that automatically designs a controller that minimizes the structured singular value, μ , of $F_l(P, K)$, as shown in Figure 3.8 [31] [46] [48]. The μ is defined as the worst case gain of the system, or the supremum of the induced L_2 gain over the uncertainty set, in the presence of uncertain perturbations. Due to the numerical difficulty of calculating μ of a system, its upper bound is calculated instead. The upper bound of μ is calculated by finding the induced-2 norm of the system. By scaling the input and output of a system with D-scales as shown in, Figure 3.9, it is possible to find the minimum of the upper bound of μ , which can equal μ in certain cases.

D-K synthesis is an iterative method that cycles between the controller synthesis step and the robustness analysis step. The cycle continues until the improvement in robust

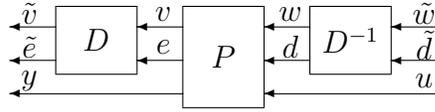


Figure 3.9: Block diagram of the scaled control problem.

performance becomes negligible from the iterative cycle. In the controller synthesis step, the inputs and outputs of P are scaled as shown in Fig. 3.9. Then a stabilizing controller is designed by solving the H_∞ problem [47]. On the first iteration, the D -scale is an identity matrix. On the analysis step, an optimization problem is solved to find the optimal D -scale that will minimize the induced 2-norm of $DF_l(P, K)D^{-1}$.

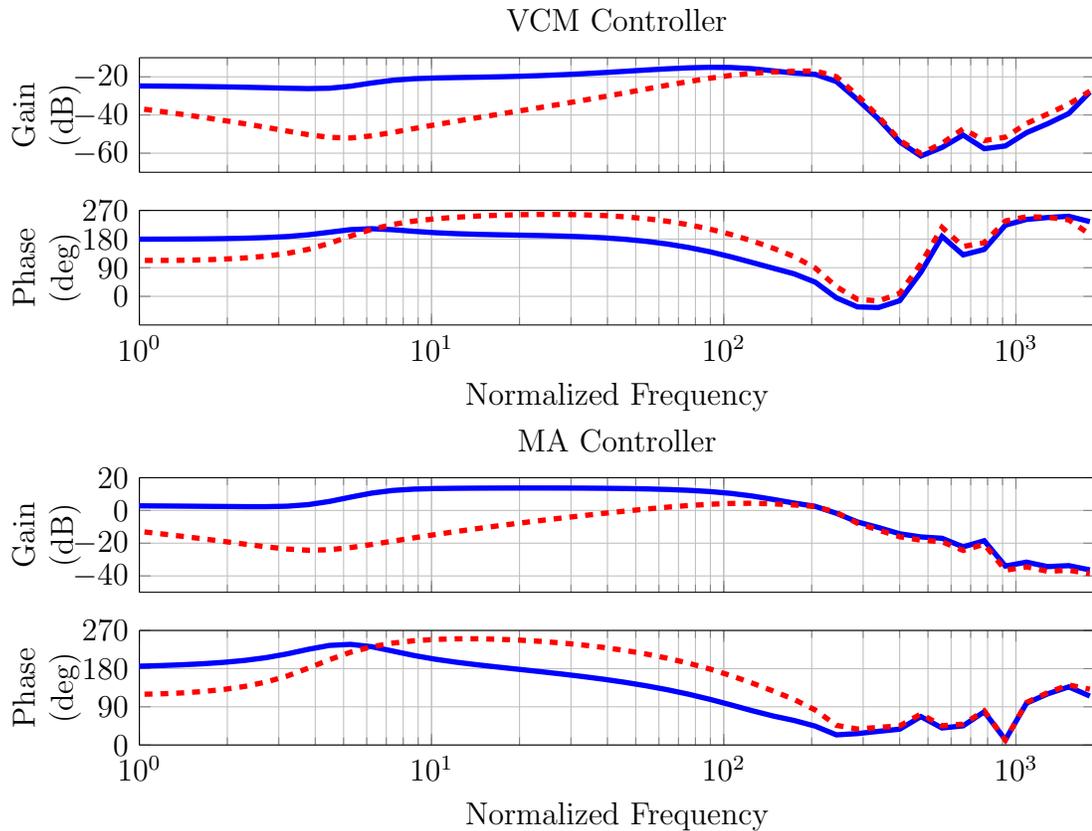


Figure 3.10: The Bode plot of the robust controller designed by D-K synthesis using W_{p1} (solid blue line) and W_{p2} (dashed red line).

The D-K synthesis is available on MATLAB's Robust Control Toolbox as `dksyn`. The controllers that were created using `dksyn` with the designed interconnected models are shown in Figure 3.10. In the figure, the controller designed using W_{p1} and W_{p2} are shown. It can be seen that the controller designed using W_{p1} has a lower gain at the low frequency.

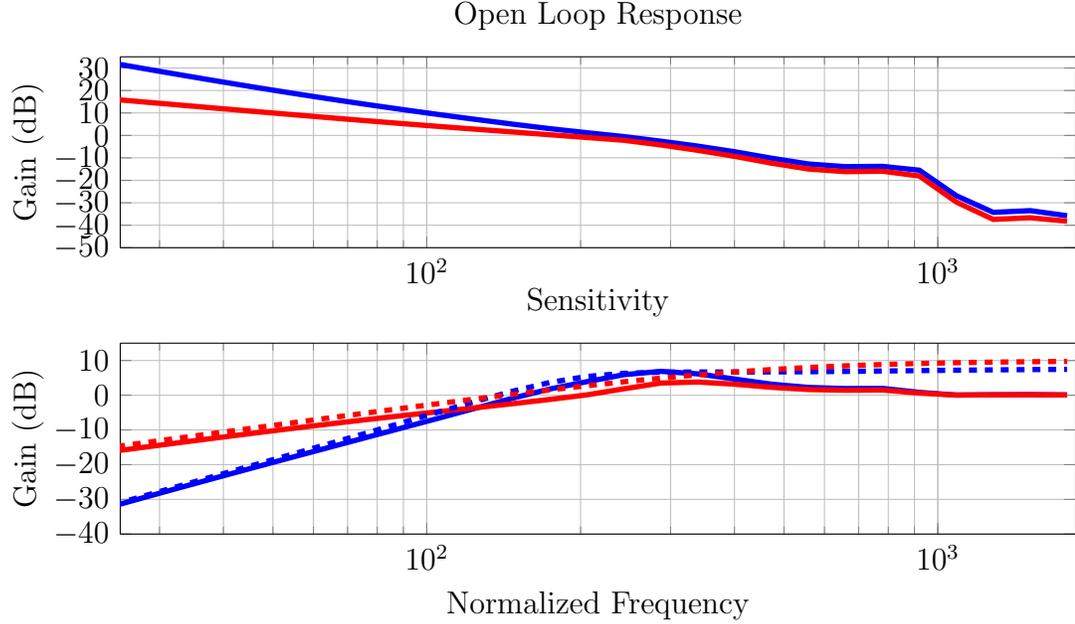


Figure 3.11: The open-loop response and closed-loop sensitivity created from robust controller designed using W_{p1} and W_{p2} . The dashed lines are the inverse of the performance weights.

3.5 Controller Performance

Once the controller is designed using D-K synthesis, the controller performance can be verified by the open-loop response and closed-loop sensitivity. The open-loop response, L , and sensitivity, S , for the HDD system is defined as,

$$L = G_V K_V + G_M K_M \quad (3.9)$$

$$S = \frac{1}{1 + L} \quad (3.10)$$

The open-loop response and the sensitivity of the controllers using the nominal models and experimental data are shown in Figure 3.11 and Figure 3.12, respectively. If the designed controller does not meet the design criteria, the process is repeated with changes made to the models and weights to design a desirable controller. In Figure 3.11, the inverse of the performance weight is also shown with the sensitivity to demonstrate how it affects the shape of the closed-loop system. The trade off in performance caused by the differences in W_{p1} and W_{p2} can be seen in both the figures. It can be seen that the controller designed with W_{p1} lowers the sensitivity peak of the closed-loop system, but has a higher gain below the bandwidth compared to the controller designed with W_{p2} . There are advantages and disadvantages to both of

these designs. The lower sensitivity peak would indicate a better robust stability, while the lower gain below the bandwidth would indicate better disturbance rejection and thus better performance. The performance weight should be chosen based on which characteristic is more favorable for the HDD systems.

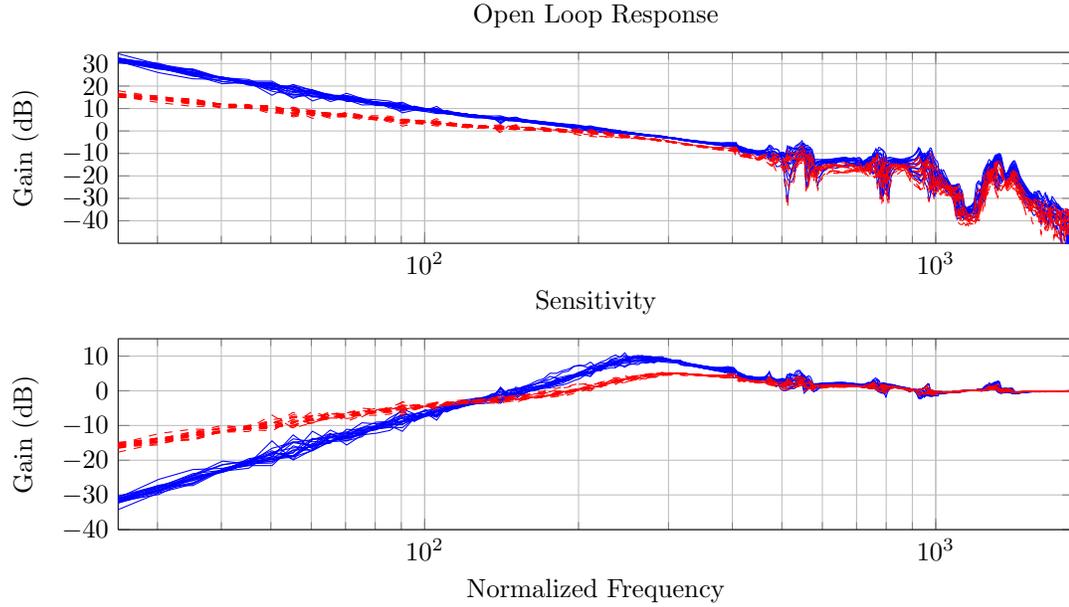


Figure 3.12: The open-loop response and closed-loop sensitivity derived from the experimental data based on a controller designed from W_{p1} (solid blue line) and W_{p2} (dashed red line).

3.6 Controller Implementation

The controllers created in this chapter were validated by implementing them into an actual HDD at Seagate. The controllers typically have to be conditioned such that it can be implemented into the HDDs. For example, there is some limitation on state order of the controller, thus an order reduction technique may be used. Once the controller is implemented into the HDD, a sinusoidal sweep is done to measure the open-loop and closed-loop sensitivity of the HDD for some frequency range. Figure 3.13 shows the sensitivity of a HDD system using the robust controllers which were designed in this chapter. The controller designed using W_{p2} was not a stabilizing controller due to not adequately attenuating the low frequency disturbance. Since the uncertainty model is based off a set of experimental FRD, a controller that may seem to be stabilizing may not necessary be stable for a HDD system. Fortunately, it is not costly to test various controllers within a HDD.

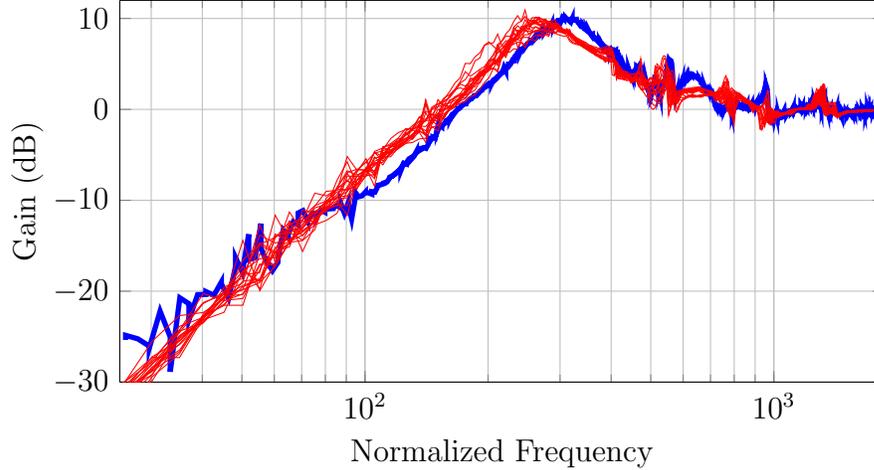


Figure 3.13: The closed-loop sensitivity measured from a real HDD — and simulated from experimental data — using the robust controller designed from W_{p1} .

Within Figure 3.13, the sensitivity simulated using the same controller is shown in red lines. During the controller implementation, dc gain of the MA has to be calibrated for each controller based on the approximate 0dB crossing of the sensitivity. Therefore the sensitivity directly measured from a HDD can have a different frequency response compared to the simulated sensitivity. It can be seen that the sensitivity derived directly from the HDD has a higher bandwidth while maintaining the peak at 10 dB.

3.7 Conclusion

This chapter describes a method that was used to design a LTI robust controller for a HDD system based on a set of experimental FRD. Although the method shown in this chapter is not novel, the chapter provides several insights into designing a robust controller for a HDD system. First, a method of designing an uncertainty model from a set of experimental FRD was shown. Second, since current HDD system is a MISO system, a method of designing a MISO model was shown. Third, a method of evaluating the performance of a controller for a MISO system was shown by measuring the sensitivity of the system. By implementing the controllers onto the HDD, it was shown that a robust controller that may seem stabilizing may not be. One of the controllers designed in this chapter was able to provide an adequate performance while stabilizing the system for a HDD. Lastly, the stabilizing LTI controller designed in this Chapter provides a baseline robust performance for the temperature dependent controller designed in Chapter 6.

Chapter 4

Optimal Uncertainty Modeling

4.1 Introduction

This chapter describes an efficient algorithm to construct an optimal uncertainty model from experimental frequency response data (FRD). The work in this chapter builds upon several results from existing literatures. The most closely related results are contained in [7,49]. Semidefinite programming (SDP) was used in [49] to construct an optimal uncertainty model from a set of FRD accounting for both noise and fitting errors. A similar approach, presented in [7], forms the basis of the Matlab command `ucover`. The `ucover` function computes a minimal uncertainty bound from a given set of FRD and a known nominal model. The nominal model is typically computed by simply averaging the FRD as shown in the previous chapter. Approaches to construct uncertainty models from time domain data have also been presented in [50–53].

This chapter uses convex optimization to construct uncertainty sets from experimental FRD. The proposed algorithm generates both the nominal model and the uncertainty bounds from empirical FRD. It is shown via a simple example that optimizing the nominal model (rather than simply averaging the data) leads to a reduction in the uncertainty bounds. This ultimately reduces the conservatism in the control design. Several practical issues that arise in the construction of uncertainty sets for the dual-stage actuator system are also described.

4.1.1 Problem Statement

The problem formulation assumes that a collection of $n \times m$ frequency responses are obtained from input/output experiments. The k^{th} experimental dataset, ($k = 1, \dots, K$), consists of the complex FRD, $D_k := \{D_k(j\omega_1), \dots, D_k(j\omega_F) \subset \mathbb{C}^{n \times m}\}$, defined on a common grid of frequencies, $\{\omega_f\}_{f=1,2,\dots,F} \subset \mathbb{R}$. This data can be easily and efficiently computed for many heads and drives in the HDD application using a basic sinusoidal frequency sweep. For robust control design it is useful to construct an uncertainty set of linear, time-invariant models, $\mathcal{S} \subset \mathbb{RL}_{\infty}^{n \times m}$, that contains all the FRD. Specifically, the uncertainty set, \mathcal{S} , is said to cover the collection of experimental data if for each frequency response, D_k , there exists a model, $G \in \mathcal{S}$, such that, $G(j\omega_f) = D_k(j\omega_f)$, for all frequencies, $f = 1, \dots, F$. Although as shown in the notation the following method is valid for unstable systems, the HDD is an inherently stable system.

This chapter focuses on two types of non-parametric uncertainty sets: additive and input multiplicative uncertainty. These uncertainty sets are both described by a nominal model, $G_0 \in \mathbb{RL}_{\infty}^{n \times m}$, and stable uncertainty weights, $W_L \in \mathbb{RH}_{\infty}^{n \times n}$, $W_R \in \mathbb{RH}_{\infty}^{m \times m}$, where the dimensions of, W_L , depend on the uncertainty set type. The additive, \mathcal{S}_A , and input multiplicative, \mathcal{S}_M , uncertainty sets are defined as:

$$\mathcal{S}_A := \{G_0 + W_L \Delta W_R : \Delta \in \mathbb{RL}_{\infty}^{n \times m}, \|\Delta\| \leq 1\} \quad (4.1)$$

$$\mathcal{S}_M := \{G_0(I_m + W_L \Delta W_R) : \Delta \in \mathbb{RL}_{\infty}^{m \times m}, \|\Delta\| \leq 1\} \quad (4.2)$$

The dimensions of W_L are $n \times n$ and $m \times m$ for the additive and multiplicative uncertainty sets, respectively. The explicit dependence of the uncertainty set on the nominal model and weights will occasionally be denoted, e.g. $\mathcal{S}_A(G_0, W_L, W_R)$.

For a given type of uncertainty set (additive or input multiplicative), the objective is to construct the nominal model, G_0 and uncertainty weights, W_L and W_R , such that the resulting uncertainty set covers the collection of experimental data. In addition, the “smallest” possible uncertainty set that covers the data should be constructed since this will reduce the conservatism in the robust control design. The following function will be used as a measure for the size of the uncertainty set at the frequency,

ω :

$$h(W_L, W_R, \omega) := \text{Tr}[W_L(j\omega)^* \Gamma_L W_L(j\omega)] + \text{Tr}[W_R(j\omega) \Gamma_R W_R(j\omega)^*] \quad (4.3)$$

where Γ_L and Γ_R are positive definite matrices of appropriate dimensions. These matrices are chosen to emphasize specific directions in the input/output space. They can also be chosen as functions of frequency, ω , to emphasize specific frequency bands. In most cases the simple choices of $\Gamma_L = I$ and $\Gamma_R = I$ provide reasonable results. The following optimization is a formal statement for the uncertainty set construction problem:

$$\begin{aligned} & \min_{G_0, W_L, W_R} \int_0^\infty h(W_L, W_R, \omega) d\omega \\ & \text{subject to: } \mathcal{S}_A(G_0, W_L, W_R) \text{ covers } \{D_k\}_{k=1, \dots, K} \end{aligned} \quad (4.4)$$

The optimization is stated for additive uncertainty sets but a similar optimization can be formulated using an input multiplicative uncertainty set. The following section will describe a numerical algorithm to approximately solve the optimizations for both uncertainty types using semidefinite programming (SDP) [54]. The use on non-parametric uncertainty sets leads to a computationally tractable algorithm for covering the FRD. The approach described in this chapter can be extended to other frequency domain uncertainty sets including output multiplicative, inverse additive, and inverse (input or output) multiplicative models [31, 46].

It is worth noting that if the optimal nominal function, M_0 , is specified and held fixed in the optimization then the results in [7, 49] can be used to construct W_L and W_R . In particular, the algorithm described in [7] forms the basis for the Matlab function `ucover` which constructs uncertainty set weights if the nominal model and the FRD is given. A contribution of this chapter stems from developing an algorithm that addresses the practical issues that arise in creating uncertainty models from FRD of dual-stage actuator HDDs.

4.2 Numerical Algorithm

The constraint in Equation 4.4 can be reformulated in terms of a frequency-dependent matrix inequality via the following lemma.

Lemma 1 ([7]) Let $\mathcal{S}_A(G_0, W_L, W_R)$ be an additive uncertainty set (Equation 4.1) defined by a nominal model $G_0 \in \mathbb{RL}_\infty^{n \times m}$ and stable uncertainty weights $W_L \in \mathbb{RH}_\infty^{n \times n}$ and $W_R \in \mathbb{RH}_\infty^{m \times m}$. In addition, assume W_L and W_R have stable inverses. Then any $\bar{G} \in \mathbb{RL}_\infty^{n \times m}$ satisfies $\bar{G} \in \mathcal{S}_A(G_0, W_L, W_R)$ if and only if

$$\begin{bmatrix} W_L W_L^* & \bar{G} - G_0 \\ (\bar{G} - G_0)^* & W_R^* W_R \end{bmatrix} (j\omega) \geq 0 \forall \omega \quad (4.5)$$

This is a minor variation of Theorem 1 in [7].

Based on Lemma 1 the optimization in Equation 4.4 can be approximated on the frequency grid as:

$$\min_{M_0, W_L, W_R} \sum_{f=1}^F h(W_L, W_R, \omega_f) \quad (4.6)$$

subject to:

$$\begin{bmatrix} W_L W_L^* & D_k - M_0 \\ (D_k - M_0)^* & W_R^* W_R \end{bmatrix} (j\omega_f) \geq 0$$

$f = 1, \dots, F$ and $k = 1, \dots, K$

where $M_0(j\omega_f)$ is the optimal nominal frequency response data defined on the same frequency grid as $D_k(j\omega_f)$. $M_0(j\omega_f)$ is used to design $G_0(s)$ further in the chapter. The constraints and objective function involve product terms $W_L W_L^*$ and $W_R^* W_R$. By defining two new variables $L := W_L W_L^*$ and $R := W_R^* W_R$, the optimization problem can now be expressed as a finite-dimensional SDP in terms of these new variables:

$$\min_{M_0, L, R} \sum_{f=1}^F \text{Tr}[\Gamma_L L + \Gamma_R R](j\omega_f) \quad (4.7)$$

subject to:

$$\begin{bmatrix} L & D_k - M_0 \\ (D_k - M_0)^* & R \end{bmatrix} (j\omega_f) \geq 0$$

$f = 1, \dots, F$ and $k = 1, \dots, K$

The decision variables in this optimization are the complex matrices $L(j\omega_f)$, $R(j\omega_f)$, and $M_0(j\omega_f)$ defined at each frequency gridpoint. It is important to note that the

cost function and constraints contain no coupling across the frequency gridpoints. Thus this optimization trivially decouples into F smaller SDP problems, one for each frequency gridpoint. Furthermore, a set of complex matrix LMIs must be converted into real matrix LMIs before a numerical solver such as LMILAB in MATLAB can be used. The SDP can be transformed from complex matrices to real matrices using a standard complex to real transformation for LMIs [55].

Equation 4.7 is a finite dimensional convex optimization that can be used to jointly compute the optimal nominal model and the uncertainty weights defined on a frequency grid. Many control design and analysis methods require a state-space or transfer function model rather than simply the frequency response defined on a frequency grid. Computational steps are described below to obtain state-space systems for the nominal model and weights. The steps are described for SISO systems $n = m = 1$ and the extension to MIMO systems is discussed in Section 4.3.

1. Solve Equation 4.7 for $\{M_0(j\omega_f)\}_{f=1}^F$, $\{L(j\omega_f)\}_{f=1}^F$, and $\{R(j\omega_f)\}_{f=1}^F$. This decouples Equation 4.7 as F independent SDPs that can be solved with available software, e.g. LMILab in Matlab.
2. A state-space model for the nominal dynamics $G_0(s)$ is fit to the optimal response $\{M_0(j\omega_f)\}_{f=1}^F$ obtained in Step 1. This can be done in Matlab using the `fitfrd` function. The order of the state-space model is chosen by the user to obtain a trade-off between model complexity and fitting accuracy.
3. The state-space model $G_0(s)$ obtained in Step 2 is substituted into Equation 4.7 and the optimization is resolved for $\{L(j\omega_f)\}_{f=1}^F$, and $\{R(j\omega_f)\}_{f=1}^F$. This can be done in Matlab using the `ucover` function. Step 3 reconstructs the uncertainty weights on the frequency grid to account for any error in fitting the state space nominal model.
4. For SISO systems, $L := |W_L|^2$ and $R := |W_R|^2$. Hence the $L(j\omega_f)$ and $R(j\omega_f)$ obtained in Step 3 specify the magnitudes of the uncertainty weights required to cover the experimental FRD. Stable, minimum phase transfer functions $W_L(s)$ and $W_R(s)$ are constructed that satisfy $|W_L(s)| \geq \sqrt{L(j\omega_f)}$ and $|W_R(s)| \geq \sqrt{R(j\omega_f)}$. This fitting step can be performed in Matlab using the `fitmagfrd` function. Constructing the state-space weights in this fashion ensures that $\mathcal{S}_A(G_0, W_L, W_R)$ will cover the data. Moreover, the set \mathcal{S}_A is unaffected by

the phase of the uncertainty weights and hence the restriction to minimum phase W_L and W_R is without loss of generality. Finally, the optimal additive uncertainty model is given by $\mathcal{S}_A(G_0, W_L, W_R)$.

An input multiplicative uncertainty set can be constructed starting with the optimization problem shown in Equation 4.4 with \mathcal{S}_A replaced by \mathcal{S}_M . The constraint in this optimization can be equivalently expressed as a frequency-dependent matrix inequality using Theorem 3 in [7]. This leads to a finite-dimensional optimization of the form:

$$\min_{M_0, W_L, W_R} \sum_{f=1}^F f(W_L, W_R, \omega_f) \quad (4.8)$$

subject to:

$$\begin{bmatrix} M_0 W_L W_L^* M_0^* & D_k - M_0 \\ (D_k - M_0)^* & W_R^* W_R \end{bmatrix} (j\omega_f) \geq 0$$

$$f = 1, \dots, F \text{ and } k = 1, \dots, K$$

It is important to note that nominal model appears as products with itself and the left uncertainty weight in the upper left block of the matrix constraint. Hence this optimization is not jointly convex in (M_0, W_L, W_R) as written. As before the products of the uncertainty weights can be handled by introducing the new variables $L := W_L W_L^*$ and $R = W_R^* W_R$. Further introduce $Q_0 := M_0^{-1}$ and multiply the matrix constraint on the left and right by $\text{diag}(M_0^{-1}, I)$ and $\text{diag}(M_0^{-*}, I)$. This leads to the following (convex) SDP problem:

$$\min_{Q_0, L, R} \sum_{f=1}^F \text{Tr}[\Gamma_L L + \Gamma_R R](j\omega_f) \quad (4.9)$$

subject to:

$$\begin{bmatrix} L & Q_0 G_k - I \\ (Q_0 G_k - I)^* & R \end{bmatrix} (j\omega_f) \geq 0$$

$$f = 1, \dots, F \text{ and } k = 1, \dots, K$$

The numerical steps to solve for the nominal model and uncertainty weights for the input multiplicative model are essentially the same as those given above for the additive uncertainty model. The only additional detail is that the nominal model on the

frequency grid $\{M_0(j\omega_f)\}_{f=1}^F$ is obtained by inverting the values of $Q_0(j\omega_f)$ computed from the optimization.

4.3 Practical Issues

4.3.1 Incorporating Prior Knowledge

In many applications there is some prior knowledge regarding the nominal system dynamics. For example the voice coil motor dynamics has a double integrator characteristic, and the micro actuator has a second order high frequency roll off [56]. Let $P(s) \in \mathbb{RL}_\infty$ denote any known characteristics of the nominal model. This prior knowledge is incorporated by constructing a nominal model of the form $G_0(s) = P(s)T_0(s)$ where $T_0(s)$ is the unknown component to be determined. To construct the nominal model, first transform the experimental frequency responses $\{D_k\}_{k=1}^K$ into equivalent frequency responses for the unknown component:

$$T_k(j\omega_f) = P^{-1}(j\omega_f)D_k(j\omega_f) \forall k, f \quad (4.10)$$

Next, use the optimization method described previously to construct T_0 and weights such that the corresponding uncertainty set covers $\{T_k\}_{k=1}^K$. The optimal nominal model on the frequency grid $M_0(j\omega_f)$ is then given by multiplying the known and unknown system components, i.e. $M_0(j\omega_f) = P(j\omega_f)T_0(j\omega_f) \forall f$. Then the process can continue on to step 2 listed in Section 1.2.

4.3.2 Limiting Magnitude of Nominal Model Derived From Multiplicative Uncertainty Set

The uncertainty set optimization attempts to minimize the magnitude of the weights. In some cases this formulation leads to impractical results for multiplicative uncertainty sets. To illustrate the issue consider a SISO multiplicative uncertainty set $\mathcal{S}_M(G_0, W_L, 1)$. Systems in this set have the form $G_0(1 + W_L\Delta)$ where $\|\Delta\| \leq 1$. Note that this set can cover any collection of frequency responses by choosing the nominal model to have sufficiently large magnitude and $\|W_L\| = 1$. Thus an optimization to construct the nominal model G_0 and W_L will never result in an uncertainty weight that exceeds 1 in magnitude, i.e. 100% multiplicative uncertainty is an upper bound on the optimal weight. The practical consequence is that the optimization will return

a very large nominal model and an uncertainty weight of magnitude near 1 for any frequency where the responses have "large" spread. In particular, this will occur for any frequency such that $\max_{k,l} |D_k(j\omega_f) - D_l(j\omega_f)| > 2$. The optimization in Equation 4.9 can be modified to add one additional constraint that prevents the nominal model from growing too large:

$$|Q_0(j\omega_f)| \geq \min_k |D_k(j\omega_f)^{-1}| \quad (4.11)$$

The crux of the issue is that the actual "size" of the multiplicative uncertainty set is $|G_0W_L|$ and the posed optimization only attempts to minimize $|W_L|$. It is not possible to directly minimize $|G_0W_L|$ in a computationally efficient manner as this is a non-convex objective.

An alternative procedure to construct a multiplicative uncertainty set for a SISO system is to first compute an additive uncertainty set and then compute the corresponding weights for a multiplicative model. For a SISO system the uncertainty weights and nominal model are commutable and thus the additive uncertainty weight and multiplicative uncertainty weight are related by,

$$W_A = W_M G_0 \quad (4.12)$$

where W_A is the additive uncertainty weight and W_M is the multiplicative uncertainty weight. Typically the multiplicative uncertainty set is favored over additive uncertainty set because it is more intuitive to visualize the uncertainty as a relative error than an absolute error.

4.3.3 Application to MIMO Systems

The optimization technique described in the previous sections can be extended to MIMO systems. Based on the type of uncertainty set the optimization is based on, the input and output size of the uncertainty block will be different. For example a two-input one-output system, like the HDD system, would have an uncertainty block size of (1×2) , (1×1) , and (2×2) for additive, output multiplicative and input multiplicative uncertainty set, respectively.

The different uncertainty block size indicates that the uncertainty of the system would interact differently based on the type of uncertainty that is chosen. Again take a two-

input one-output system. The additive uncertainty model for this system would be defined as,

$$\begin{aligned}
G &= G_0 + W_L \Delta W_R \\
&= G_0 + W_L \begin{bmatrix} \Delta_1 & \Delta_2 \end{bmatrix} \begin{bmatrix} W_{R1} & 0 \\ 0 & W_{R2} \end{bmatrix} \\
&= G_0 + \begin{bmatrix} W_L \Delta_1 W_{R1} & W_L \Delta_2 W_{R2} \end{bmatrix}
\end{aligned} \tag{4.13}$$

While the input multiplicative uncertainty model would be defined as,

$$\begin{aligned}
G &= G_0(I + W_L \Delta W_R) \\
&= G_0 \left(I + \begin{bmatrix} W_{L1} & 0 \\ 0 & W_{L2} \end{bmatrix} \begin{bmatrix} \Delta_1 & \Delta_2 \\ \Delta_3 & \Delta_4 \end{bmatrix} \begin{bmatrix} W_{R1} & 0 \\ 0 & W_{R2} \end{bmatrix} \right) \\
&= G_0 \left(I + \begin{bmatrix} W_{L1} \Delta_1 W_{R1} & W_{L1} \Delta_2 W_{R2} \\ W_{L2} \Delta_3 W_{R1} & W_{L2} \Delta_4 W_{R2} \end{bmatrix} \right)
\end{aligned} \tag{4.14}$$

It can be seen that the interaction of the uncertainty weights are different based on the type of uncertainty block that is used. Therefore the optimal nominal that is derived for a MIMO system will be different for each dynamic uncertainty type.

Another type of uncertainty block that could be used for a HDD system is a simplified version of the input multiplicative uncertainty where the uncertainty block is a diagonal system such that,

$$G = G_S \begin{bmatrix} G_V & 0 \\ 0 & G_M \end{bmatrix} \left(I + \begin{bmatrix} W_{L1} & 0 \\ 0 & W_{L2} \end{bmatrix} \begin{bmatrix} \Delta_1 & 0 \\ 0 & \Delta_2 \end{bmatrix} \begin{bmatrix} W_{R1} & 0 \\ 0 & W_{R2} \end{bmatrix} \right) \tag{4.15}$$

In this case, the optimal nominal model and uncertainty weight for the VCM and MA can be derived as independent SISO systems because there are no off-diagonal uncertainty terms.

Some simplifications are typically required in the state-space fitting steps to create a MIMO nominal model. For example `fitfrd` in Matlab applies only for vector systems (either one input or one output). Thus in the case of a MIMO system, one possible method is to fit a state-space model to the data one column or one row at a time. This method will naturally increase the order of non-vector MIMO system, thus utilizing

some order reduction technique may be useful. For the MISO system like the HDD, it is beneficial to fit the system as a vector system compared to modeling each input-output pair as SISO systems to reduce the state order of the system. However to achieve a good fit, each input-output pair of the system should share similar poles. For the uncertainty weights the fitting process is simplified by restricting the weights, a priori, to be diagonal.

4.4 Simple Numerical Example

In this example, a stable second order system with uncertainties in its natural frequency and damping ratio is used to demonstrate the approach described in the previous section. First a transfer function with uncertainty is defined as,

$$G_{true}(s) = \frac{p^2}{s^2 + 2\zeta sp + p^2} \quad (4.16)$$

where $\zeta = [0.025, 0.1]$ and $p = [3, 7]$. In a real application the true underlying model, specified here by G_{true} would not be known. The "true" model is defined here to generate the FRD that is used in the proposed model uncertainty construction procedure. An uncertainty set such as $\mathcal{S}_A(G_0, W_L, 1)$ does not precisely capture the parametric uncertainties that appears in the true model. Instead, \mathcal{S}_A covers (over approximates) the parametric uncertainties. A parameterized form for the "true" model is unknown in some cases and hence the uncertainty set \mathcal{S}_A constructed using only input/output data is still useful.

Experimental data $\{D_k\}_{k=1}^K$ was simulated by taking $K = 200$ random samples of ζ and p with a frequency grid containing 100 frequency points from $\omega_1 = 10^{-1}$ rad/sec to $\omega_{100} = 10^2$ rad/sec. Since the magnitude difference between the different $D_k(j\omega_f)$ was too large, the nominal model was constructed using the LMI constraints for an additive uncertainty and then the multiplicative uncertainty weight for the $M_0(j\omega_f)$ was constructed using the method described in section 4.3.2. A second nominal model was created by taking the average of $D_k(j\omega_f)$ and utilizing `ucover` to create the $W_L(j\omega_f)$ for this model. The following results are solutions based on the frequency grid, ω_f , and no state-space fitting has been done to compare the two different methods without additional steps. With 200 samples and 100 frequency points, the convex optimization algorithm took roughly 65 seconds to run on a modern desktop on MATLAB. The average of plant models is not necessarily a good method to design a nominal model

for robust controller design, thus this example was created to show the advantage of the optimal nominal FRD created from the LMI constraints.

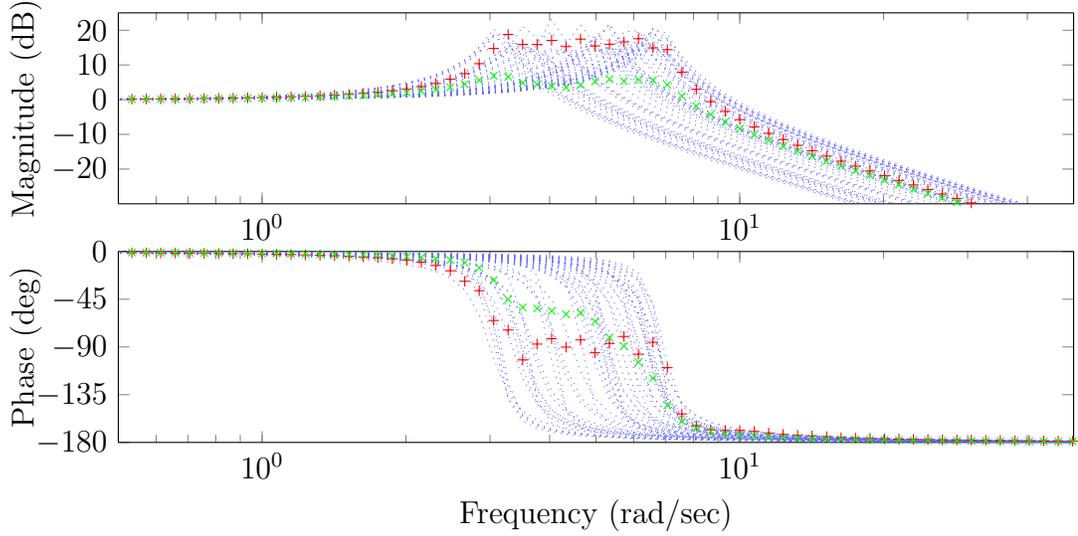


Figure 4.1: Bode plot of sampled models \dots , optimal values $+$ and the mean \times .

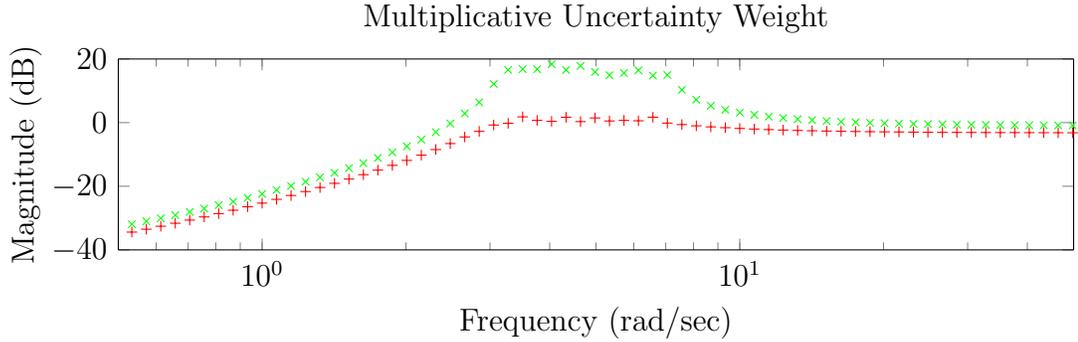


Figure 4.2: The minimum multiplicative uncertainty weight required to cover all sampled data for the optimal nominal $+$ and the mean \times .

Figure 4.1 shows the Bode plot of $\{D_k(j\omega_f)\}_{k,f}$, the average of the data, and the optimal FRD from the LMI method. Since there are uncertainties in both ζ and p values, the frequency and magnitude of the peak are different for each sample. Since only 1 peak exists for each $D_k(j\omega_f)$, there are more FRD without a large peak at some frequency ω_f . Therefore the gain of the mean at ω_f is naturally reduced by data points without a peak at ω_f . The LMI method finds a complex value at each frequency that minimizes its distance from all of $D_k(j\omega_f)$. Therefore the quantity of $D_k(j\omega_f)$ without a peak at ω_f does not affect its gain. The phase of the FRD between the two methods have different characteristics as well. The LMI method derives a

FRD with phase around -90 degrees from 3 to 7 rad/sec, which is in the middle of the different possible phases from -180 to 0 degrees. While the average model has a phase that is influenced by the majority of the phase value and is closer to -45 degrees from 3 to 7 rad/sec. The differences in gain and phase between these two FRD results in different optimal uncertainty weights as shown in Figure 4.2.

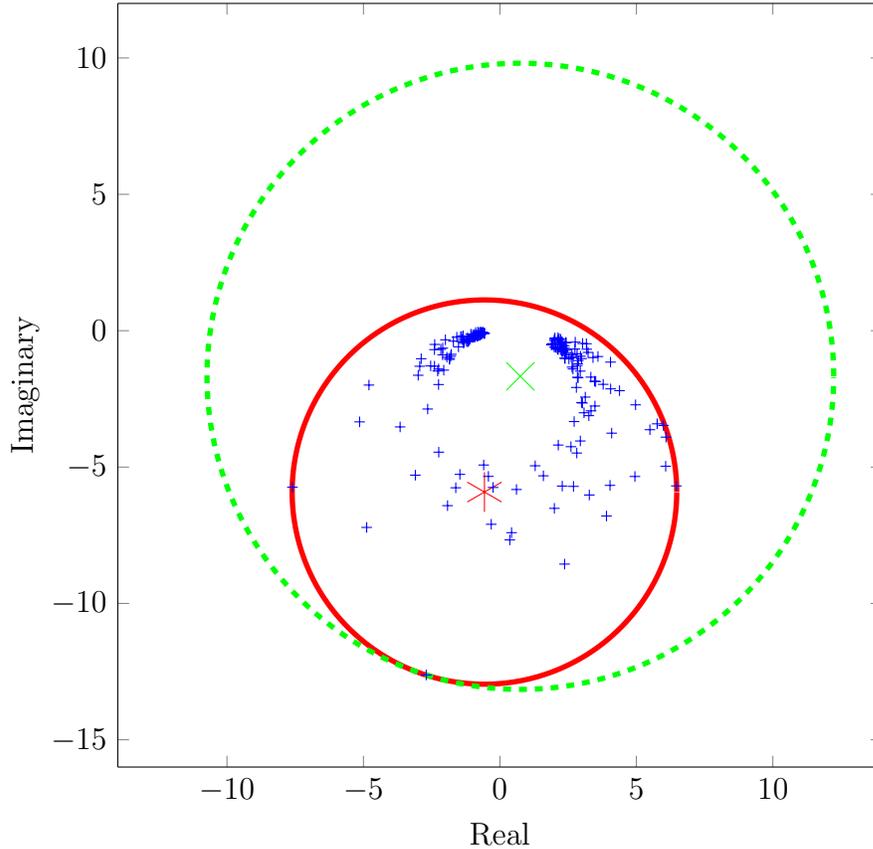


Figure 4.3: The smallest circle required for the LMI method, —, and the averaging method, - - -, to cover the sampled data points + from the nominal point created by LMI method x and averaging method x at $\omega=5\text{rad/sec}$ are shown.

It can be seen from Figure 4.2 that the gain of the uncertainty weight required for the mean of $D_k(j\omega_f)$ is greater than the uncertainty weight for the optimal FRD. For a SISO system the LMI constraint finds a complex number at each frequency that will require a circle with the smallest radius to cover all of the data points. This point is exemplified in Figure 4.3, which shows a scatter plot of the simulated data, two nominal data points, and the smallest circle centered at each of the nominal data points required to cover all of the data points at $\omega = 5 \text{ rad/sec}$. At this frequency many of the data points has a small gain, and thus they are close to the origin. Due to

the large number of points near the origin, the average of the data points is naturally near the origin as well. However, one of the experimental data has a large peaks at 5 rad/sec and is represented by points at (-3,-12) and (-7,-6). These points need to be contained by the circle to fully define the uncertainty of the system and thus the average of the data points requires a much larger circle compared to the point derived by the LMI method. The nominal point created by the averaging method requires a circle with a radius of 11.5 to cover all of the data points, while the nominal point created using the LMI method requires a circle with a radius of 7.0 to cover all of the data. Compared to the average, the optimal nominal requires an uncertainty weight with smaller gain to cover the FRD, which should result in a controller with better performance.

4.5 Application of Optimization Method on HDD Data

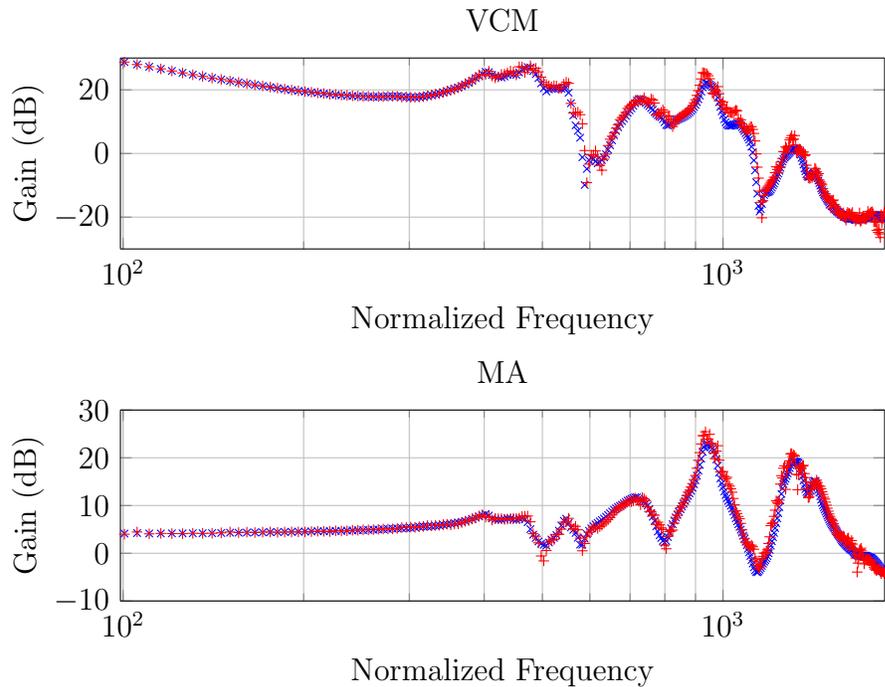


Figure 4.4: The Bode plot of the average of the data \times and the optimal FRD based on the LMI method $+$.

The convex optimization method was used on the same FRD of the HDD that was shown in Chapter 3. These results are based on the frequency grid solution, and no state-space fitting has been done. The solution from the optimization problem is compared to the average of the data as shown in Figure 4.4. It can be seen from the Bode plot that the two FRDs are very similar and the similarity is also reflected in

the required gain for the additive uncertainty weight as shown in Figure 4.5.

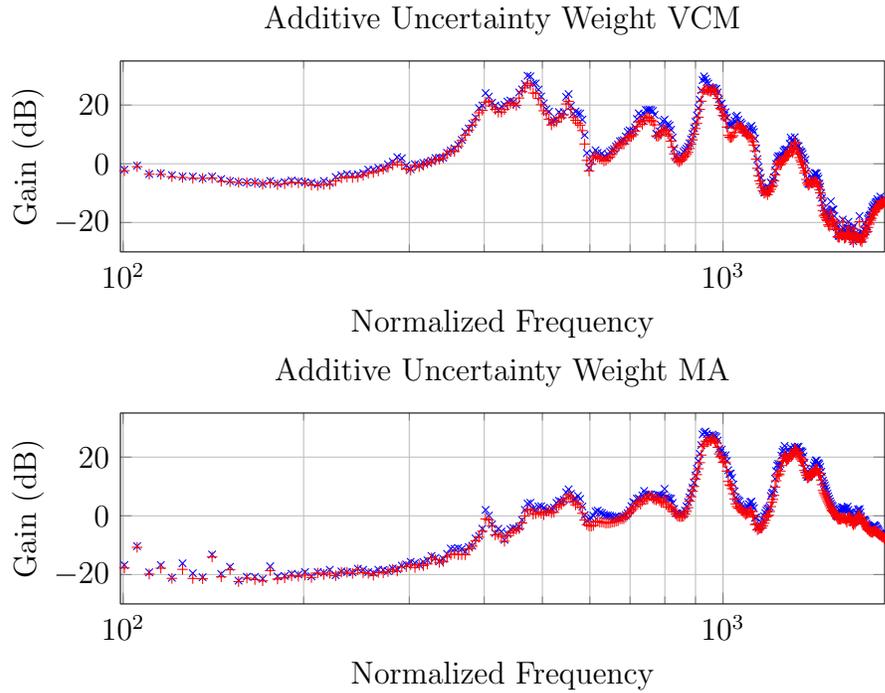


Figure 4.5: The optimal additive uncertainty weight required to cover the data for the average of the data \times and the optimal nominal model $+$.

Figure 4.5 shows the optimal additive uncertainty weight for two of the methods. Similar to the nominal FRD, there is a small difference between the two additive uncertainty weights. It can be seen that the optimal uncertainty weight required for the optimal FRD has a slightly less gain at all frequency compared to the optimal uncertainty weight required for the average. Although for this data set there is a small difference between the average of the data and the optimal FRD, the average does not always guarantee an optimal uncertainty model as shown in the previous section. The optimal FRD is designed based off of the metric of minimizing the optimal uncertain weight, and thus should be used over the average to guarantee that there is less conservatism in the model design.

The optimization problem derives the optimal nominal FRD defined on ω_f . As previously stated, most controller synthesis method typically requires a state-space model or transfer function of the system. In Chapter 3, a simple method of modeling the two actuators as separate SISO system was described. In the following sections other methods of modeling the HDD system are described.

4.5.1 Designing MISO Models

It is possible to model the HDD system as a MISO system. The benefit of this method is that it can reduce the number of state orders required to model the system, which in turn will reduce the state order of the controller. This method also models the physical system more accurately as the VCM and MA are connected to each other.

A row vector state-space system can be fit to a MISO system using a function such as `fitfrd`. To achieve a good fit the two systems must share common poles. Therefore the system needs to be defined such that,

$$G_0 = G_S \begin{bmatrix} G_V & 0 \\ 0 & G_M \end{bmatrix}, \quad (4.17)$$

where $G_V \in \mathbb{RL}_{\infty}^{(1 \times 1)}$ is the VCM model component, $G_M \in \mathbb{RL}_{\infty}^{(1 \times 1)}$ is the MA model component, and $G_S \in \mathbb{RL}_{\infty}^{(1 \times 2)}$ is the model component shared between the VCM and MA. The block diagram for this system is shown in Figure 4.6.

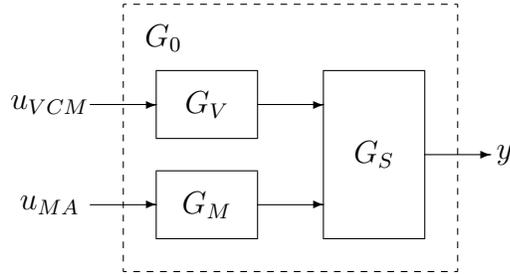


Figure 4.6: The block diagram of the HDD system using MISO state-space model.

The characteristic that is distinctly different between the VCM and the MA are the second order roll offs, thus G_V is a double integrator like characteristic, and G_M is a high frequency roll off. These components are divided out from their respective data. Then the optimization algorithm from this chapter is used to find the optimal FRD, M_S , to create the optimal nominal model, G_S . For this section, the optimal nominal was based on an input multiplicative uncertainty weight with diagonal uncertainty block as shown in Equation 4.5. With this type of uncertainty, the optimal nominal model from Section 4.5 based on SISO system can be used as there are no off-diagonal terms in the uncertainty block. A tenth order MISO state-space system was fit onto the optimal FRD using `fitfrd` to derive G_S . The resulting fourteenth order MISO model is shown in Figure 4.7. By separating the second order roll off from the optimal FRD, `fitfrd` is able to fit a state-space system well.

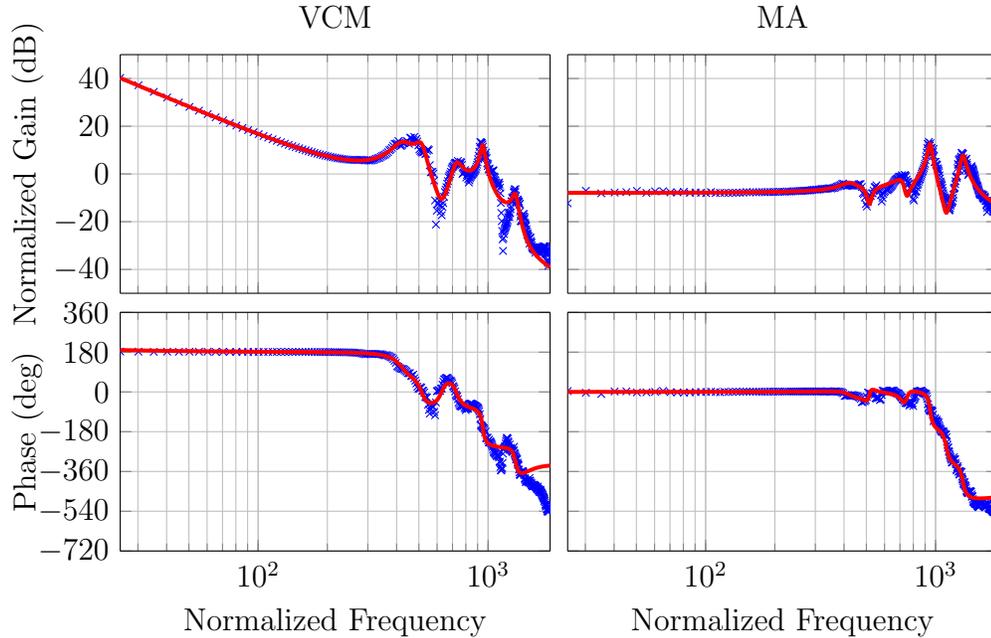


Figure 4.7: The FRD of optimal models \times and MISO state space system — .

4.6 Conclusion

A practical method of creating an uncertainty set from a set of experimental FRD was presented in this chapter. The numerical algorithm was derived through definitions of additive and input multiplicative uncertainty sets. Furthermore this method was tailored to the practical application of robust controller synthesis for a dual-stage actuator of a hard disk drive. A simple example was provided to show the benefits of the optimization method compared to averaging the data. For this particular set of experimental HDD FRDs, which included the HDD dynamics for sixteen drives measured at six different temperature points, the optimization method provided minimal advantage over the averaging of the data. It is advisable, however, to use the optimization method to guarantee that the conservatism of the uncertainty model is minimized. In the next chapter, the temperature dependent models are designed using the optimal nominal FRD.

Chapter 5

Temperature Dependent Modeling

5.1 Introduction

One environmental factor that can affect the dynamic response of hard disk drive (HDD) system is temperature. As the temperature of the system increases, the natural frequency of the actuators decreases as the metal softens and its elasticity increases. In Chapter 3, a method for designing a linear time invariant model was shown using a set of frequency response data (FRD). This set of data was a collection of frequency responses taken from different drives at various temperature points. In this chapter, sets of FRDs are organized into different temperature points at which the FRD was measured. Each set is used to design a temperature specific model, and through these models, a temperature dependent model is created. In Chapter 6, temperature dependent models of the VCM and MA will be used to design a temperature dependent controller.

One assumption made in this chapter is that the temperature of the HDD is a slowly varying parameter. This assumption is made because a HDD is typically used indoors in a controlled environment where the temperature does not change rapidly. Based on this assumption, the temperature dependent model is designed with no dependency on the temperature rate of change. Thus the temperature dependent model that is designed in this chapter is a parameterized LTI model, which is a special case of an linear parameter varying (LPV) model.

One method of characterizing a nonlinear system that is dependent on certain pa-

rameters is to design an LPV model. Typically an LPV model is defined such that,

$$\dot{x} = A(\rho(t))x + B(\rho(t))u \quad (5.1)$$

$$y = C(\rho(t))x + D(\rho(t))u \quad (5.2)$$

where $\rho(t)$ is a parameter that depends on time. The value of $\rho(t)$ is not known ahead of time, however it is assumed that the system has a method of measuring the parameter in real time [57]. LPV theory is a formalism of ad-hoc gain-scheduling system which is used in various industrial applications. In aerospace engineering research, aircraft systems have been characterized using LPV models to design an LPV controller [58]. It is possible that some systems are simple enough that an LPV model can be directly derived using existing models of the system [59–62]. For example a transfer function of a mass spring system can be modified into an LPV model by making the stiffness of the spring be dependent on a parameter such as temperature. An LPV model can also be designed by deriving a nonlinear model of the system and using Jacobian Linearization at various trim points to create a state-space model that is linear between different equilibrium points [63–66]. A nonlinear system is expressed as an LPV model for the purpose of designing an LPV controller.

The experimental FRD is used to design a temperature dependent model with a state order of six for the voice coil motor (VCM) and ten for the micro actuator (MA). Using an accurate mathematical model of the HDD system would result in a controller with large state-order that can not be implemented into the system. As shown in Chapter 3, the robust controller is designed such that the MA system has higher control effort at high frequency compared to the VCM. Therefore the MA model is designed to characterize more high frequency modes than the VCM model. In this chapter, the method of designing a temperature dependent model is demonstrated using the experimental FRD of the MA system, however the same method was applied to both the VCM and the MA. The available experimental FRDs for the VCM and MA were organized into sets of data according to the temperature at which they were measured. The convex optimization method described in Chapter 4 is used to design an optimal nominal FRD at each temperature point. Then state-space models are designed for each temperature point using a method that keeps their states consistent. The states must be kept consistent such that the state-space matrices can be linearized. Specifically a linear fit of the state-space matrices are used to create the temperature dependent model.

5.2 Problem Formulation

The problem formulation assumes that a collection of $n \times m$ frequency responses are obtained at l^{th} temperature point ($l = 1, \dots, L$) from input/output experiments. The k^{th} experimental dataset ($k = 1, \dots, K$) consists of the complex FRD $D_k(T_l) := \{D_k(j\omega_1, T_l), \dots, D_k(j\omega_F, T_l) \in \mathbb{C}^{2 \times 1}\}$ defined on a common grid of frequencies $\{\omega_f\}_{f=1,2,\dots,F} \subset \mathbb{R}$. This data can be easily and efficiently computed for many drives in the HDD application using a basic sinusoidal frequency sweep inside a temperature chamber.

Figure 5.1 and 5.2 shows the Bode plot of the experimental FRD from multiple HDD at internal temperatures of 16.3°C and 59.6°C . The first several modes occur from normalized frequency from 350 to 600 as indicated by the black dotted lines. As stated earlier, the natural frequency of the modes decreases as the temperature increases and the damping ratio changes as well. Temperature specific models are designed for the VCM and MA using these experimental FRDs.

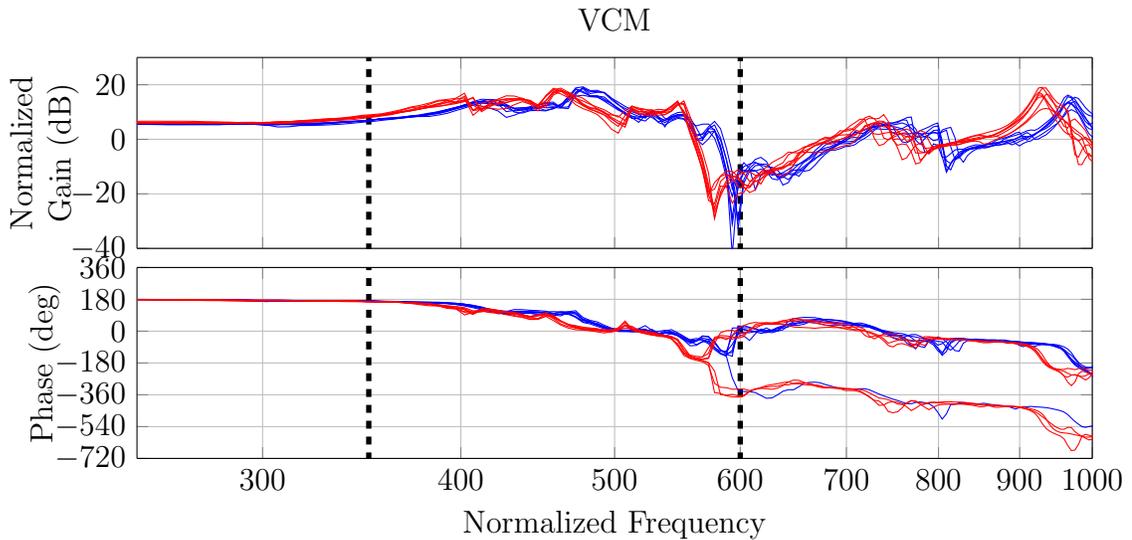


Figure 5.1: The experimental FRD of HDD data at 16.3°C , —, and 59.6°C — for the VCM. The range from frequency of 350 to 600 is where the first several high frequency modes occur, ■■■.

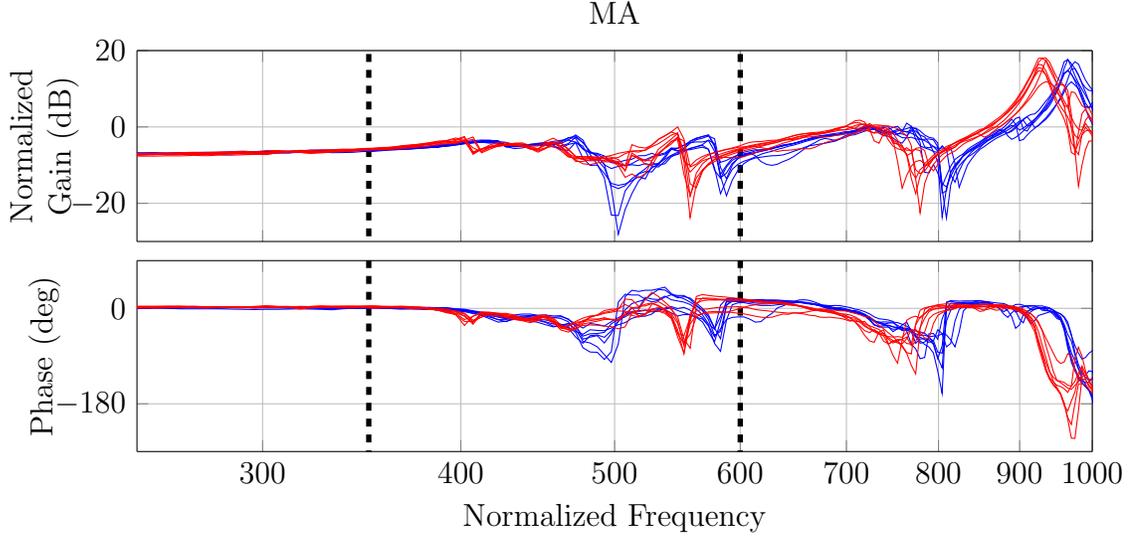


Figure 5.2: The experimental FRD of HDD data at 16.3°C, —, and 59.6°C, —, for the MA. The range from frequency of 350 to 600 is where the first several high frequency modes occur, ■■■.

5.3 Temperature Dependent Modeling

A temperature dependent model is defined as,

$$\begin{aligned} \dot{x} &= A(T)x + B(T)u \\ y &= C(T)x + D(T)u \end{aligned} \quad (5.3)$$

where T is the temperature and the state-space matrices are linearly dependent on temperature such that

$$A(T) = A_0 + (T - T_0)A_1, \quad (5.4)$$

where T_0 is a reference temperature point, and A_0 and A_1 are constant matrices. Since it is assumed that the temperature is a slow varying parameter, the temperature rate of change is neglected in this model. The reference temperature point was selected as $T_0 = 16.3^\circ\text{C}$. This is the lowest temperature at which experimental FRD was measured.

5.3.1 Benefits of Temperature Dependent Modeling For HDD System

Experimental FRDs were collected for the VCM and MA on 16 different HDDs at chamber temperature of $\{5, 15, 25, 35, 45, 55\}^\circ\text{C}$. This yields a total of 72 frequency responses (6 temperatures \times 16 drives). These chamber temperatures corresponded

to internal HDD temperature of $\{16.3, 20.9, 30.6, 40.1, 50.0, 59.6\}^{\circ}\text{C}$, which were measured through the HDD’s internal sensor. Each temperature point corresponds to the index l as shown in Table 5.1. Using the method from Chapter 4, an optimal

l	1	2	3	4	5	6
Chamber Temperature ($^{\circ}\text{C}$)	5	15	25	35	45	55
Internal Temperature ($^{\circ}\text{C}$)	16.3	20.9	30.6	40.1	50.0	59.6

Table 5.1: The temperature of the HDD for index l .

nominal FRD, $M_{T_l}(j\omega_f)$ was derived for the HDD at each temperature point. Then using the optimal FRD, the optimal additive uncertainty weight required to cover the uncertainty of the system was derived. The additive uncertainty weight for a SISO system represents the longest distance from the nominal model to the set of experimental data. In other words, they represent the size of the uncertainty at each frequency point.

Figure 5.3, shows the optimal additive uncertainty weight, $W_{opt}(M, D_k, \omega_f)$, based on temperature independent and temperature specific optimal nominal FRD, $M_{T_{ind}}(j\omega_f)$ and $M_{T_l}(j\omega_f)$ respectively. The frequency was normalized by some value for proprietary reasons. The first several modes of the VCM and MA systems occur in the normalized frequency of 350 to 600, as shown in the dotted black lines in Figure 5.3. It can be seen that the required uncertainty weight can be reduced using temperature dependent models. At a normalized frequency of 400, the optimal uncertainty weight gain is about 10 dB less for the temperature specific systems from 16.3°C to 30.6°C . At a normalized frequency of 500, the optimal uncertainty weight gain is about 5dB less for the temperature specific systems from 20.9°C to 59.6°C . Figure 5.3 shows that it is possible to design a temperature dependent model that can reduce the uncertainty of the system at the frequency of first several high frequency modes.

In this chapter, temperature specific models are created based on the experimental FRD which was collected at six different temperature points. The temperature specific models are created to specifically reduce the uncertainty at the frequency from 350 to 600, where the first several high frequency modes occur. Then the temperature dependent model is designed based off these temperature specific models.

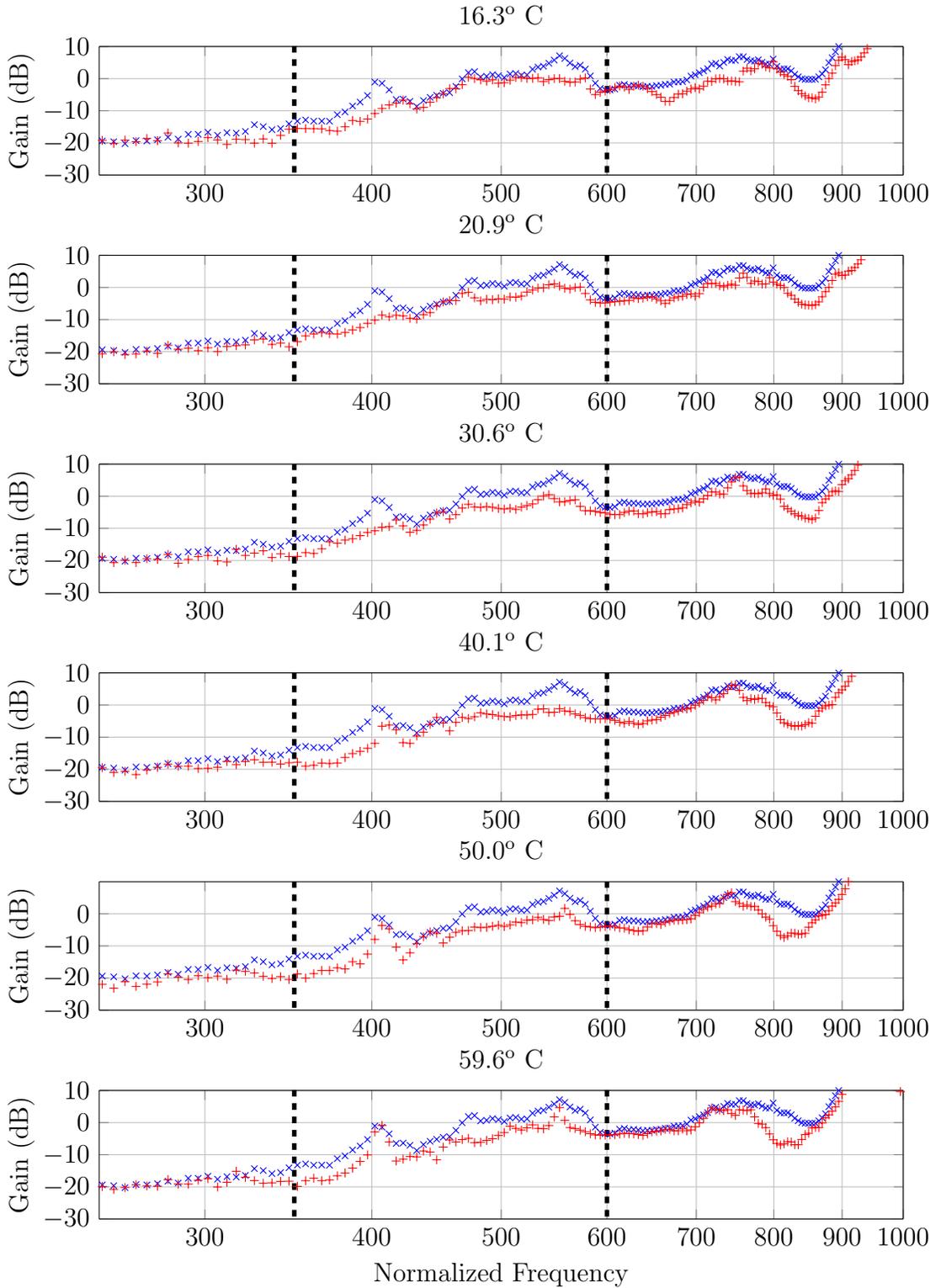


Figure 5.3: The optimal uncertainty weight for optimal MA FRD, for temperature independent system, $M_{T_{ind}}(j\omega_f)$, \times and temperature dependent system, $M_{T_{dep}}(T, j\omega_f)$, $+$. The range from frequency of 350 to 600 is where the first several high frequency modes occur, \blacksquare .

5.3.2 Method

An assumption was made such that the natural frequency and the damping ratio of the high frequency modes changes linearly with temperature. A temperature specific model, $G_{T_i}(s)$, is designed at each temperature point. Then linear regression is used to design the linear dependent state-space matrices using the state-space matrices from $G_{T_i}(s)$. As shown in Equation 5.3, the linear dependent state-space matrices is used to create the temperature dependent model, $G_{T_{dep}}(s, T)$.

For this method to be valid, each temperature specific model, $G_{T_i}(s)$, must follow certain restrictions. The state-order of each $G_{T_i}(s)$ must be kept the same, and every state of $G_{T_i}(s)$ must be kept consistent across all temperature points. In other words, the states for each state-space realization of $G_{T_i}(s)$ must hold the same physical interpretation for all temperature points. If the states are not consistent between the temperature specific models, the state-space matrices will not be linearly interpolatable. The method described in this chapter identifies the high frequency modes that exists at each temperature point with a slightly different natural frequency and damping ratio. Second order state-space fits are used to model these high frequency modes, such that $G_{T_i}(s)$ can be constructed as a series of second order state-space systems to maintain consistent states.

For each, l^{th} temperature point a nominal model, $G_{T_l}(s)$ is designed by multiplying a series of second order systems, $L(s)$, and $H_n(s)$, ($n = 1, \dots, N$) with a specific state-space realization. A second order system, $L(s)$, is the same at all temperature points, while $H_n(s, T)$ is unique at each temperature point for all n . For sake of simplicity although $H_n(s, T)$ is dependent on temperature it is not explicitly denoted in the following sections. For the VCM, $L(s)$ is a double integrator or low frequency pair of poles, and for the MA, $L(s)$ is a high frequency pair of poles. A second order state-space system, $H_n(s)$, is created by fitting a second order state-space system to the d^{th} component of the optimal FRD, $M_{T_l}(j\omega_f)$. The $H_n(s)$ is then converted into a second order transfer function of the form,

$$H_n(s) = \frac{a_{2,n}s^2 + a_{1,n}s + a_{0,n}}{s^2 + b_{1,n}s + b_{0,n}} \quad (5.5)$$

then a controllable canonical realization of this transfer function is defined as:

$$H_n(s) = \left[\begin{array}{c|c} A_{H_n} & B_{H_n} \\ \hline C_{H_n} & D_{H_n} \end{array} \right] = \left[\begin{array}{cc|c} 0 & 1 & 0 \\ -b_{0,n} & -b_{1,n} & 1 \\ \hline a_{0,n} - a_{2,n}b_{0,n} & a_{1,n} - a_{2,n}b_{1,n} & a_{2,n} \end{array} \right] \quad (5.6)$$

The second order roll off, $L(s)$, is defined such that,

$$L(s) = \frac{a_{0,L}}{s^2 + b_{1,L}s + b_{0,L}} \quad (5.7)$$

and the same state-space realization that is used for $H_n(s)$ is used for $L(s)$. The step by step method of designing $H_n(s)$, then $G_{T_l}(s)$, and ultimately the temperature dependent model $G_{T_{dep}}(s, T)$ is described next.

The following steps are taken to construct the temperature dependent model.

1. The optimal nominal FRD, $M_{T_l}(j\omega_f)$ is derived for each temperature point using the convex optimization method described in the previous chapter.
2. Divide out $L(s)$ from each l^{th} $M_{T_l}(j\omega_f)$. The state-space system $L(s)$ is assumed to be part of the system at all temperature points. For the VCM system $L(s)$ is a pair of low frequency poles and for the MA system $L(s)$ is a pair of high frequency poles.
3. For the d^{th} second order fit isolate the frequency of $M_{T_l}(j\omega_f)/L(j\omega_f)$ from $f = [f_n, \bar{f}_n]$ corresponding to a mode of interest.
4. Fit a second order system, $H_d(s)$, to $\{M_{T_l}(j\omega_f)/L(j\omega_f)\}_{f=f_n}^{\bar{f}_n}$ using a function such as `fitfrd` on MATLAB.
5. Divide out the second order fit, $H_d(s)$, from the optimal FRD, $M_{T_l}(j\omega_f)/L(j\omega_f)$.
6. Repeat step 3-5 until the desired number, N , of second order system is designed.
7. Define, $G_{T_l}(s)$ such that $G_{T_l}(s) := L(s) \prod_{d=0}^D H_n(s)$ where the product of the state-space system is done using the same state-space realization for each $H_n(s)$ and $L(s)$.
8. Find a linear temperature dependent fit for each state-space matrices using A_l , B_l , C_l , and D_l to design $A(T)$, $B(T)$, $C(T)$, and $D(T)$.

9. Construct state-space system $G_{T_{dep}}(s, T)$ from $A(T)$, $B(T)$, $C(T)$, and $D(T)$.

These steps are illustrated by a series of figures using the experimental FRD data for the MA. Figure 5.4 shows the optimal FRD, $M_{T_3}(j\omega_f)$ of the MA data at 30.6°C and the experimental FRD, $\{D_k(j\omega_f, T_3)\}_{k=1}^K$.

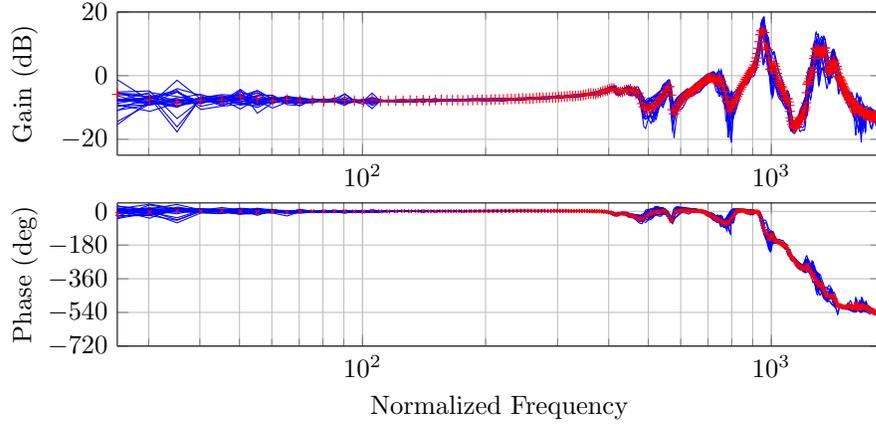


Figure 5.4: The FRD of HDD data at 30.6°C, —, and the optimal FRD at 30.6°C, +.

First the high frequency roll off, $L(j\omega_f)$, is divided out from $M_{T_3}(j\omega_f)$. Then $M_{T_3}(j\omega_f)/L(j\omega_f)$ is isolated from the normalized frequency of $\omega_{\underline{f}_1} = 60$ to $\omega_{\bar{f}_1} = 427$ as shown in Figure 5.5.

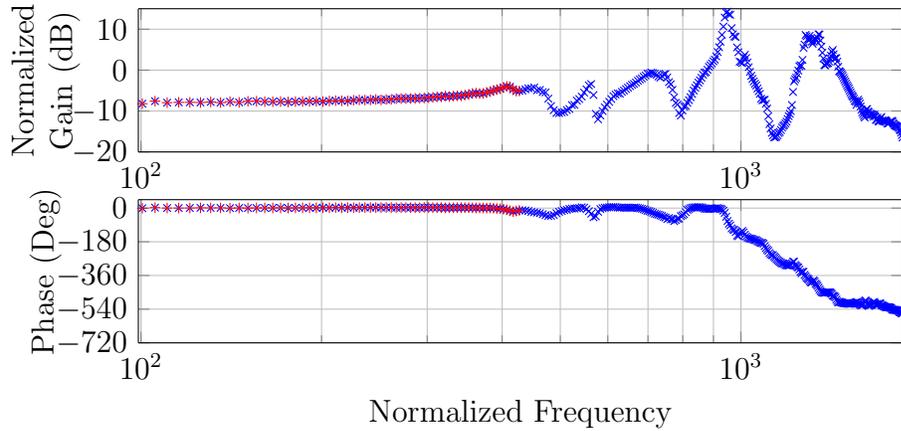


Figure 5.5: The optimal FRD, $\{M_{T_3}(j\omega_f)/L(j\omega_f)\}_{\bar{f}_1}^F$, x, and the same FRD isolated from the normalized frequency of $\omega_{\underline{f}_1} = 60$ to $\omega_{\bar{f}_1} = 427$, +.

Then a second order state-space system, $H_1(s)$ is fit onto the isolated FRD using `fitfrd` from MATLAB as shown in Figure 5.6.

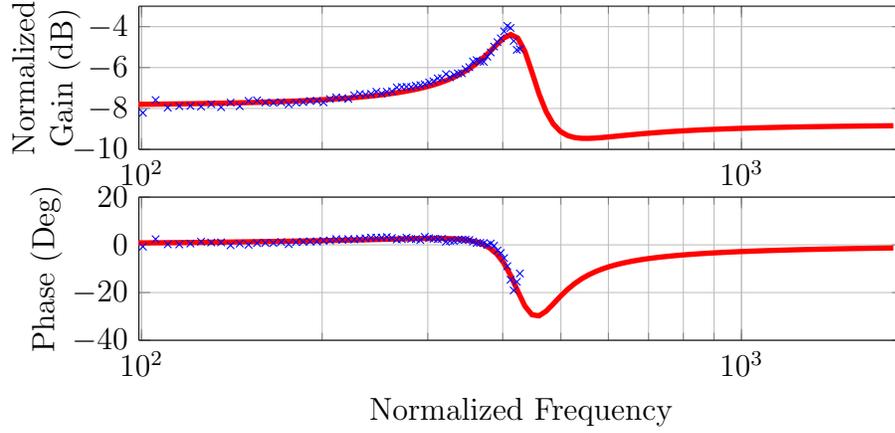


Figure 5.6: The $M_{T_3}(j\omega_f)/L(j\omega_f)$, \times , and the second order fit $H_1(s)$, --- .

$H_1(s)$ is divided out from $M_{T_3}(j\omega_f)/L(j\omega_f)$, and the next mode of interest from $\omega_{\underline{f}_2} = 60$ to $\omega_{\bar{f}_2} = 497$ is isolated as shown in Figure 5.7. The first two modes have very similar natural frequency, and thus $\omega_{\bar{f}_2}$ is similar to $\omega_{\bar{f}_1}$.

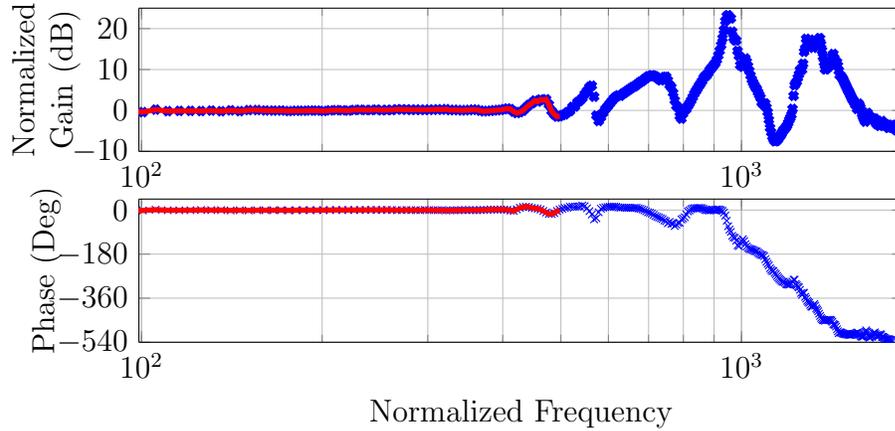


Figure 5.7: The $\{M_{T_3}(j\omega_f)/(L(j\omega_f)H_1(j\omega_f))\}_{f=1}^F$, \times , and the same FRD with the second mode of interest isolated from $\omega_{\underline{f}_2} = 60$ to $\omega_{\bar{f}_2} = 497$, --- .

These steps are done up to $D = 4$ to form a tenth order state-space system. The normalized frequency that was isolated for each second order fit at $T_3 = 30.6^\circ\text{C}$ and the corresponding second order state-space fit is shown in Table 5.2.

The $G_{T_3}(s)$ is then constructed by multiplying a series of second order systems H_n

d	ω_{f_a}	ω_{f_b}	H_n
0	25	1859	$\frac{1.93e7}{s^2 + 8800s + 1.93e7}$
1	60	427	$\frac{1.45s^2 + 172.5s + 2.98e5}{s^2 + 81.08s + 1.84e5}$
2	60	497	$\frac{0.99s^2 + 13.66s + 2.18e5}{s^2 + 9.33s + 2.18e5}$
3	492	583	$\frac{0.87s^2 + 28.24s + 2.77e5}{s^2 + 13.39s + 3.13e5}$
4	60	804	$\frac{1.04s^2 + 119.4s + 5.77e5}{s^2 + 40.61s + 5.19e5}$

Table 5.2: The isolated frequencies and the second order system for each H_n .

such that,

$$G_{T_3} = L(s) \prod_{d=0}^4 H_n \quad (5.8)$$

where each product has the state-space realization,

$$H_1(s)H_2(s) = \left[\begin{array}{cc|c} A_{H_1} & B_{H_1}C_{H_2} & B_{H_1}D_{H_2} \\ 0 & A_{H_2} & B_{H_2} \\ \hline C_{H_1} & D_{H_1}C_{H_2} & D_{H_1}D_{H_2} \end{array} \right] \quad (5.9)$$

The same high frequency modes are modeled using second order systems for all temperature points. By multiplying the second order systems using the same state-space realization, the states can be kept consistent between each temperature points.

Figure 5.8 shows the resulting model for the MA at $T_3 = 30.6^\circ\text{C}$. The last two major peaks that were not modeled will be characterized by the uncertainty weight in the next chapter. By closely modeling the optimal nominal FRD, the required uncertainty weight gain will be minimized at each temperature point.

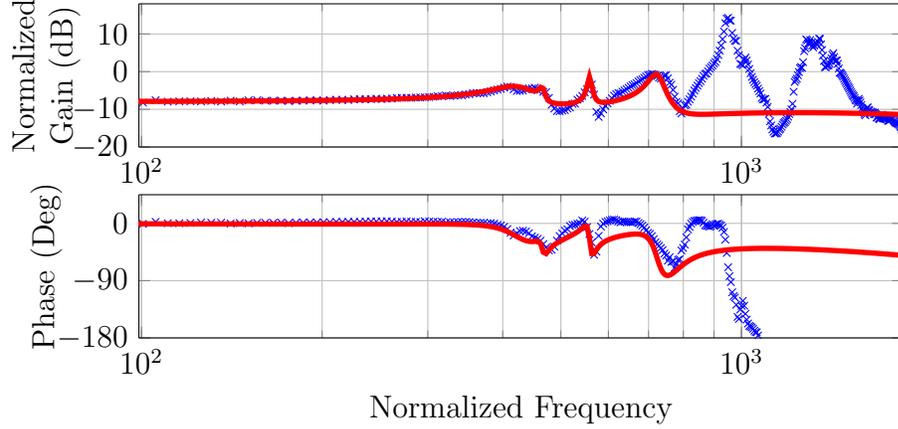


Figure 5.8: The optimal FRD, $M_{T_3}(j\omega_f)$, \times , and the state-space system $G_{T_3}(s)$ constructed from $H_n(s)$, $—$.

5.3.3 Constructing Temperature Dependent Models

Once $G_{T_l}(s)$ is designed at each temperature point a linear fit is done on each state-space matrices to create the temperature dependent system of the HDD. An example of a linear fit is shown in Figure 5.9 for the (1,3) entry of the A matrix. Since the state-space matrices are not perfectly linear, each state-space matrix entry does not lie on the linear fit. Using these linear fits, a temperature dependent model $G_{T_{dep}}(s, T)$

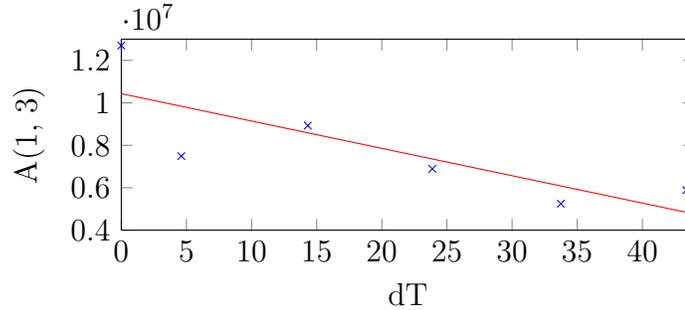


Figure 5.9: The linear fit, $—$, of the (1,3) entry of the A matrix based on the state space system designed at each temperature point, \times .

is constructed as,

$$G_{T_{dep}} = \left[\begin{array}{c|c} A_0 + (T - T_0)A_1 & B_0 + (T - T_0)B_1 \\ \hline C_0 + (T - T_0)C_1 & D_0 + (T - T_0)D_1 \end{array} \right] \quad (5.10)$$

The temperature dependent MA model, $G_{T_{dep}}(s, T_3)$, is compared to $G_{T_3}(s)$ in Figure 5.10. As seen in the linear fit, the state-space matrices are not perfectly linear, and thus there are some differences between $G_{T_{dep}}(s, T_3)$ and $G_{T_3}(s)$. The differences

are not significant and the optimal uncertainty weight required for the two models are similar.

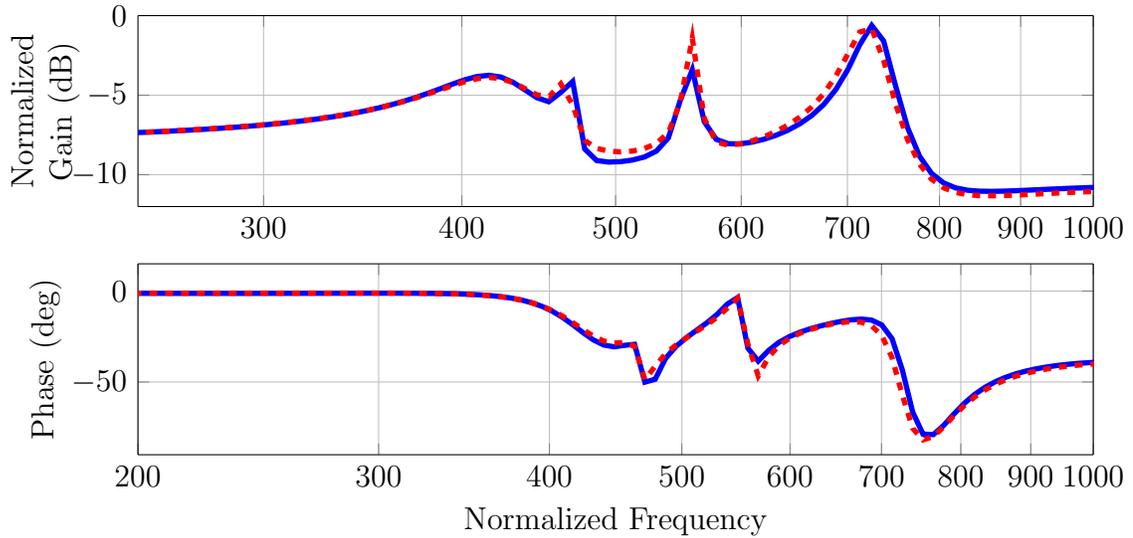


Figure 5.10: Temperature dependent model at 30.6°C, $G_{T_{dep}}(30.6, s)$, —, and the model designed at 30.6°C, $G_{T_3}(s)$, - - - .

5.3.4 Comparison with Temperature Independent Model

In Chapter 6, the temperature dependent model designed in this chapter is used to design a temperature dependent controller. As shown in the beginning of the chapter and Figure 5.12, it is possible to design a model that requires an additive uncertainty with reduced gain by limiting the scope of the experimental data set to specific temperatures. Using the temperature dependent nominal model, the bandwidth of the closed loop system can be increased for these reasons. The temperature dependent nominal model reduces the uncertainty at the normalized frequency from 350 to 600 compared to the temperature independent nominal model. It also characterizes the high frequency modes of the actuator dynamics more accurately than the temperature independent nominal models.

Figure 5.11 shows the tenth order temperature dependent and independent nominal models for the MA at six different temperature points. As it can be seen, the natural frequency and damping ratio of the temperature dependent models change as the temperature changes. This allows the uncertainty of the system to decrease.

The minimum uncertainty weight gain required to cover the experimental FRD for

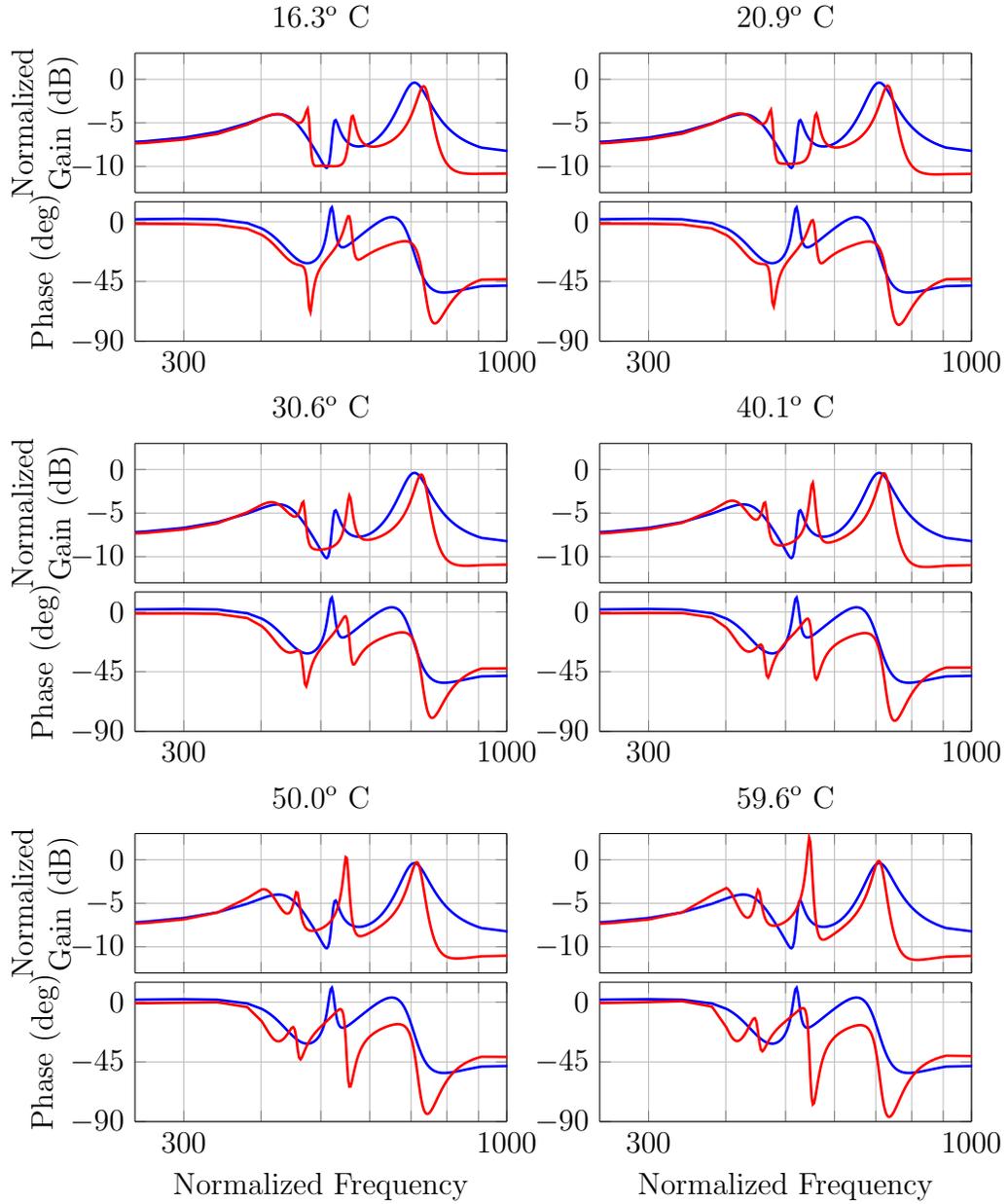


Figure 5.11: The temperature independent, —, and dependent, —, model of the MA used for the control design.

the temperature independent and dependent models at six different temperatures are shown in Figure 5.12. The temperature independent model from Chapter 3 was used for this comparison.

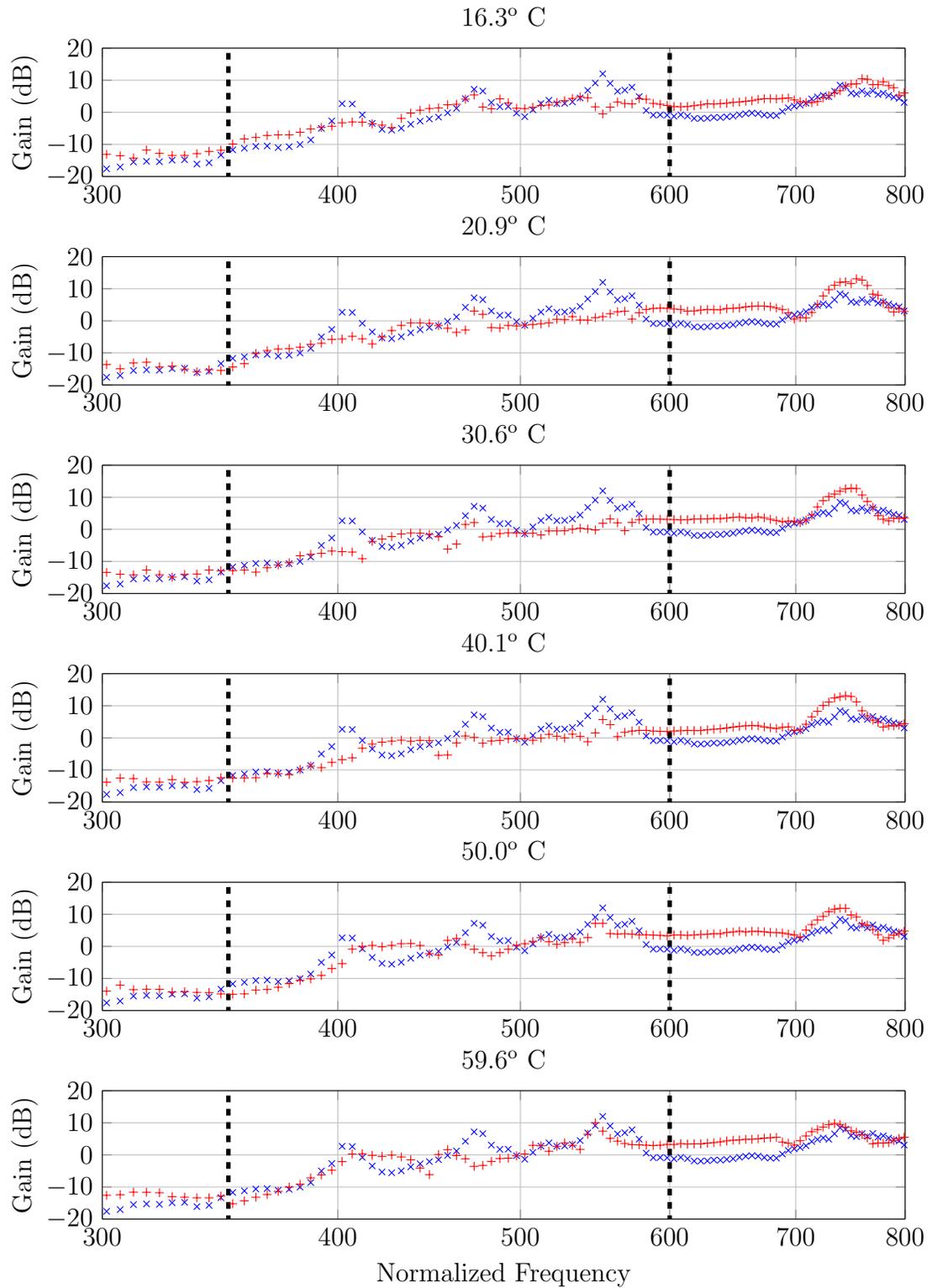


Figure 5.12: The optimal uncertainty weight for temperature independent MA model, $G_{T_{ind}}(s)$, \times , and temperature dependent MA model, $G_{T_{dep}}(s, T)$, $+$. The range from frequency of 350 to 600 is where the first several high frequency modes occur, --- .

From Figure 5.12, there are various observations that can be made between the temperature dependent and independent nominal models. The bode plot of the optimal uncertainty weight shows that there are frequencies at which the temperature independent model seems to have lower uncertainty compared to the temperature dependent model. Therefore the area underneath the optimal weight, as shown in Table 5.3, was used as a metric to show the reduction of overall uncertainty from normalized frequency of 350 to 600. This metric was chosen because it represents the overall reduction in uncertainty within the frequencies of interest.

Overall the temperature dependent model was able to reduce the overall uncertainty from the normalized frequency of 350 to 600. The uncertainty was reduced the least at 16.3°C with the most reduction occurring at 30.6°C. Furthermore, the temperature dependent models better characterize the frequency shifting of the high frequency modes compared to the temperature independent model. Although the reduction in the uncertainty of the system using a temperature dependent model seems small, the modern HDD system already pushes the limits of performance. Thus any improvements to the HDD controller system is beneficial for the system.

T	16.3°C	20.9°C	30.6°C	40.1°C	50.0°C	59.6°C	T_{ind}
Area	277	225	214	223	263	271	308

Table 5.3: The area under the optimal weights from normalized frequency of 350 to 600 for temperature specific models and temperature independent model.

For frequency below 350, the temperature dependent model has greater uncertainty compared to the temperature independent system, however the optimal uncertainty weight gain is less than -10dB below frequency of 350. It was shown in Chapter 3 that the actuator weight for the MA is designed such that the control effort for the MA is reduced at low frequency and increased at high frequency. Therefore this difference in uncertainty at low frequency does not significantly affect the control design.

The uncertainty for both the models continue to increase past the normalized frequency of 600. The controller has to roll off past normalized frequency of 600 to robustly stabilize the closed loop system. Thus at high frequency, the difference in the uncertainty past this frequency does not significantly affect the performance of the controller. By lowering the required uncertainty weight near the first several high frequency mode of the MA system, it will be shown that a higher bandwidth

controller can be designed in the Chapter 6.

5.4 Conclusion

In this chapter a method to construct a temperature dependent model of the HDD system was shown under the assumption that the temperature was a slow varying parameter. Actuator nominal models with consistent states were designed at each temperature point to design a temperature dependent nominal model. By designing a temperature dependent nominal model, the optimal uncertainty weight at the first several modes was reduced. The method described in this chapter is also useful for designing a slow varying parameter dependent models for other systems based on their experimental FRD. The temperature dependent nominal model that was designed in this chapter will be used to design a temperature dependent controller in Chapter 6.

Chapter 6

Temperature Dependent Robust Controller

6.1 Introduction

The controller designed for hard disk drives (HDDs) must be robust to different HDDs of the same product, and varying temperature. As described in Chapter 5, changes in temperature affect the elastic property of the actuator, and alter the natural frequency and damping ratio of the high frequency modes. In Chapter 3, a linear time invariant (LTI) robust controller was designed using a standard D-K synthesis method. Using the experimental frequency response data (FRD), that was provided by Seagate, an LTI uncertainty model of the system was designed for the robust controller design. The experimental FRD was measured from sixteen different HDDs at six different temperature points. In this chapter, using the same set of experimental data, a temperature dependent controller is designed and compared to the LTI controller designed from Chapter 3. The goal of the temperature dependent controller was to increase the bandwidth of the closed-loop sensitivity while maintaining its low frequency and peak gains.

There are existing techniques for designing and implementing linear parameter varying (LPV) controllers. One such type of LPV controller is the gain-scheduled controller. In general, a gain-scheduled controller is implemented by varying the controller coefficients based on the current value of the measured parameter. These controller coefficients could be a coefficient of a transfer function, or entire state-

space matrices of the gain-scheduled controller [67–69]. Gain scheduled controllers have been implemented for autopilot systems for missiles [70], VSTOL aircraft [71], and automobile engines [72] in the past. The advantage of gain scheduling is that the controller can be designed using existing linear control design methods and extended to design a LPV controller. There are various methods of designing a gain-scheduled controller, and therefore the proper method must be developed and used based on the constraints and assumptions of the given problem.

As stated in Chapter 5, it is assumed that the temperature of the HDD system is a slow varying parameter, therefore the rate of change of the temperature is negligible. There are several constraints present for designing a controller for HDD systems. There is a limited data storage available to store the state-space system of the controller. Therefore the controllers can not take up arbitrary memory, which means that a large number of state-space matrices can not be stored. Furthermore HDDs have a short product cycle of two to three years and a significant amount of time can not be spent tuning robust controllers. The interconnected model required to design a high performance controller reaches a state order of over 50 for the HDD system, and thus a computationally efficient method must be chosen to design its controller. Any methods that would require days of computation to design a controller would not be applicable for HDD systems. Finally, the method needs to be able to design a robust controller by taking into account the uncertainty of the system. In summary, the temperature dependent controller for the HDD system must not require a large storage space, require small computation time to design, and be robust to uncertainties.

There are various methods of designing a gain-scheduled controller. One method requires solving a set of linear matrix inequalities to find an optimal LPV controller for a given system [68, 73–76]. These LPV controllers take into account the rate of change of the parameters as well, which the HDD controller does not require. The size of the linear matrix inequalities scale with the state-order of the interconnected model. Thus for a HDD system with more than 50 state orders, currently, it would take a significant amount of computation time to solve these optimization problems. Another design method is to linearize the pole-zero mapping of the controllers [77–80]. These interpolation methods, however require the user to manually align the poles and zeros of the controller, which again is inefficient for a system with more than 50 states. There are also ad-hoc methods that have been used in the past, such as interpolating the solutions of the Riccati equations from H_∞ synthesis to design

a linearly interpolatable controller [70]. Another method linearly interpolates the state-space matrices of a balanced controller realization to design a gain-scheduled controller [71].

Due to the assumptions and constraints of the HDD problem, an ad-hoc approach has been chosen and developed to design the temperature dependent controller. By using a D-K synthesis like method, as shown in Chapter 3, a robust controller that is linearly interpolatable as a function of temperature was designed using the temperature dependent nominal models that were constructed in Chapter 5. As H_∞ synthesis is the foundation for constructing the controller, a controller can be designed with relative quickness by solving an Algebraic Riccati Equations compared to methods based on solving a set of linear matrix inequalities. The temperature dependent controller was validated through the use of experimental FRDs and by implementing the controller for a fixed temperature point into a real HDD. After the results section, the H_∞ synthesis method is closely examined to describe the conditions that are required for the controllers to be linearly interpolatable. Most of the information of H_∞ synthesis is derived from Zhou’s textbook to validate the interpolation method [31].

6.2 Problem Formulation

The problem formulation assumes that a temperature dependent model, $G_{T_{dep}}(s, T) \in \mathbb{RH}_\infty^{(2 \times 1)}$, was designed from a collection of $n \times m$ frequency responses measured at the l^{th} temperature point T_l ($l = 1, \dots, L$) derived from input/output experiments. The k^{th} experimental dataset ($k = 1, \dots, K$) consists of the complex FRD, $D_k(T_l) := \{D_k(j\omega_1, T_l), \dots, D_k(j\omega_F, T_l) \subset \mathbb{C}^{2 \times 1}\}$, defined on a common grid of frequencies, $\{\omega_f\}_{f=1,2,\dots,F} \subset \mathbb{R}$. This data can be easily and efficiently computed for many drives in the HDD application using a basic sinusoidal frequency sweep inside a temperature chamber. For the results shown later, there were sixteen experimental dataset for six different temperature points for a total of ninety-six datasets.

Consider the robust synthesis problem for an uncertain temperature dependent system, $P(s, T)$ as shown in Figure 6.1. It will be shown later that $P(s, T)$ is an interconnected model created using the temperature dependent nominal models, $G_{T_{dep}}(s, T)$, uncertainty, actuator, and performance weights. The uncertain temperature dependent interconnected model is described by an open-loop temperature dependent system, $P(s, T)$, and an uncertain LTI dynamic, Δ . The objective is to synthesize a

temperature dependent controller, $K(s, T)$, to minimize the robust performance of the closed-loop system at each T_l :

$$\inf_{K(s, T_l) \text{ stabilizing}} \|F_L(P(s, T_l), K(s, T_l))\|_\infty \quad (6.1)$$

For the HDD controller problem, it will be shown that by minimizing the gain of $F_L(P(s, T_l), K(s, T_l))$ for each l , the temperature dependent controller will provide similar robust performance for all T from T_1 to T_L .

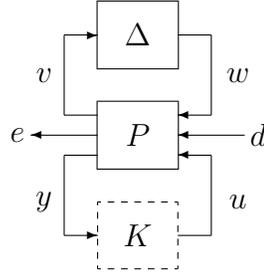


Figure 6.1: Uncertain temperature dependent robust synthesis problem.

6.3 Temperature Dependent Robust Controller

Experimental FRD of sixteen different HDDs were measured at six different temperature points. Table 6.1 shows the temperature points at which the experimental FRDs were collected. Based on these experimental FRDs, the temperature dependent nominal models were designed using the method described in Chapter 5. With the temperature dependent nominal models and the experimental FRDs, the uncertainty weights for the system were created. Furthermore the same experimental FRDs were used to simulate the open-loop and closed-loop sensitivity of the HDD system.

l	1	2	3	4	5	6
Chamber Temperature ($^{\circ}\text{C}$)	5	15	25	35	45	55
Internal Temperature ($^{\circ}\text{C}$)	16.3	20.9	30.6	40.1	50.0	59.6

Table 6.1: The temperature of the HDD for index l .

6.3.1 Method

The fully interconnected model design for the HDD system is shown in Figure 6.2. From the figure, $G_V(s, T)$ and $G_M(s, T)$ are the temperature dependent nominal

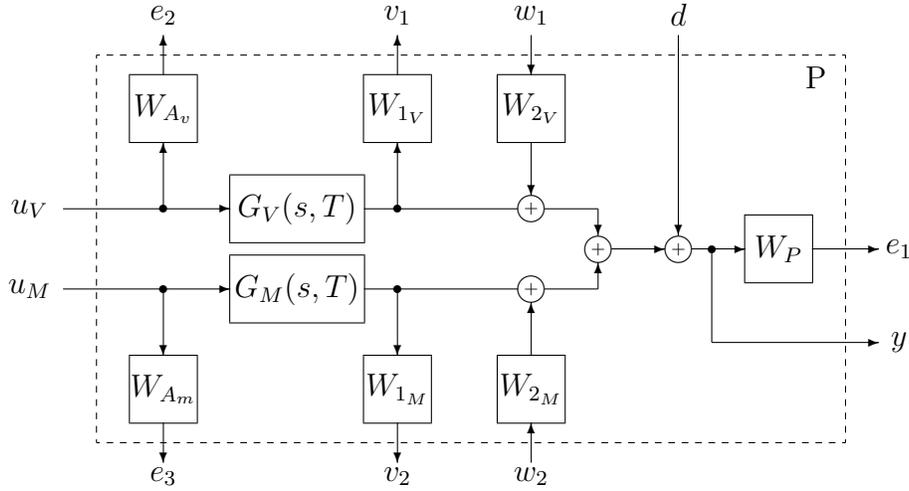


Figure 6.2: The interconnected model used to design a robust controller, $K(s)$

models for the VCM and MA, respectively. W_{A_v} and W_{A_m} are the actuator weights for the VCM and MA, respectively. The W_{1V} and W_{1M} are the output multiplicative uncertainty weight for the VCM and MA, respectively. W_{2V} and W_{2M} are the input multiplicative uncertainty weights for the VCM and MA, respectively. The methods used to design the uncertainty, actuator, and performance weights were the same as the method described in Chapter 3 and the relevant MATLAB functions are listed in Appendix A.

Once the temperature dependent interconnected model is designed, the D-K iteration can be used at a single temperature point, T_l , to derive a D-scale for all temperature points. For this case, the highest available temperature point was used to derive the D-scale. All of the weights and the D-scale are designed as LTI systems, and therefore the only temperature dependent components are the nominal models. Once the D-scale is derived, H_∞ synthesis is used to construct a robust controller at each of the experimental temperature points. The temperature dependent controller is implemented by linearly interpolating the state-space matrices of the controller as a function of temperature. In this chapter, the lowest and highest experimental temperature points (16.3°C and 59.6°C) are used as references for the gain-scheduling. While the controllers designed at four other temperature points (20.9°C, 30.6°C, 40.1°C and 50.0°C) are used to validate the gain-scheduled controller.

The following steps are taken to design a temperature dependent controller,

1. Design temperature dependent nominal models $G_V(s, T)$ and $G_M(s, T)$ for the

VCM and MA.

2. Design temperature independent uncertainty weights, W_{1V} , W_{2V} , W_{1M} and W_{2M} to characterize the uncertainty of the system.
3. Design temperature independent actuator weights, $W_{Av}(s)$ and $W_{Am}(s)$ to define the desired actuator effort for the VCM and MA at different frequencies.
4. Design the performance weight, $W_p(s)$ to define the desired loop shape of the closed-loop sensitivity.
5. Convert continuous models and weights into discrete systems.
6. Construct a discrete temperature dependent interconnected model, $P(z, T)$ using the temperature dependent nominal models, uncertainty weights, performance weights, and actuator weights.
7. Run D-K synthesis at, T_l for some l , to derive a LTI D-scale, $D_{scale}(z)$, that will be used to design a robust temperature dependent controller.
8. Run H_∞ synthesis on $D_{scale}(z)P(z, T)D_{scale}(z)^{-1}$ at $T = T_l$ from $(l = 1, \dots, L)$ to design temperature specific controllers $K_{T_l}(z)$ at all l .
9. The state-space matrices of temperature dependent robust controller, $K_{dep}(z, T)$ are defined as a linear interpolation of the state-space matrices of $K_{T_1}(z)$ and $K_{T_L}(z)$ as a function of T .
10. Validate the controller design by comparing the linearly interpolated controller $K_{dep}(z, T)$ against the controller, $K_{T_1}(z)$.
11. Further validate the controller by implementing the controller $K_{dep}(z, T)$ into a HDD for some T .

The temperature dependent controller is defined as,

$$\begin{aligned} x_k[n+1] &= A_k(T)x_k[n] + B_k(T)y[n] \\ u[n] &= C_k(T)x_k[n] + D_k(T)y[n] \end{aligned} \tag{6.2}$$

where T is the temperature and the state-space matrices are linearly dependent on T such that

$$A_k(T) = A_{k_1} + (T - T_1)A_{k_L}, \tag{6.3}$$

where T_1 is a reference temperature point, corresponding to l in Table 6.1, and A_{k_1} and A_{k_L} are the state matrices from the controller designed at T_1 and T_L , respectively. Since it is assumed that the temperature is a slow varying parameter, the temperature rate of change is neglected by this controller. The described method is not valid for most parameter dependent system as the controllers designed from H_∞ synthesis are typically not linearly interpolatable. For a special case, however, the controllers designed from H_∞ synthesis are linearly interpolatable, and these conditions will be described in the later section.

The gain-scheduled controller is validated in several ways. First, the control effort of the gain-scheduled controller, $K(z, T)$ is compared to the temperature specific controllers, $K_{T_l}(z)$. Second, the temperature dependent controller and the experimental FRD were used to simulate the open-loop and closed-loop sensitivity of the system. Lastly, the temperature dependent controller was implemented on a HDD to validate its performance on a real system.

6.4 Results

In this section, the results from each step of the construction of the temperature dependent controller are shown.

6.4.1 Temperature Dependent Nominal Models

The temperature dependent nominal models were designed using the method described in Chapter 5. Figure 6.3 and Figure 6.4 show the Bode plots of the temperature independent and dependent VCM and MA models, respectively, at each T_l . The LTI and temperature dependent VCM models are both sixth order systems. The MA models are tenth order systems for both the LTI and temperature dependent nominal models. It can be seen that the frequency response dynamics of the temperature dependent model changes as the temperature changes, while the temperature independent model does not. The temperature dependent model was designed to minimize the uncertainty of the system at each temperature point while the temperature independent model was designed to minimize the uncertainty of the system for all temperature point.

Figure 6.5 and Figure 6.6 compare the high frequency modes for the VCM and MA, respectively, at 16.3°C and 59.6°C. The end points of the experimental temperature

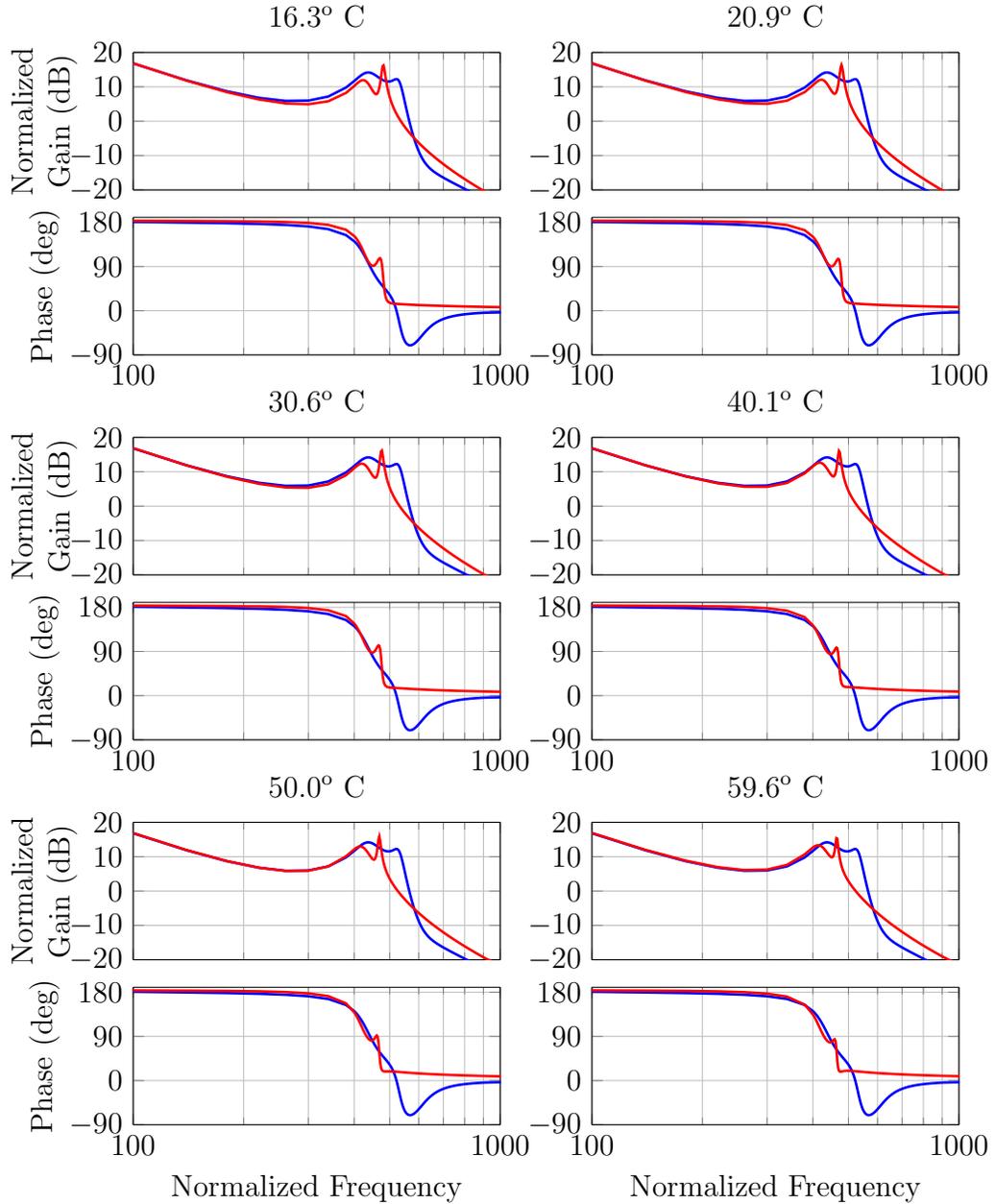


Figure 6.3: The temperature independent, —, and dependent, —, nominal model of the VCM used for the temperature dependent and independent controller design.

points were chosen to illustrate the difference in the frequency response of both the models as temperature changes. It can be seen from Figure 6.5 that the natural frequency of the high frequency modes decreases as the temperature increases. The changes in the damping ratio can be seen from the high frequency mode that occurs near the normalized frequency of around 500 in Figure 6.6. The peak of the mode at 16.3°C has a gain of around -5dB, while the gain of the peak at 59.6° is about

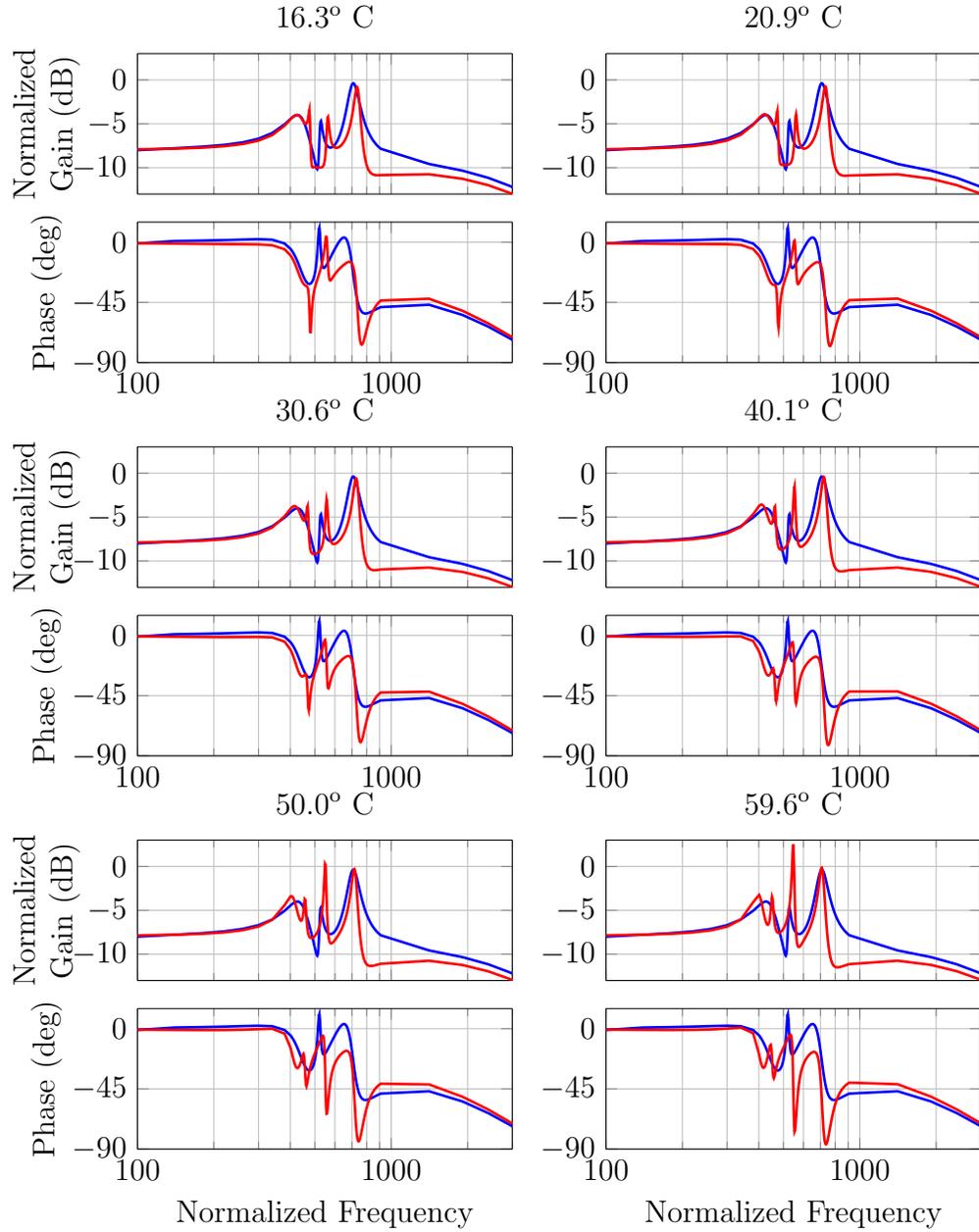


Figure 6.4: The temperature independent, —, and dependent, —, nominal model of the MA used for the temperature dependent and independent controller design.

3dB. Both the natural frequency and damping ratios change as the temperature of the system changes.

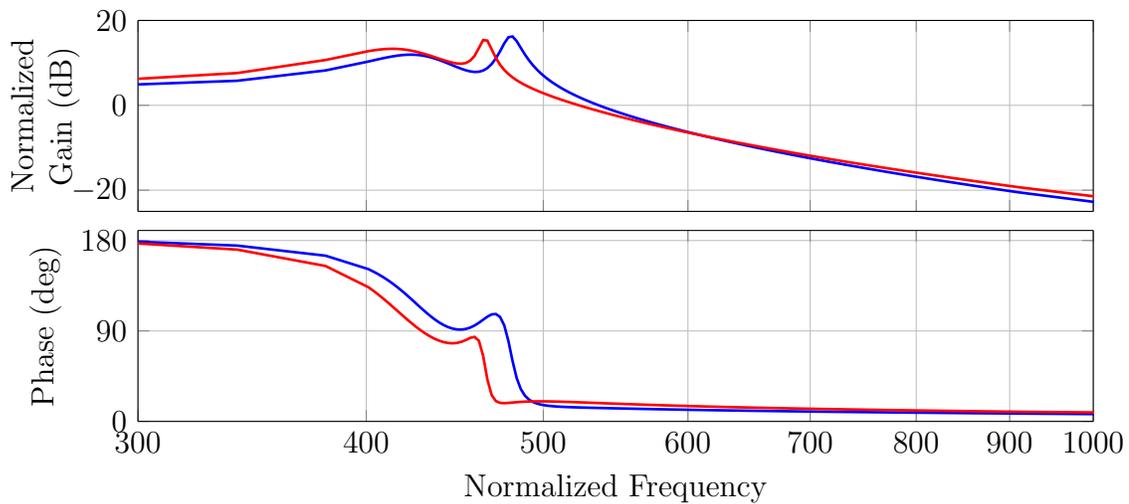


Figure 6.5: The temperature dependent model of the VCM at 16.3° — and at 59.6°C —.

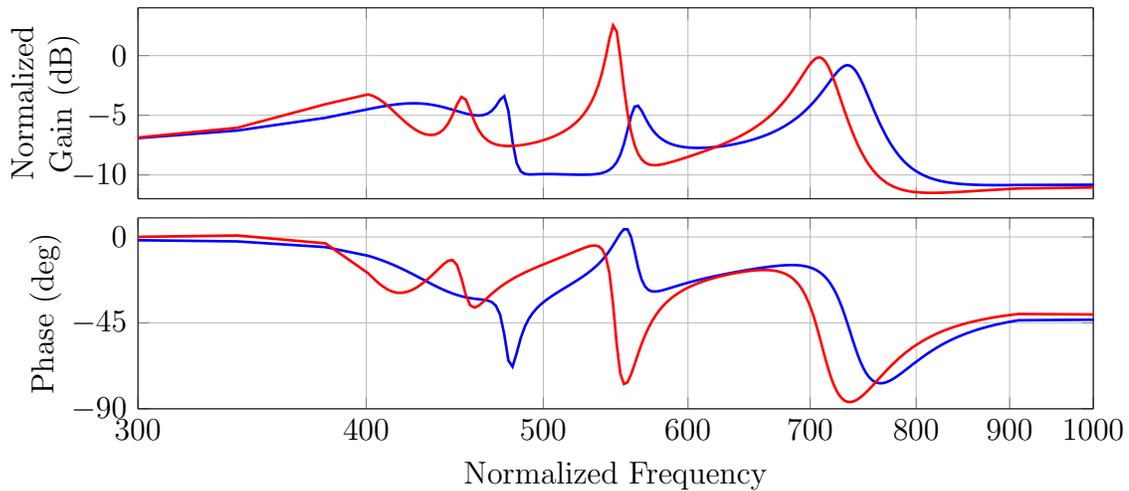


Figure 6.6: The temperature dependent model of the MA at 16.3° — and at 59.6°C —.

6.4.2 Uncertainty Weights

Once the temperature dependent nominal models are designed, the uncertainty weights are designed to create temperature dependent uncertainty models of the HDD. The purpose of the uncertainty weight with the experimental FRD is to capture the largest difference between the nominal model, and the set of experimental data from which the model was designed from. The uncertainty weights were modeled as LTI systems to keep the states of the interconnected model consistent across temperature.

The multiplicative uncertainty set, \mathcal{S}_M , for a SISO system is defined as [31],

$$\mathcal{S}_M := \{(1 + W_L \Delta W_R)G_0, \Delta \in \mathbb{RL}_\infty, |\Delta|_\infty \leq 1\} \quad (6.4)$$

The goal is to design uncertainty weights $W_L(s)$ and $W_R(s)$ such that all $D_k(j\omega_i, T_l)$ exists within the set while minimizing the size of \mathcal{S}_M . The uncertainty weights for the VCM and MA are designed independently and thus the following steps are repeated for each component. Since the uncertainty weights are designed for SISO systems, the uncertainty weight can first be designed as a single state-space system where $W(s) = W_L(s)W_R(s)$. The initial step is to derive optimal uncertainty weights based on the temperature dependent model $G_{dep}(s, T_l)$ and $D_k(j\omega_f, T_l)$ for all l . A function called `ucover` on MATLAB can be used to derive optimal uncertainty weight, $W_{opt}(G_{dep}, D_k, j\omega_f, T_l)$, for the temperature dependent model at each T_l [7].

Then a non-minimum phase fitting method, such as `fitmagfrd` on MATLAB, is used to create a state-space model based on $W_{opt}(G_{dep}, D_k, \omega_f, T_l)$. For this chapter, an iterative fitting method using a manually designed frequency response data which over bounds $W_{opt}(G_{dep}, D_k, \omega_f, T_l)$ was used to create the uncertainty weights. This method is described in Chapter 3 and the MATLAB code for it is shown in Appendix A. Figure 6.7 show the uncertainty weights that were designed for the VCM and the optimal uncertainty weights derived from `ucover`. The same VCM uncertainty weight was used for both the LTI and temperature dependent model as the optimal uncertainty weight required for both models were similar.

Figure 6.8 shows the MA uncertainty weight that was designed, and the optimal uncertainty weights created by `ucover` from MATLAB. It can be seen that for a temperature dependent model, a smaller gain is required for the MA uncertainty weight from 200 to 500 normalized frequency. Another benefit, which will be demonstrated

later in the chapter, is that the temperature dependent nominal models characterize the high frequency modes of the actuator dynamics more accurately than the temperature independent nominal models. The temperature independent nominal models are essentially an average of the system across various temperature points and thus do not capture the transition of the high frequency modes as the temperature changes.

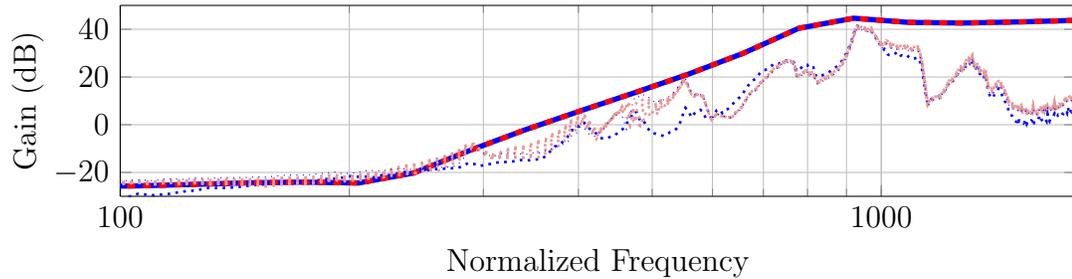


Figure 6.7: The VCM uncertainty weight for temperature independent system — and temperature dependent system - - -. The ···· and ···· are the optimal uncertainty weight for the given models.

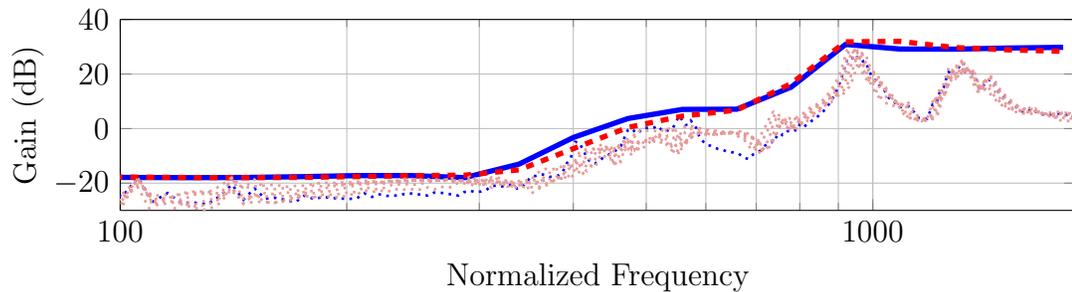


Figure 6.8: The MA uncertainty weight for temperature independent system — and temperature dependent system - - -. The ···· and ···· are the optimal uncertainty weight for the given models.

6.4.3 Actuator Weights

The actuator weights are used to define a relative control effort desired by the VCM and MA at different frequencies. Typically, it is desired for the MA to have higher controller effort at high frequency as the VCM has lower actuator bandwidth compared to MA [9]. The VCM actuator weight is a constant gain while the MA actuator weight is designed to be a low pass filter. Figure 6.9 shows the actuator weights for the VCM and MA that were used to design the temperature dependent interconnected model. It can be seen that the actuator effort for the MA quickly ramps up past the normalized frequency of 4 at about -50 dB/Dec. The large slope is used to rapidly increase the control effort from the MA as the frequency increases.

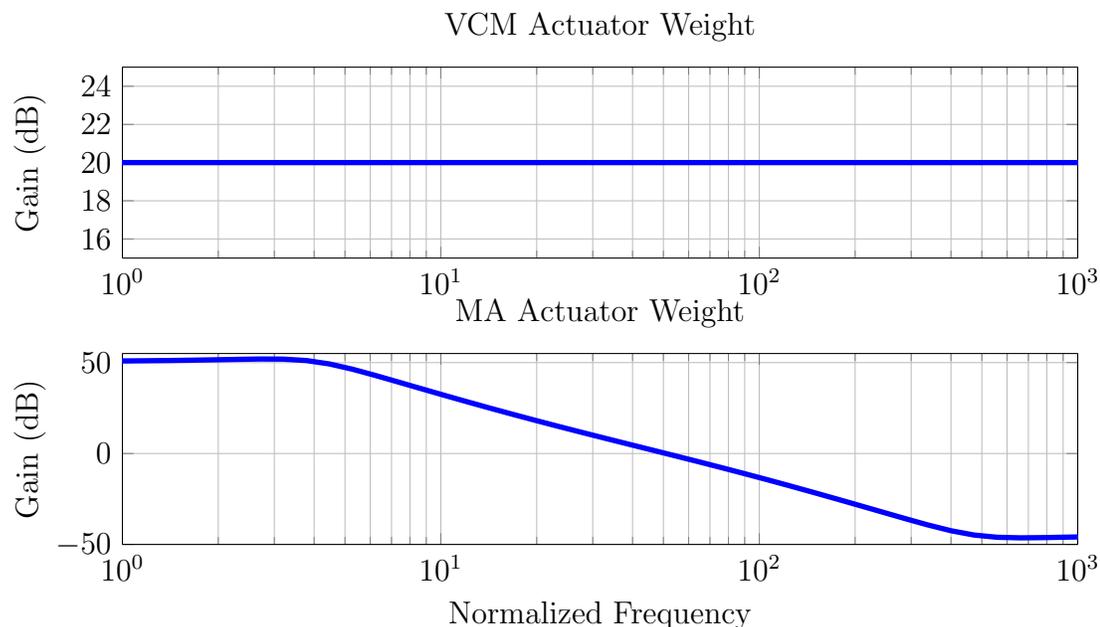


Figure 6.9: The actuator weights used for VCM and MA for temperature dependent and independent controller design.

6.4.4 Performance Weight

The same method described in Chapter 3 is used to design a performance weight. A frequency response data with the desired shape of the closed-loop sensitivity is manually designed. Then an iterative fitting process using a non-minimum phase fitting tool such as `fitmagfrd` is used to design a state-space system for the performance weight. Figure 6.10 shows the inverse of the performance weight designed for the LTI system and the temperature dependent system. The inverse represents the desired over bound on the closed-loop sensitivity of the system. The performance weights are chosen to be similar for both the systems so that they are comparable in low frequency performance and peak sensitivity gain.

6.4.5 Discretization

Before the interconnected model can be constructed using either `connect` or `sysic` function in MATLAB, all of the components must be discretized. The HDD system is inherently discrete as the magnetic head obtains positional data from the servo section in fixed time intervals. The temperature dependent models are discretized using zero order hold and the weights are discretized using the matched pole-zero method. The same zero order hold method is used for each temperature point, thus

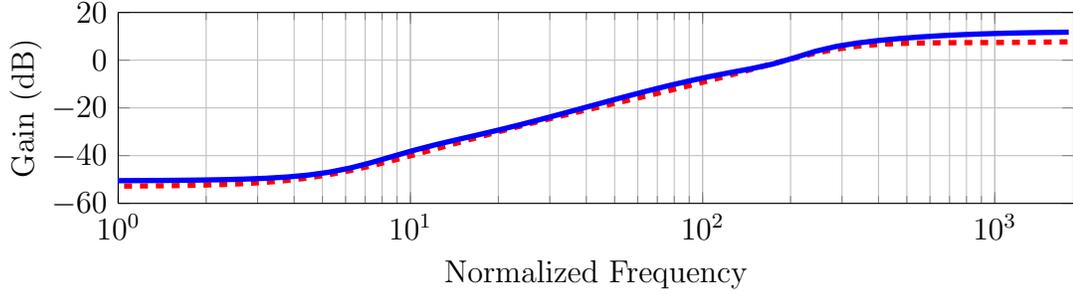


Figure 6.10: The inverse of the performance weight for temperature independent system — and temperature dependent system - - .

the states of the temperature dependent nominal models are kept consistent. Once all of the components are discretized and the interconnected model is constructed, the robust synthesis problem, as shown in Figure 6.1, is setup.

6.4.6 Deriving a D-Scale

D-K synthesis is used on the interconnected model, as shown in Figure 6.2, at a single temperature point to construct a D-scale. D-K synthesis is an iterative method that designs a controller which minimizes the upper bound on the robust performance, γ , of the system. Using the function `dksyn` from MATLAB, D-K synthesis was done on the interconnected model at T_l for all l . Table 6.2 shows the lowest upper bound on γ that could be achieved for each l using the function `dksyn`.

l	1	2	3	4	5	6
Internal Temperature ($^{\circ}\text{C}$)	16.3	20.9	30.6	40.1	50.0	59.6
Robust Performance (γ)	1.451	1.448	1.449	1.455	1.456	1.456

Table 6.2: The robust performance achieved by D-K synthesis at each temperature point.

The D-scale that is derived at one temperature is not optimal for other temperature points. Thus, the D-scale derived from the temperature point with the largest robust performance was chosen. By choosing the "worst" temperature point, the temperature with the worst performance would not become even worse. As seen from Table 6.2, the worst temperature point was either $T_5 = 50.0^{\circ}\text{C}$ or $T_6 = 59.6^{\circ}\text{C}$. Since, T_1 and T_6 are chosen as the reference temperature points for the gain-scheduling, the D-scale derived for T_6 was chosen.

The HDD system is setup with two independent uncertainty channels, and thus the

D-scale is structured such that,

$$D_{scale} = \text{diag}(d_1, d_2, I) \quad (6.5)$$

where d_1 and $d_2 \in \mathbb{C}$. Figure 6.11 shows the set of D-scale that was derived for the temperature dependent system. Once the D-scale was derived, it was used to scale the uncertainty channels of the temperature dependent system. By using an LTI D-scale for all temperature points, the states were kept consistent.

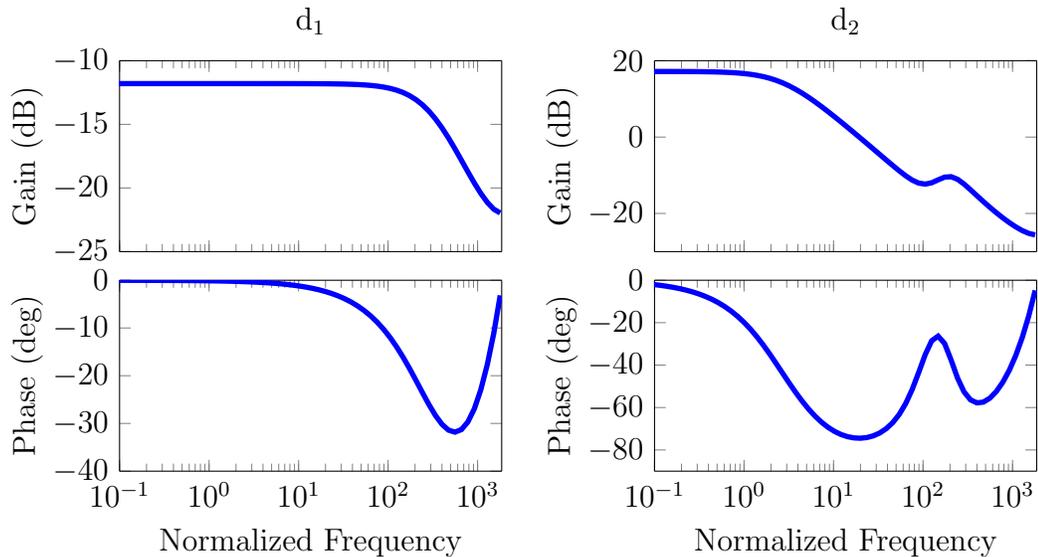


Figure 6.11: The Bode plot of the D-scale used to design a temperature dependent controller. The figure on the left represents d_1 and figure on the right represents d_2 .

6.4.7 Conditioning State-Space Matrices For Numerical Reasons

In certain cases, the numerical conditioning of the state space matrices may be poor. This can affect the results from `hinfsyn`, and therefore a balancing of state-space realization may need to be done. Balancing the system at each temperature point individually will not maintain the state-space variables to be consistent. Therefore a transformation matrix is derived using a function such as `balreal` on MATLAB for the temperature dependent system at some temperature. The systems at other temperature points are balanced using the same transformation matrix. By using the same transformation matrix, the states are kept consistent. Although the transformation matrix is not optimal at other temperature points, due to their similarity, the numerical conditioning is improved at all temperature points.

6.4.8 Gain Scheduled Controller

Once the D-scale for the system was derived, the scaled interconnected model was constructed. The scaled interconnected model is defined as,

$$P_{scaled} = D_{scale}(z)P(z, T)D_{scale}(z)^{-1} \quad (6.6)$$

The function `hinfsyn` from MATLAB was used with a fixed performance level of $\gamma = 1.456$ at each temperature point to design temperature specific controllers, $K_{T_l}(z)$ at all T_l . The γ was derived by `dksyn` in MATLAB concurrently with the derivation of the D-scale. The state-space matrices from K_{T_1} and K_{T_6} are used to create the temperature dependent controller $K_{dep}(z, T)$. The temperature specific controllers, $K_{T_l}(z)$ are also used to validate $K_{dep}(z, T)$.

For the HDD system, the open loop response, L is defined as,

$$L(s) = KG = \begin{bmatrix} K_V & K_M \end{bmatrix} \begin{bmatrix} G_V \\ G_M \end{bmatrix} = K_V G_V + K_M G_M \quad (6.7)$$

where K_V and K_M are the VCM and MA component of the controller, respectively, and G_V and G_M are the VCM and MA models, respectively. The closed loop sensitivity, S , is defined as,

$$S = \frac{1}{1 + L} \quad (6.8)$$

In addition to the nominal models, the experimental FRD, $D_k(j\omega_j, T_l)$ can be used in lieu of $G(z, T)$ to simulate the open-loop response and closed-loop sensitivity of the system.

Figure 6.12 shows the Bode plot of the temperature specific controller at 30.6°C, $K_{T_3}(z)$, and the gain-scheduled controller at 30.6°C, $K_{dep}(z, T_3)$. It can be seen that the gain-scheduled controller is very similar to the temperature specific controller and no discernible differences are visible in the Bode plot.

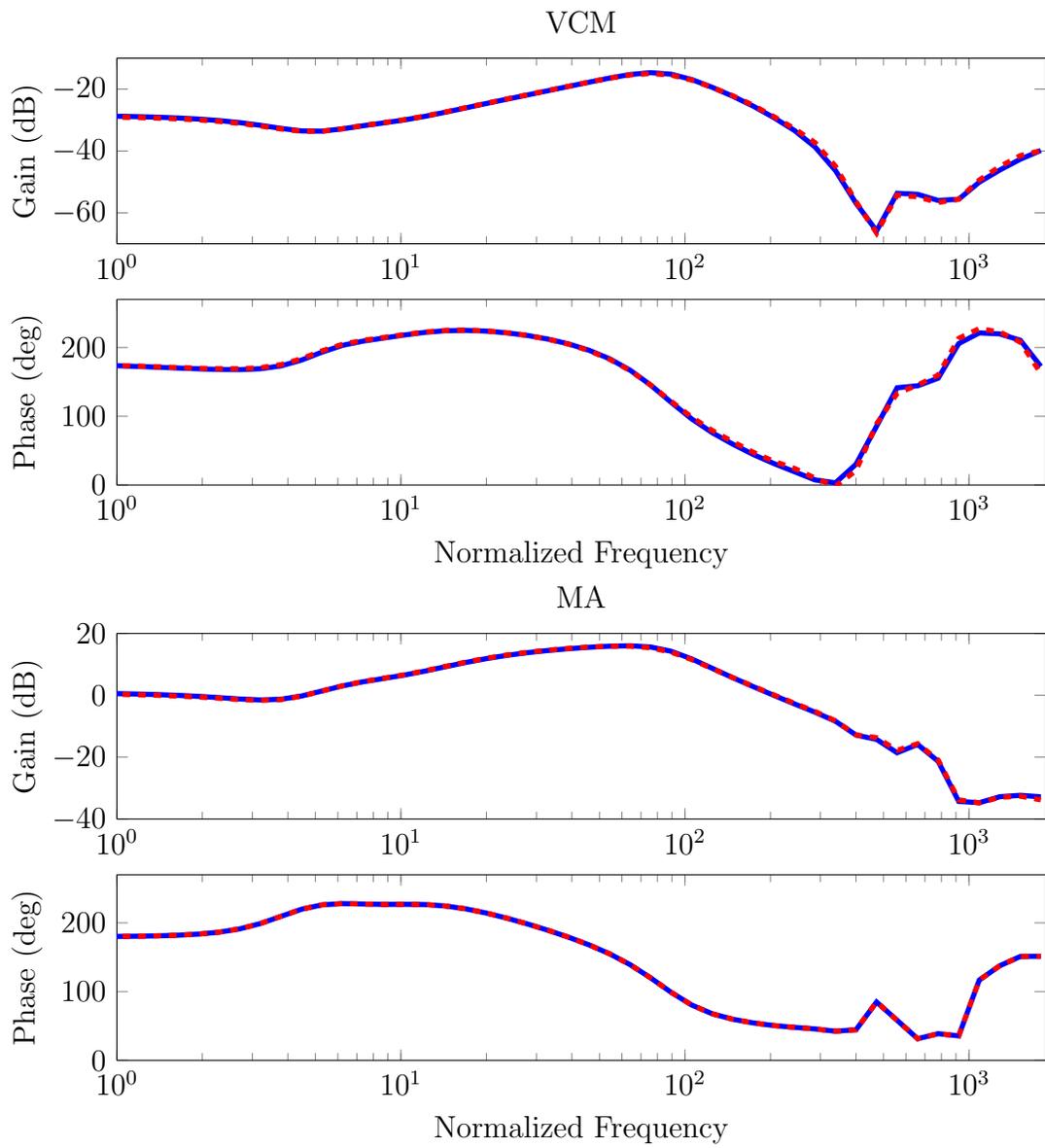


Figure 6.12: The Bode plot of the temperature specific controller designed at 30.6°C — and linearly interpolated controller at 30.6°C - - .

Figure 6.13 shows the open-loop response and closed-loop sensitivity of $K_{T_3}(z)$ and $K_{dep}(z, T_3)$ using the experimental FRD, $D_k(j\omega_f, T_3)$. Again, there is no significant difference between the two controllers. The linear interpolation of the temperature specific controllers between T_1 and T_6 is able to produce the same control effort as $K_{T_3}(z)$.

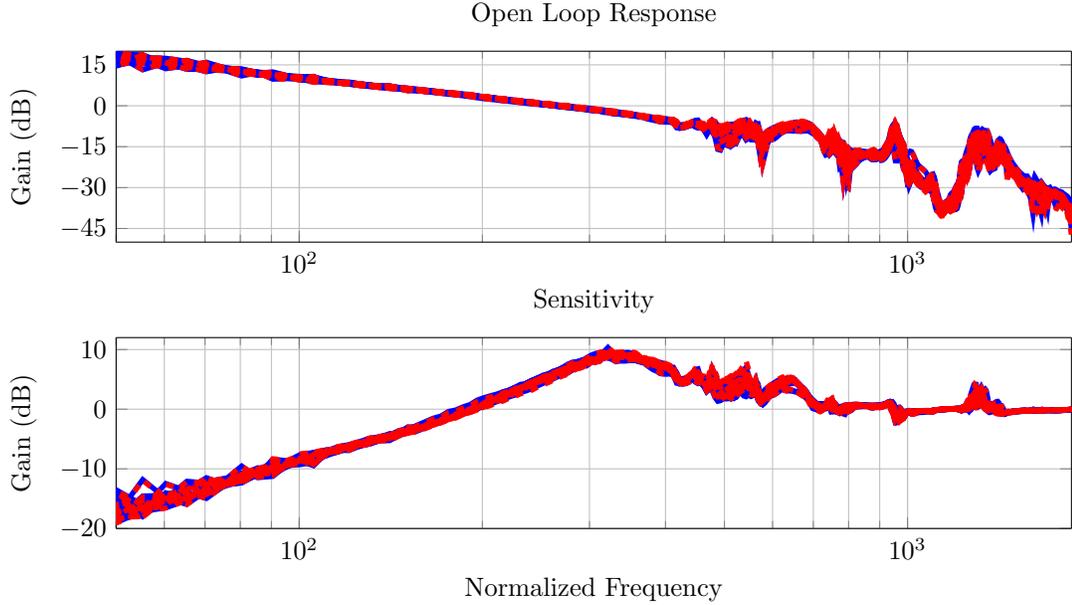


Figure 6.13: The open-loop and closed-loop sensitivity of the controller designed at 30.6°C — and gain scheduled controller at 30.6°C - - .

By linearly interpolating the temperature dependent scaled interconnected model with a fine mesh, the upper bound on the robust performance of the controllers can be compared. Figure 6.14 shows the upper bound on the robust performance for the temperature independent, temperature specific, and temperature dependent controllers. The temperature dependent controller, $K_{dep}(z, T)$, has a slightly greater upper bound on the robust performance compared to the temperature specific controllers, $K_{T_i}(z)$. Although there is some loss of performance from gain-scheduling, The temperature dependent controllers have nearly 3 times better performance compared to the temperature independent controller, $K_{ind}(z)$. Also through frequency domain analysis, it will be shown that the temperature dependent controller has better performance than the temperature independent controller.

Next, the nominal closed-loop sensitivity between the temperature dependent and independent controllers are compared. Figure 6.15 shows the nominal sensitivity of

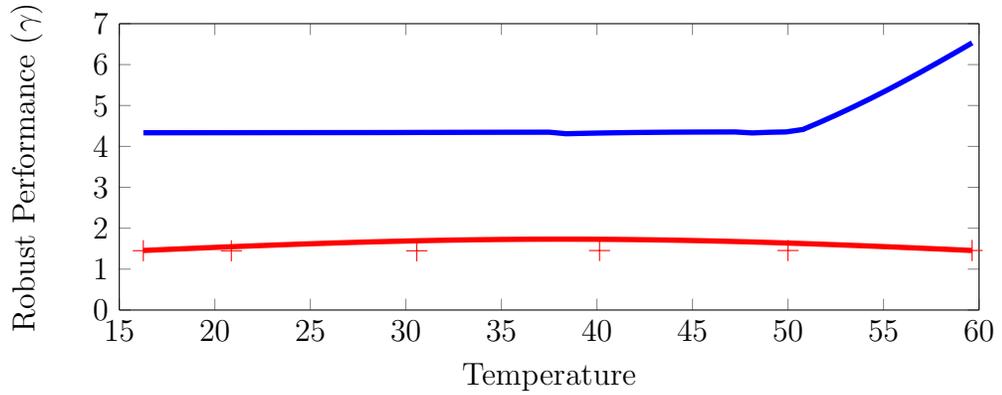


Figure 6.14: The upper bound on the robust performance of the temperature independent —, temperature specific +, and temperature dependent — controllers using the temperature dependent scaled interconnected model.

the HDD system using the temperature dependent nominal models at all T_i . The temperature dependent controllers is able to increase the bandwidth of the nominal sensitivity compared to the temperature independent controllers. Furthermore the temperature independent controller have large peaks that occur from frequency of 450 to 550. The temperature independent controller is not able to compensate for the shift in the high frequency modes that is characterized by the temperature dependent nominal models.

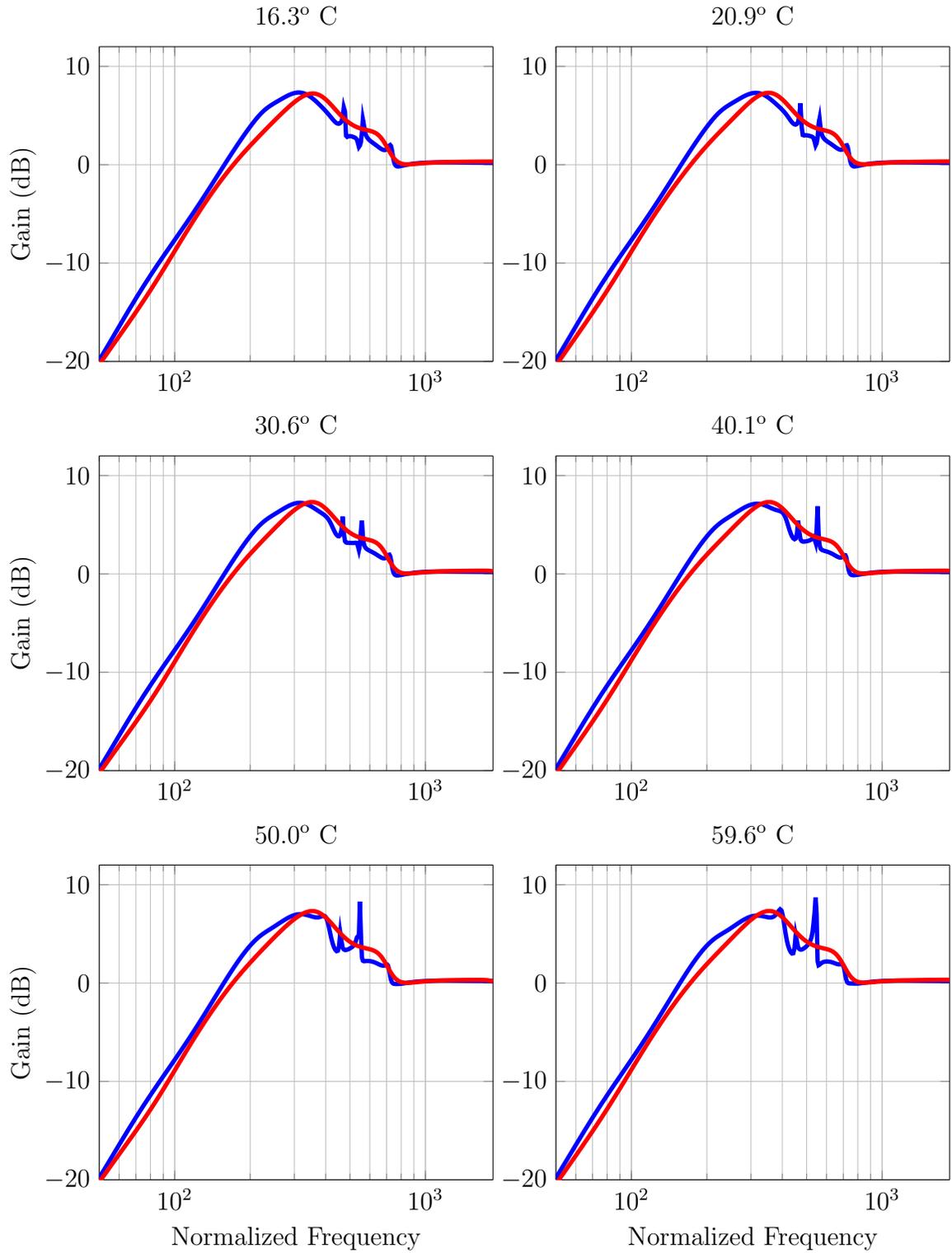


Figure 6.15: The nominal sensitivity of the HDD system using the temperature dependent model, temperature independent controller —, and temperature dependent controller —.

Now the experimental FRD at 30.6°C, $D_k(j\omega_j, T_3)$, are used to compare the performance between the temperature independent and dependent controllers. Figure 6.16 shows the open-loop response and closed-loop sensitivity using the experimental FRD at 30.6°C, $D_k(j\omega_j, T_3)$, for all k , with either the temperature independent controller and temperature dependent controller. The figure shows that the temperature dependent controller is able to increase the bandwidth of the closed-loop system while maintaining similar peak and high frequency gain.

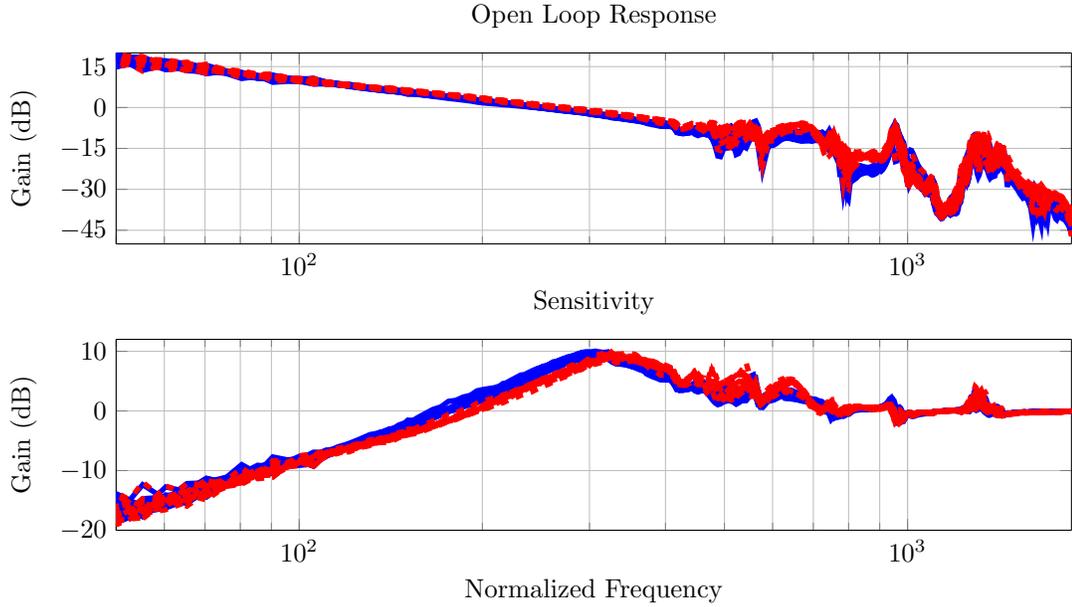


Figure 6.16: The open-loop and closed-loop sensitivity of the temperature independent controller — and temperature dependent controller - - at 30.6°C using the experimental FRD, $D_k(j\omega_j, T_3)$.

The closed-loop sensitivities computed from the set of experimental FRDs were averaged for all six experimental temperature point and is shown in Figure 6.17. By taking the mean of the sixteen different HDD data at each temperature point, the benefits of the temperature dependent controller can be seen more easily. At each temperature point, the temperature dependent controller is able to increase the bandwidth of the system by about 10%. The temperature dependent controller is also able to reduce the peak of the sensitivity except at 59.6°C. The temperature dependent controller is able to reject disturbances at higher frequency and thus has a higher performance compared to the temperature independent controller [31].

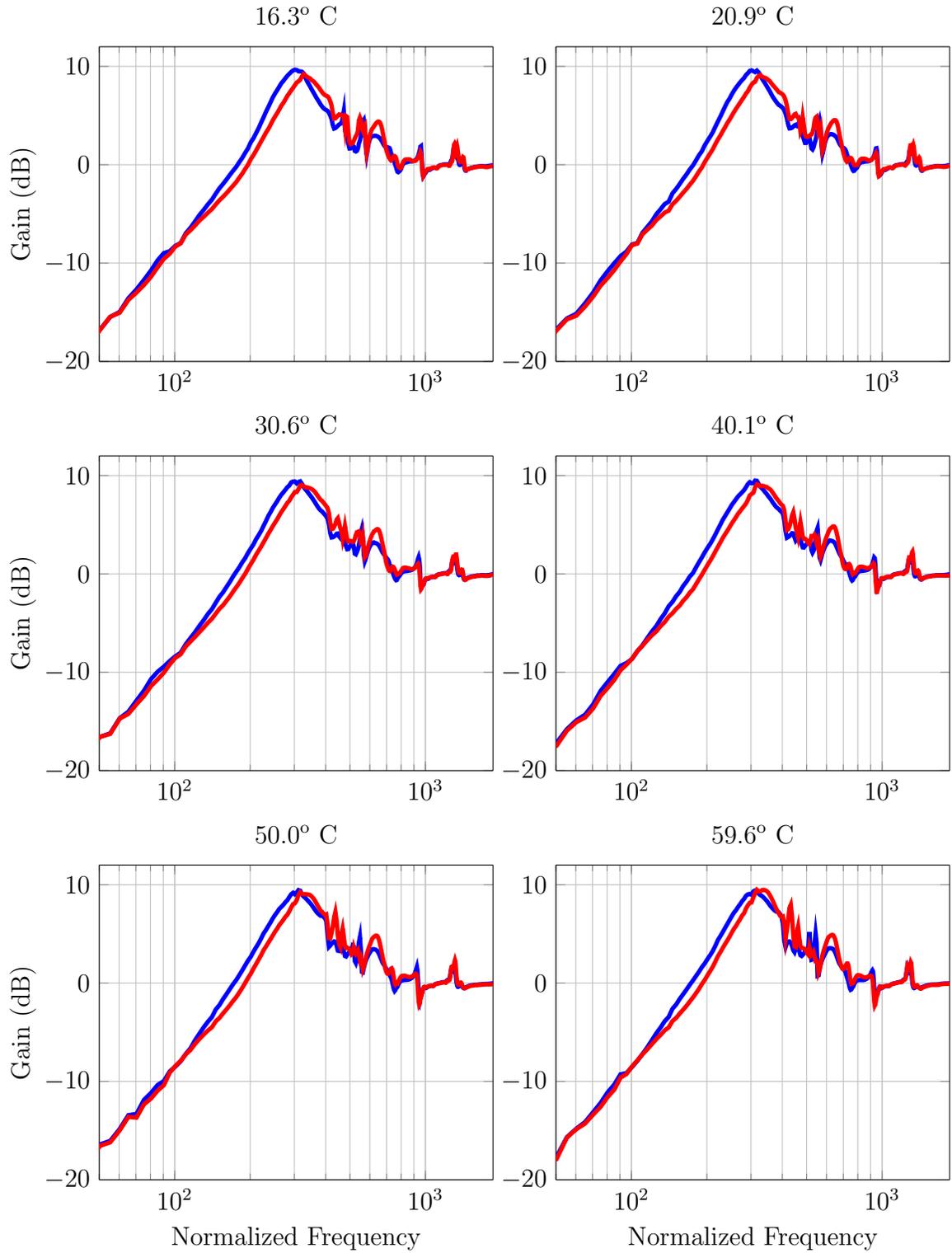


Figure 6.17: The average of the FRD of the closed-loop sensitivity based on the temperature independent controller — and temperature dependent controller —.

6.4.9 Experimental Results

The temperature dependent controller was implemented onto a HDD and tested at 30°C. Since the gain scheduled controller scheme has not been implemented into the HDD, the state-space matrices that are gain-scheduled at 30°C was used to test the controller. The resulting controller was compared to the temperature independent controller. The resulting closed-loop sensitivity is shown in Figure 6.18.

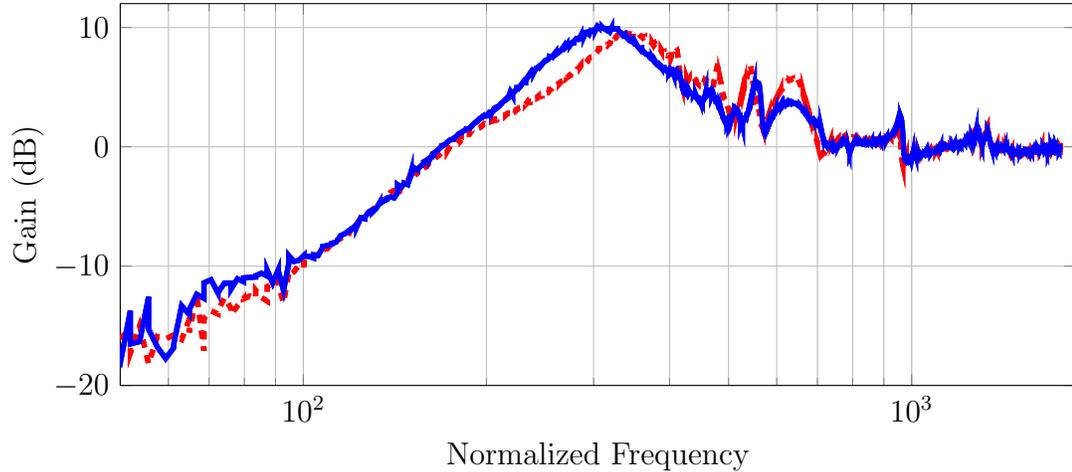


Figure 6.18: The closed-loop sensitivity measured from a HDD using temperature independent controller — and temperature dependent controller interpolated at 30°C - - .

It can be seen from the figure that simulated data and the experimental data do not match perfectly. The low frequency and high frequency sensitivity gains are similar between the experimental and simulated data. The increase in the bandwidth is less than expected based on the simulated sensitivity, but the gain above the bandwidth is reduced compared to the temperature independent controller. It can be seen that the gain from frequency of 200 to 300 have been reduced by about 2-3 dB. One of the reason for the difference in the expected bandwidth is that the DC gain of the MA actuator needs to be calibrated for each HDD. To calibrate the MA actuator, a frequency is chosen where the 0 dB crossing of the closed-loop sensitivity is expected. Therefore the calibration process can slightly alter the closed-loop sensitivity measured from the HDD. A different calibration could have resulted in a closed-loop sensitivity that is more similar to the simulated result.

6.5 Special H_∞ Problems

The procedure described in the previous section constructs linearly interpolatable generalized plants using the temperature dependent models and various weights. H_∞ synthesis is then performed at the largest and smallest temperature. The resulting controllers are then scheduled via linear interpolation. This ad-hoc method was shown to be effective in the previous section, however, this procedure requires some theoretical justification.

In particular, even if the generalized plants are linearly interpolatable, the H_∞ controllers created from the plant at two different parameter points are not always linearly interpolatable. This is because the solutions to the H_∞ problem is not inherently linear as a function of a parameter and specific conditions must be met for it to be linear. In this section, special cases of H_∞ problems that are applicable to the HDD system are introduced. These special cases help to simplify the H_∞ problem, and demonstrate the conditions that must be met for the controller to be linearly interpolatable. Most of the results mentioned in this section is from Zhou's robust control textbook [31].

6.5.1 Disturbance Feedforward and Full Information Problem

Let $P \in \mathbb{RL}_\infty^{(n_v+n_e+n_y) \times (n_w+n_d+n_u)}$ be an interconnected model used for the H_∞ synthesis problem as shown in Figure 6.19. This system has a state-space system defined as,

$$\begin{aligned} \dot{x} &= Ax + B_1p + B_2u \\ z &= C_1x + D_{11}p + D_{12}u \\ y &= C_2x + D_{21}p + D_{22}u \end{aligned} \tag{6.9}$$

where $p^T = \begin{bmatrix} w^T & d^T \end{bmatrix}$ and $z^T = \begin{bmatrix} v^T & e^T \end{bmatrix}$. A standard state-space realization representation is used as follows,

$$P = \left[\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{array} \right] \tag{6.10}$$

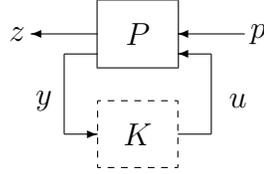


Figure 6.19: H_∞ problem

In this section, two special cases of state-space systems are introduced. First is the disturbance feedforward (DF) system which is defined as,

$$P_{DF} := \left[\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & D_{11} & D_{12} \\ C_2 & I & 0 \end{array} \right] \quad (6.11)$$

This system has the disturbance channel feed directly into the output.

Second is the full information (FI) system, which is defined as,

$$P_{FI} := \left[\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & D_{11} & D_{12} \\ \left[\begin{array}{c} I \\ 0 \end{array} \right] & \left[\begin{array}{c} 0 \\ I \end{array} \right] & \left[\begin{array}{c} 0 \\ 0 \end{array} \right] \end{array} \right] \quad (6.12)$$

The FI system has the states and the disturbances of the system as direct outputs. In Zhou's textbook in Section 12.2.2, it is shown that the DF and FI problems are identical under an assumption that $A - B_1 C_2$ is stable.

6.5.2 Equivalence of DF and FI Problem

A FI system can be transformed into a DF system with the following,

$$P_{DF} = \begin{bmatrix} I & 0 & 0 \\ 0 & C_2 & I \end{bmatrix} P_{FI} \quad (6.13)$$

This can be shown with a simple matrix operation.

It can be seen from the DF system, the equation for p can be derived in terms of y

and x .

$$y = C_2x + p \quad (6.14)$$

$$p = y - C_2x \quad (6.15)$$

A new state space system, V , that has inputs of y and u and outputs of x and p can be defined based on the DF system such that,

$$V = \left[\begin{array}{c|cc} A - B_1C_2 & B_1 & B_2 \\ \hline I & 0 & 0 \\ -C_2 & I & 0 \end{array} \right] \quad (6.16)$$

For the transformed system to be stable, $A - B_1C_2$ must be Hurwitz. Now the output of the DF system can be input into V as shown in Figure 6.20 to convert the system into a FI system. Since the DF and FI systems are equivalent, it can be shown that the optimal H_∞ controllers for the two systems are equivalent as well.

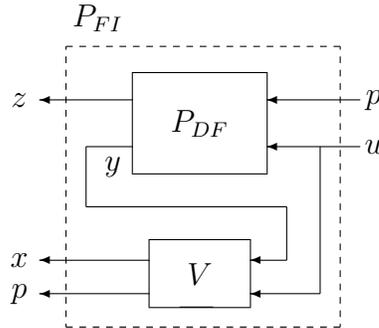


Figure 6.20: Transformation of DF to FI system.

6.5.3 Equivalent Controllers for DF and FI Problems

Since the DF and FI problems are equivalent, the controller that is designed for a DF system can be transformed for a FI system and vice versa. First a transformation system G_{DF} is defined such that,

$$G_{DF} = \left[\begin{array}{c|cc} A - B_1C_2 & B_1 & B_2 \\ \hline 0 & 0 & I \\ \left[\begin{array}{c} I \\ -C_2 \end{array} \right] & \left[\begin{array}{c} 0 \\ I \end{array} \right] & \left[\begin{array}{c} 0 \\ 0 \end{array} \right] \end{array} \right] \quad (6.17)$$

G_{DF} is the system V with the channel u passing through the system as one of the outputs.

The following theorem from Section 12.2 of Zhou's textbook shows that the DF and FI controllers are equivalent [31].

Theorem 1 $K_{FI} := K_{DF} \begin{bmatrix} C_2 & I \end{bmatrix}$ internally stabilizes P_{FI} if K_{DF} internally stabilizes P_{DF} , and

$$F_l \left(P_{FI}, K_{DF} \begin{bmatrix} C_2 & I \end{bmatrix} \right) = F_l(P_{DF}, K_{DF}). \quad (6.18)$$

If $A - B_1 C_2$ is Hurwitz, then $K_{DF} := F_l(G_{DF}, K_{FI})$ as shown in Figure 6.21, internally stabilizes P_{DF} if K_{FI} internally stabilizes P_{FI} . Furthermore,

$$F_l(P_{DF}, F_l(G_{DF}, K_{FI})) = F_l(P_{FI}, K_{FI}) \quad (6.19)$$

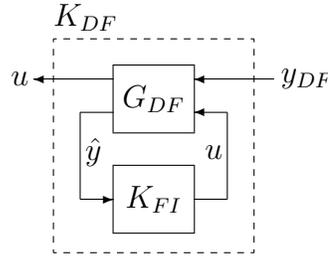


Figure 6.21: DF controller

This theorem follows from the equivalence of DF and FI problems.

6.5.4 H_∞ Synthesis For DF and FI Systems

In this section a brief overview on the algorithm for H_∞ synthesis for a DF system is described. The following results on H_∞ synthesis is from Chapter 17 of Zhou's textbook on general H_∞ solutions [31]. Once the transformation system, G_{DF} , is constructed, a DF problem can be transformed into a FI controller as shown in previous section. For a DF H_∞ problem where,

$$P_{DF} := \left[\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & D_{11} & D_{12} \\ C_2 & I & 0 \end{array} \right] \quad (6.20)$$

the following assumptions are made.

1. (A, B_2) is stabilizable;
2. $D_{12} = \begin{bmatrix} 0 \\ I \end{bmatrix}$;
3. $\begin{bmatrix} A - j\omega I & B_2 \\ C_1 & D_{12} \end{bmatrix}$ has full column rank for all ω .
4. $\begin{bmatrix} A - j\omega I & B_1 \\ C_2 & D_{21} \end{bmatrix}$ has full row rank for all ω .
5. $A - B_1 C_2$ is stable.

With the transformation system, G_{DF} , as shown in Equation 6.17, the H_∞ controller for a FI system can be transformed into a controller for a DF system. The following problem is solved to design a H_∞ controller for a FI system. Let $\gamma > 0$ be given and define,

$$R := D_{1\bullet}^* D_{1\bullet} - \begin{bmatrix} \gamma^2 I_{m1} & 0 \\ 0 & 0 \end{bmatrix} \quad (6.21)$$

where $D_{1\bullet} := \begin{bmatrix} D_{11} & D_{12} \end{bmatrix}$.

Then a Hamiltonian can be defined such that,

$$H_\infty := \begin{bmatrix} A_H & R_H \\ -Q_H & -A_H^* \end{bmatrix} := \begin{bmatrix} A & 0 \\ -C_1^* C_1 & -A^* \end{bmatrix} - \begin{bmatrix} B \\ -C_1^* D_{1\bullet} \end{bmatrix} R^{-1} \begin{bmatrix} D_{1\bullet}^* C_1 & B^* \end{bmatrix} \quad (6.22)$$

where $B := \begin{bmatrix} B_1 & B_2 \end{bmatrix}$. Then, X_∞ is the solution to,

$$A_H^* X_\infty + X_\infty A_H + X_\infty R_H X_\infty + Q_H = 0 \quad (6.23)$$

which is an algebraic Riccati Equation (ARE). There are standard solvers on MATLAB that can be used to solve for X_∞ .

The H_∞ controller for the FI system is then constructed as,

$$K_{FI} = \begin{bmatrix} k_x & k_d \end{bmatrix} = \begin{bmatrix} \left[D_{12}^* D_{11} & I \right] F & -D_{12}^* D_{11} \end{bmatrix} \quad (6.24)$$

where

$$F := -R^{-1} [D_{1\bullet}^* C_1 + B^* X_\infty] \quad (6.25)$$

The H_∞ controller for a FI problem is a static system. Once the FI controller is created, a DF controller can be derived with the use of G_{DF} as shown in section 6.5.3 [31].

It is noted here that the greatest gain of $F_l(P, K)$ is γ and the objective of the H_∞ synthesis is to design a controller such that γ is minimized. In function such as `hinfsyn`, a bisection algorithm is used to find a controller with a minimum γ that will stabilize the system.

6.6 Interpolation of HDD Controller

In this section, it will be shown that the HDD control design problem can be setup as a DF problem, such that the conditions required for a synthesis of DF and FI controllers are fulfilled. Then the conditions required for the DF and FI controller to be linearly interpolatable for a static D-scale is shown. At the end of the section, separation theory is briefly described to show that the interpolation method is valid using an LTI D-scale.

6.6.1 HDD Problem as a DF System

For the HDD problem to be setup as a DF system, the state-space matrices D_{21} and D_{22} must be transformed into identity and zero matrices, respectively. This transformation is possible if the uncertainty weights are setup such that the input uncertainty weights, W_{2V} and W_{2M} , are chosen to be static gains. Therefore the output uncertainty weights, W_{1V} and W_{1M} , contain all of the dynamic components of the uncertainty weights. With this condition, the system from inputs $p(= \begin{bmatrix} w^T & d^T \end{bmatrix}^T)$ to output y are static, and with some block diagram manipulation it can be seen from Figure 6.22 that inputs w_1 , w_2 , and d can enter additively to the same location. It will be shown later that this allows the multiple inputs w_1 , w_2 , and d to be simplified

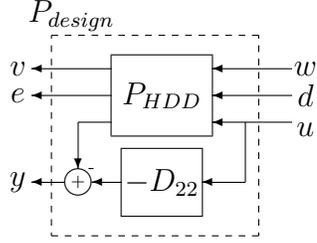


Figure 6.23: Robust synthesis problem with D_{22} loop-shifted.

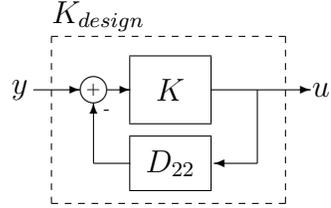


Figure 6.24: Robust controller with D_{22} loop-shifted.

	v	e	y	w	d	u
Dimension	2	3	1	2	1	2

Table 6.3: The input and output channel dimensions of the interconnected model.

D_{12} and D_{21} such that,

$$D_{12} = Q_{12} \begin{bmatrix} 0 \\ I \end{bmatrix} r_{12} \quad (6.28)$$

and

$$D_{21} = r_{21} \begin{bmatrix} 0 & I \end{bmatrix} Q_{21} \quad (6.29)$$

where Q_{12} and Q_{21} matrices are square and unitary. Furthermore matrix r_{12} and r_{21} are invertible, thus

$$Q_{12}^* D_{12} r_{12}^{-1} = \begin{bmatrix} 0 \\ I \end{bmatrix} \quad (6.30)$$

and

$$r_{21}^{-1} D_{21} Q_{21}^* = \begin{bmatrix} 0 & I \end{bmatrix} \quad (6.31)$$

By using the transformation matrices from the SVD of D_{21} the following normalization

can be done.

$$L_1 P_{HDD} R_1 = \begin{bmatrix} I_e & 0 \\ 0 & r_{21}^{-1} \end{bmatrix} \left[\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & 0 \end{array} \right] \begin{bmatrix} Q_{21}^* & 0 \\ 0 & I_y \end{bmatrix} \quad (6.32)$$

$$= \begin{bmatrix} A & B_1 Q_{21}^* & B_2 \\ \hline C_1 & D_{11} Q_{21}^* & D_{12} \\ r_{21}^{-1} C_2 & r_{21}^{-1} D_{21} Q_{21}^* & 0 \end{bmatrix} \quad (6.33)$$

$$= \begin{bmatrix} A & B_1 Q_{21}^* & B_2 \\ \hline C_1 & D_{11} Q_{21}^* & D_{12} \\ r_{21}^{-1} C_2 & \begin{bmatrix} 0 & I \end{bmatrix} & 0 \end{bmatrix} \quad (6.34)$$

For the problem to be DF, the matrix D_{21} must be an identity matrix. Since, as seen from Figure 6.22, the inputs w_1 , w_2 and d all add into the system at the same point, the three inputs can be consolidated into one input such that,

$$L_1 P_{HDD} R_1 = \begin{bmatrix} A & \bar{B}_1 & B_2 \\ \hline C_1 & \bar{D}_{11} & D_{12} \\ r_{21}^{-1} C_2 & 1 & 0 \end{bmatrix} \quad (6.35)$$

where the dimensions of $\bar{B}_1 = BQ_{21}^*$ and $\bar{D}_{11} = D_{11}Q_{21}^*$ have been truncated to match that of $D_{21} = 1$.

Now using the matrices from SVD of D_{12} , The HDD system is further normalized such that,

$$L_2 P_{HDD} R_2 = \begin{bmatrix} Q_{12}^* & 0 \\ 0 & I_y \end{bmatrix} \left[\begin{array}{c|cc} A & \bar{B}_1 & B_2 \\ \hline C_1 & \bar{D}_{11} & D_{12} \\ r_{21}^{-1} C_2 & 1 & 0 \end{array} \right] \begin{bmatrix} I_e & 0 \\ 0 & r_{12}^{-1} \end{bmatrix} \quad (6.36)$$

$$= \begin{bmatrix} A & \bar{B}_1 & B_2 r_{12}^{-1} \\ \hline Q_{12}^* C_1 & Q_{12}^* \bar{D}_{11} & Q_{12}^* D_{12} r_{12}^{-1} \\ r_{21}^{-1} C_2 & 1 & 0 \end{bmatrix} \quad (6.37)$$

$$= \begin{bmatrix} A & \bar{B}_1 & \bar{B}_2 \\ \hline \bar{C}_1 & \bar{D}_{11} & \begin{bmatrix} 0 \\ I \end{bmatrix} \\ r_{21}^{-1} C_2 & 1 & 0 \end{bmatrix} \quad (6.38)$$

Now the HDD problem is setup as a DF problem that matches the assumptions on the state-space realization of the interconnected model. When the controller is designed based on the normalized system, the normalization must be reverted. Thus once the controller, $K(z)$ is designed based on the transformed P_{HDD} , the final controller for the HDD system will be,

$$K_{HDD} = r_{12}Kr_{21}(I + D_{22}r_{12}Kr_{21})^{-1} \quad (6.39)$$

The details of the normalization argument is from Section 17.2 of Zhou's textbook [31].

6.6.2 Interpolation of H_∞ Controller

In this section, the conditions required for H_∞ controller to be linearly interpolatable based on temperature will be shown. There are two systems that need to be a linear function of temperature for the DF controller to be interpolatable, the transformation system, G_{DF} and the Full Information controller, K_{FI} . First the system G_{DF} must be interpolatable, such that the linearly interpolatable DF system that is transformed into a FI system is still interpolatable. The DF system is assumed to be interpolatable from using the temperature dependent nominal models, LTI weights and LTI D-scale. The only state-space matrix of transformation system, G_{DF} , with a matrix multiplication is $A_{G_{DF}} = A - B_1C_2$, as seen in Equation 6.17. Therefore, G_{DF} is linearly interpolatable if either B_1 or C_2 is a constant and the other is linearly interpolatable.

Next the conditions required for K_{FI} to be linearly interpolatable will be shown. K_{FI} is a matrix composed of k_x and k_d , where,

$$k_x = \begin{bmatrix} D_{12}^*D_{11} & I \end{bmatrix} \left(- \begin{bmatrix} -\gamma^2 I_{m1} + D_{11}^*D_{11} & D_{11}^*D_{12} \\ D_{12}^*D_{11} & D_{12}^*D_{12} \end{bmatrix}^{-1} [D_{1\bullet}^*C_1 + B^*X_\infty] \right) \quad (6.40)$$

and

$$k_d = -D_{12}^*D_{11} \quad (6.41)$$

As a reminder, $D_{1\bullet} = \begin{bmatrix} D_{11} & D_{12} \end{bmatrix}$.

First, k_d is linearly interpolatable if either D_{12} or D_{11} is a constant and the other is linearly interpolatable. For the HDD system, the whole matrix D is a constant. Since there are double integrator like characteristic for both the VCM and MA models, the

D matrices for the temperature dependent nominal models are zero at all temperature. Furthermore, all of the weights are LTI systems thus the D matrix for P_{HDD} does not vary as a function of temperature. Thus the conditions for k_d are met.

The matrix k_x is interpolatable if X_∞ and C_1 are linearly interpolatable as a function of temperature, and γ , B , D_{11} and D_{12} are constants. Since X_∞ is a solution to an algebraic Riccati equation (ARE), the solution to the ARE must be linear as a function of temperature. Normally, the algebraic Riccati equation is a quadratic problem, however if R_H is small compared to the other matrices, then the equation becomes a linear problem where,

$$A_H^* X_\infty + X_\infty A_H + Q_H = 0 \quad (6.42)$$

If the ARE becomes a linear problem, its solution, X_∞ , is linearly interpolatable as well.

For R_H to be negligible, certain conditions must be met. From the Hamiltonian matrix, the matrix component of R_H exists only in the second term of Equation 6.22. The second term of the Hamiltonian is expanded such that,

$$\begin{aligned} & \begin{bmatrix} B \\ -C_1^* D_{1\bullet} \end{bmatrix} R^{-1} \begin{bmatrix} D_{1\bullet}^* C_1 & B^* \end{bmatrix} \\ &= \begin{bmatrix} B \\ -C_1^* D_{1\bullet} \end{bmatrix} \begin{bmatrix} -\gamma^2 I_{m1} + D_{11}^* D_{11} & D_{11}^* D_{12} \\ D_{12}^* D_{11} & D_{12}^* D_{12} \end{bmatrix}^{-1} \begin{bmatrix} D_{1\bullet}^* C_1 & B^* \end{bmatrix} \end{aligned} \quad (6.43)$$

For simplification assume that $D_{11} = 0$, and define $C_1 := \begin{bmatrix} C_{11} & C_{12} \end{bmatrix}$, where $C_{11} \in \mathbb{R}^{n_v \times n_A}$ and $C_{12} \in \mathbb{R}^{n_e \times n_A}$. Furthermore from the SVD normalization, $D_{12} = \begin{bmatrix} 0 & I_e \end{bmatrix}^T$. Thus the second term of Hamiltonian can be simplified into,

$$\begin{aligned} & \begin{bmatrix} B \\ -C_1^* D_{1\bullet} \end{bmatrix} \begin{bmatrix} -\gamma^{-2} I_{m1} & D_{11}^* D_{12} \\ D_{12}^* D_{11} & (D_{12}^* D_{12})^{-1} \end{bmatrix} \begin{bmatrix} D_{1\bullet}^* C_1 & B^* \end{bmatrix} \\ &= \begin{bmatrix} \begin{bmatrix} B_1 & B_2 \end{bmatrix} \\ -C_1^* \begin{bmatrix} 0 & D_{12} \end{bmatrix} \end{bmatrix} \begin{bmatrix} -\gamma^{-2} I_{m1} & 0 \\ 0 & I_e \end{bmatrix} \begin{bmatrix} \begin{bmatrix} 0 \\ D_{12}^* \end{bmatrix} C_1 \\ \begin{bmatrix} B_1^* \\ B_2^* \end{bmatrix} \end{bmatrix} \\ &= \begin{bmatrix} B_2 D_{12}^* C_{12} & -\gamma^{-2} B_1 B_1^* + B_2 B_2^* \\ -C_{12}^* D_{12} D_{12}^* C_{12} & -C_{12}^* D_{12} B_2^* \end{bmatrix}, \end{aligned} \quad (6.44)$$

thus $R_H = -\gamma^{-2}B_1B_1^* + B_2B_2^*$. If $-\gamma^2B_1B_1^* + B_2B_2^*$ is sufficiently small compared to A_H and $-Q_H$, then the solution to ARE, X_∞ , is linearly interpolatable.

To summarize, K_{DF} is interpolatable if the following are true,

1. A matrix is linearly interpolatable from two end points of the parameter.
2. B , C , and D matrix are constant or close to a constant.
3. R_H is sufficiently small compared to A_H and Q_H

This shows that the A matrix can vary linearly, however the input-output relation between p , u , z , and y must be very similar at different values of the parameter. Realistically, the B , C , and D matrices can vary by small amount. For the HDD problem the C and D matrices were constants, however some entries of the B matrix varied up to 10%. Furthermore, in this section D_{11} was assumed to be a zero matrix to simplify the Hamiltonian equation. The HDD system did have a non-zero D_{11} matrix, however if the R_H entry of the Hamiltonian matrix is sufficiently small, the interpolation method is still valid. In this case, the 2-norm of R_H was 0.046, compared to 2-norms of A_H and Q_H with values of $7.7e5$ and $2.94e14$, respectively for $T = T_3$

6.7 Separation Theory

The previous sections describe the interpolation theory based on constant D-scale to maintain the HDD problem as a FI problem. In most cases, however, a dynamic D-scale is used to design a robust controller. The dynamic D-scale changes the HDD control problem from a FI problem into a more general case, however according to separation theory, it is possible for the interpolation method to be valid using a LTI dynamic D-scale.

The FI problem has a dual problem, which is the full control (FC) problem with the state-space realization,

$$P_{FC} := \left[\begin{array}{c|c|cc} A & B_1 & I & 0 \\ \hline C_1 & D_{11} & 0 & I \\ C_2 & D_{21} & 0 & 0 \end{array} \right] \quad (6.45)$$

Most control synthesis problems are combination of the FI and FC problem and the H_∞ synthesis method designs a controller based on these two problems. A simplified case is shown to demonstrate this characteristic of H_∞ synthesis. The following results are from Section 16.2.3 of Zhou's textbook [31].

Take a system, $P_{central}$, with a state-space realization of

$$P_{central} = \left[\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & 0 & D_{12} \\ C_2 & D_{21} & 0 \end{array} \right] \quad (6.46)$$

The following assumptions are made,

1. (A, B_1) is stabilizable and (C_1, A) is detectable;
2. (A, B_2) is stabilizable and (C_2, A) is detectable;
3. $D_{12}^* \begin{bmatrix} C_1 & D_{12} \end{bmatrix} = \begin{bmatrix} 0 & I \end{bmatrix}$;
4. $\begin{bmatrix} B_1 \\ D_{21} \end{bmatrix} D_{21}^* = \begin{bmatrix} 0 \\ I \end{bmatrix}$.

Then the system $P_{central}$ has a H_∞ controller, K , such that,

$$K := \left[\begin{array}{c|c} \hat{A}_\infty & -Z_\infty L_\infty \\ \hline F_\infty & 0 \end{array} \right] \quad (6.47)$$

where

$$\hat{A}_\infty := A + \gamma^{-2} B_1 B_1^* X_\infty + B_2 F_\infty + Z_\infty L_\infty C_2 \quad (6.48)$$

$$F_\infty := -B_2^* X_\infty \quad (6.49)$$

$$L_\infty := -Y_\infty C_2^* \quad (6.50)$$

$$Z_\infty := (I - \gamma^{-2} Y_\infty X_\infty)^{-1} \quad (6.51)$$

The X_∞ and Y_∞ are solutions to the Hamiltonian matrices,

$$H_\infty := \begin{bmatrix} A & \gamma^{-2} B_1 B_1^* - B_2 B_2^* \\ -C_1^* C_1 & -A^* \end{bmatrix} \quad (6.52)$$

and

$$J_\infty := \begin{bmatrix} A^* & \gamma^{-2}C_1C_1^* - C_2^*C_2 \\ -B_1^*B_1 & -A \end{bmatrix} \quad (6.53)$$

It is shown in Section 16.8 of Zhou’s textbook that the controller component that utilizes Y_∞ is effectively the output estimator for the system $P_{central}$. The controller component of Y_∞ is designed to minimize the gain from the FC component of the system and is separate from the FI component of the system.

With a HDD problem, the dynamic D-scale is the only possible FC component of the scaled interconnected model. Since the same D-scale is used for all temperature points, the FC controller component must be constant across all temperature points. Therefore, the matrix Y_∞ must be a constant across all temperature points as well. The FI and FC component of the controller interacts linearly, even in the most general case as shown in Section 17.1 from Zhou’s textbook [31]. Thus if the FI controller is linearly interpolatable with constant D-scale, the H_∞ controller created using a LTI dynamic D-scale is linearly interpolatable.

6.8 Conclusion and Future Work

A method that can be used to design a temperature dependent controller for a HDD system was described in this chapter. It was shown that the temperature dependent controller is able to increase the performance of the HDD system compared to a temperature independent controller. Because the firmware of the HDD system needed to be improved to properly implement a gain-scheduled controller, the controller was not validated under a changing temperature condition. Instead, the controller was validated on the HDD system by implementing the linearly interpolated state-space matrix for a single temperature point. Furthermore, the H_∞ problem was closely examined to describe the conditions required for the interpolation method to be valid. For the next step, the gain-scheduled controller needs to be implemented into the HDD system, such that the performance of the system can be analyzed as the temperature of the system changes very slowly.

Chapter 7

Conclusion and Future Work

In this thesis, a method of designing a temperature dependent controller for a hard disk drive (HDD) system was described to increase its performance.

Initially, a method to design a temperature independent controller using standard D-K synthesis was described in Chapter 3. The uncertainty model for a two input one output system was defined and a basic modeling method using the average of the data was described. The MATLAB functions that were used to design the weights for the interconnected system were described as well. The temperature independent controller was used as a baseline system, which the temperature dependent controller was compared against.

Secondly, an optimal nominal modeling method based on convex optimization was described in Chapter 4. The optimization problem was set up to find the optimal frequency response data which would minimize the uncertainty of the model. The practical issues, such as designing an uncertainty weight with the presence of measurement noise, were discussed in the chapter. Furthermore the different types of models that would be applicable to the HDD system were shown.

Thirdly, a method to create a temperature dependent model for the HDD was described in Chapter 5. The benefits of temperature dependent model over temperature independent model was shown by comparing the optimal uncertainty weight required for the two types of models. Then the step by step method used to design the temperature dependent nominal models was described.

Finally, Chapter 6 described the method of designing a temperature dependent controller for the HDD system. Chapter 6 also showed that the temperature dependent controller had better performance over the temperature independent controller. The temperature dependent controller increased the bandwidth frequency of the closed-loop sensitivity while maintaining the same peak gain at six different temperatures. The H_∞ synthesis method was explored in detail to explain the validity of the temperature dependent controller design method.

7.1 Recommendations For Future Work

Based on this thesis there are couple of research projects that can work as a continuation of this work. First is the exploration of different control schemes for the HDD system. With the parallel control architecture, if the micro actuator (MA) fails, the system is not guaranteed to be stable. Therefore some HDD control system will design a VCM controller first, then design a MA controller [34]. Different control system schematics may be able to improve the performance of the system. Second is exploring other types of uncertainty models, such as uncertainty parametric models. By testing out different uncertainty models, it may be possible to design better controllers. Finally, the temperature dependent controller was designed with the assumption that the temperature was slow varying. As the HDDs did not have the proper firmware to run a temperature dependent controller, experiments with changing temperature have not been done. It would be interesting to test how slow the changes in temperature must be for the performance of the system to be not affected.

The temperature dependent controller was able to increase the performance of the system by decreasing the uncertainty of the system at each temperature point. There is, however, a fundamental limit to the performance of a system even if the uncertainty of the system was minimized as much as possible. In control theory, the water-bed effect of sensitivity gain is a known phenomenon, and sets a fundamental limit to performance that can be achieved from a feedback control system [31]. Although the water-bed effect applies to SISO systems, its effect for a MISO system is not well known yet and would be possible future work. In Chapter 3 it was shown from two different controllers that there are trade offs between bandwidth, peak gain, and low frequency gain of the close-loop sensitivity. Once the limit is reached, the mechanical system itself need to be improved to increase its performance.

To improve the HDD performance some research groups have worked on improving the micro actuator [81]. Recently, there has been work done on adding a third actuator to create a three-stage actuation system [82]. Other groups look to improve the system by implementing additional sensors [83]. Another factor that could limit the performance of the controller is the sampling frequency based on the number of servo sectors [81]. Depending on how the actuator system is improved, the control synthesis method will have to change to adapt to the new system.

This thesis focused on designing a controller for the track following mode, where the magnetic head is kept on the center of the track while the HDD reads or writes data. During track following, the primary goal is to reject any disturbances to maintain the magnetic head at the center of the track. During track seeking, however, the HDD must be able to follow a reference pathway and reject disturbances to move the magnetic head across thousands of tracks. Due to the range of motion, saturation of the MA can limit the performance and in some cases cause the system to become unstable. It has been shown, however, that having both the VCM and MA enabled for the track seeking mode can decrease seek time [33].

With the continuous demand for more storage, manufacturers will continue to push limits of HDDs, and there will continue to be a need for more innovative solutions [2]. Starting from the large system in the RAMAC, the HDD has gone through a significant transformation in the last 60 years. The HDD is packed with state of the art technology, and the requirement of robust control theory to achieve the desired performance is one such indication. As long as there are continued demand for HDD, it will continue to provide variety of research projects in the future.

References

- [1] Narayanan, D., Thereska, E., Donnelly, A., Elnikety, S., and Rowstron, A., “Migrating Server Storage to SSDs: Analysis of Tradeoffs,” *Proceedings of The 4th ACM European conference on Computer systems*, ACM, 2009, pp. 145–158.
- [2] Coughlin, T., “HDD Annual Unit Shipments Increase In 2014,” *Forbes*, January 2015, [<http://www.forbes.com/sites/tomcoughlin/2015/01/29/hdd-annual-unit-shipments-increase-in-2014/>], Online: posted 29-January-2015].
- [3] Thompson, D. and Best, J., “The Future of Magnetic Data Storage Technology,” *IBM Journal of Research and Development*, Vol. 44, No. 3, 2000, pp. 311–322.
- [4] Al Mamun, A., Lee, H., Guo, Gand Wong, E., and Ye, W., “Measurement of Position Offset in Hard Disk Drive Using Dual Frequency Servo Bursts,” *Instrumentation and Measurement, IEEE Transactions on*, Vol. 52, No. 6, 2003, pp. 1870–1880.
- [5] “Exceeding Capacity, Speed and Performance Expectations: Seagate AcuTrac Technology Defines New Standards,” Tech. rep., Seagate, 2011.
- [6] Al Mamun, A., Guo, G., and Bi, C., *Hard Disk Drive: Mechatronics and Control*, CRC press, 2006.
- [7] Balas, G., Packard, A., and Seiler, P., “Uncertain Model Set Calculation from Frequency Domain Data,” *Model-Based Control*, edited by P. Hof, C. Scherer, and P. Heuberger, Springer US, 2009, pp. 89–105.
- [8] Abramovitch, D. and Frankling, G., “A Brief History of Disk Drive Control,” *IEEE Control Systems Magazine*, Vol. 272, No. 1708/02, 2002.

- [9] Chen, B. M., Lee, T. H., Peng, K., and Venkataramanan, V., *Hard Disk Drive Servo Systems*, Springer Science & Business Media, 2006.
- [10] Grochowski, E. and Hoyt, R. F., “Future Trends in Hard Disk Drives,” *IEEE Transactions on Magnetics*, Vol. 32, No. 3, 1996.
- [11] Grochowski, E. and Halem, R. D., “Technological Impact of Magnetic Hard Disk Drives on Storage Systems,” *IBM Systems Journal*, Vol. 42, No. 2, 2003, pp. 338–346.
- [12] Cherubini, G., Chung, C. C., Messner, W. C., and Moheimani, S. R., “Control Methods in Data-Storage Systems,” *IEEE Transactions on Control Systems Technology*, Vol. 20, No. 2, 2012, pp. 296–322.
- [13] Ross, C. A., “Patterned Magnetic Recording Media,” *Annual Review of Materials Research*, Vol. 31, No. 1, 2001, pp. 203–235.
- [14] Wood, R., “Future Hard Disk Drive Systems,” *Journal of Magnetism and Magnetic Materials*, Vol. 321, No. 6, 2009, pp. 555–561.
- [15] Abramovitch, D. and Franklin, G., “Disk Drive Control: The Early Years,” *Annual Reviews in Control*, Vol. 26, No. 2, 2002, pp. 229–242.
- [16] Horowitz, R., Li, Y., Oldham, K., Kon, S., and Huang, X., “Dual-Stage Servo Systems and Vibration Compensation in Computer Hard Disk Drives,” *Control Engineering Practice*, Vol. 15, No. 3, 2007, pp. 291–305.
- [17] Freitas, R. F. and Wilcke, W. W., “Storage-Class Memory: The Next Storage System Technology,” *IBM Journal of Research and Development*, Vol. 52, No. 4.5, 2008, pp. 439–447.
- [18] Chen, X. and Tomizuka, M., “A Minimum Parameter Adaptive Approach For Rejecting Multiple Narrow-Band Disturbances With Application to Hard Disk Drives,” *Control Systems Technology, IEEE Transactions on*, Vol. 20, No. 2, 2012, pp. 408–415.
- [19] Chew, K. K. and Tomizuka, M., “Digital Control of Repetitive Errors in Disk Drive Systems,” *American Control Conference, 1989*, IEEE, 1989, pp. 540–548.

- [20] Graham, M. and De Callafon, R., “An Iterative Learning Design For Repeatable Runout Cancellation in Disk Drives,” *Control Systems Technology, IEEE Transactions on*, Vol. 14, No. 3, 2006, pp. 474–482.
- [21] Jia, Q.-W., Wang, Z.-F., and Wang, F.-C., “Repeatable Runout Disturbance Compensation With a New Data Collection Method For Hard Disk Drive,” *Magnetics, IEEE Transactions on*, Vol. 41, No. 2, 2005, pp. 791–796.
- [22] Morris, J. and Zirps, T., “Compression and Storage of Written-In Error Compensation Tables in an Embedded Servo Disc Drive,” Sept. 10 2002, US Patent 6,449,116.
- [23] McAllister, J. S., “The Effect of Disk Platter Resonances on Track Misregistration in 3.5 inch Disk Drives,” *Magnetics, IEEE Transactions on*, Vol. 32, No. 3, 1996, pp. 1762–1766.
- [24] Jia, Q., Cao, S., Guo, G., and Yu, J., “External Disturbance Rejection By Use of an Add-on Nonlinear Controller in HDD Servo Systems,” *Magnetics, IEEE Transactions on*, Vol. 49, No. 6, 2013, pp. 2624–2627.
- [25] Rahman, A., Al Mamun, A., and Yao, K., “Discrete Time Adaptive Controller For Suppression of Resonance in Hard Disk Drive Servo System,” *International Journal of Control, Automation and Systems*, Vol. 13, No. 5, 2015, pp. 1161–1172.
- [26] Mori, K., Munemoto, T., Otsuki, H., Yamaguchi, Y., and Akagi, K., “A Dual-Stage Magnetic Disk Drive Actuator Using a Piezoelectric Device For a High Track Density,” *Magnetics, IEEE Transactions on*, Vol. 27, No. 6, 1991, pp. 5298–5300.
- [27] Nagamune, R., Huang, X., and Horowitz, R., “Robust Control Synthesis Techniques For Multirate and Multisensing Track-Following Servo Systems in Hard Disk Drives,” *Journal of Dynamic Systems, Measurement, and Control*, Vol. 132, No. 2, 2006.
- [28] Boettcher, U., Callafon, R., and Talke, F., “Modeling and Control of a Dual Stage Actuator Hard Disk Drive,” *Journal of Advanced Mechanical Design, Systems, and Manufacturing*, Vol. 4, No. 1, 2010, pp. 107–118.

- [29] Numasato, H. and Tomizuka, M., “Settling Control and Performance of a Dual-Actuator System For Hard Disk Drives,” *Mechatronics, IEEE/ASME Transactions on*, Vol. 8, No. 4, 2003, pp. 431–438.
- [30] Schroeck, S. and Messner, W., “On Controller Design For Linear Time-Invariant Dual-Input Single-Output Systems,” *American Control Conference, 1999. Proceedings of The 1999*, Vol. 6, IEEE, 1999, pp. 4122–4126.
- [31] Zhou, K., Doyle, J., and Glover, K., *Robust and Optimal Control*, Prentice-Hall, 1996.
- [32] Takaishi, K., Imamura, T., Mizoshita, Y., Hasegawa, S., Ueno, T., and Yamada, T., “Microactuator Control For Disk Drive,” *Magnetics, IEEE Transactions on*, Vol. 32, No. 3, 1996, pp. 1863–1866.
- [33] Kobayashi, M. and Horowitz, R., “Track Seek Control For Hard Disk Dual-Stage Servo Systems,” *Magnetics, IEEE Transactions on*, Vol. 37, No. 2, 2001, pp. 949–954.
- [34] Bashash, S., “Robust Control Optimization For High Performance Track Following in Hard Disk Drives,” *American Control Conference (ACC), 2015*, IEEE, 2015, pp. 4634–4639.
- [35] Guo, L., Martin, D., and Brunnett, D., “Dual-Stage Actuator Servo Control For High Density Disk Drives,” *Advanced Intelligent Mechatronics, 1999. Proceedings. 1999 IEEE/ASME International Conference on*, IEEE, 1999, pp. 132–137.
- [36] Rotunno, M. and de Callafon, R., “Fixed Order H_∞ Control Design For Dual-Stage Hard Disk Drives,” *Decision and Control, 2000. Proceedings of The 39th IEEE Conference on*, Vol. 4, IEEE, 2000, pp. 3118–3119.
- [37] Oboe, R., Beghi, A., and Murari, B., “Modeling and Control of a Dual Stage Actuator Hard Disk Drive With Piezoelectric Secondary Actuator,” *Advanced Intelligent Mechatronics, 1999. Proceedings. 1999 IEEE/ASME International Conference on*, IEEE, 1999, pp. 138–143.
- [38] Conway, R., Felix, S., and Horowitz, R., “Model Reduction and Parametric Uncertainty Identification For Robust H_2 Control Synthesis For Dual-Stage Hard Disk Drives,” *Magnetics, IEEE Transactions on*, Vol. 43, No. 9, 2007, pp. 3763–3768.

- [39] Nie, J. and Horowitz, R., “Design and Implementation of Dual-Stage Track-Following Control For Hard Disk Drives,” *ASME 2009 Dynamic Systems and Control Conference*, American Society of Mechanical Engineers, 2009, pp. 565–572.
- [40] De Callafon, R., Nagamune, R., and Horowitz, R., “Robust Dynamic Modeling and Control of Dual-Stage Actuators,” *Magnetics, IEEE Transactions on*, Vol. 42, No. 2, 2006, pp. 247–254.
- [41] Nie, J., Conway, R., and Horowitz, R., “Optimal H_∞ Control For Linear Periodically Time-Varying Systems in Hard Disk Drives,” *Mechatronics, IEEE/ASME Transactions on*, Vol. 18, No. 1, 2013, pp. 212–220.
- [42] Huang, D., Venkataramanan, V., Xu, J.-X., and Huynh, T. C. T., “Contact-Induced Vibration in Dual-Stage Hard Disk Drive Servo Systems and Its Compensator Design,” *Industrial Electronics, IEEE Transactions on*, Vol. 61, No. 8, 2014, pp. 4052–4060.
- [43] Hong, F. and Pang, C. K., “A Peak Filtering Method With Improved Transient Response For Narrow-Band Disturbance Rejection in Hard Disk Drives,” *International Journal of Systems Science*, Vol. 45, No. 2, 2014, pp. 109–119.
- [44] Wong, D. and Hencsey, B., “A Controller Switching Design Approach Via Parameterization For Control of Hard Disk Drive Dual-Stage Actuators,” *American Control Conference (ACC), 2013*, IEEE, 2013, pp. 836–841.
- [45] Rahman, A., Al Mamun, A., Yao, K., and Daud, Y., “Discrete-Time Model Predictive Control For Head-Positioning Servomechanism in a Dual-Stage Hard Disk Drive,” *Mechatronics and Automation (ICMA), 2014 IEEE International Conference on*, IEEE, 2014, pp. 8–13.
- [46] Skogestad, S. and Postlethwaite, I., *Multivariable Feedback Control: Analysis and Design*, John Wiley & Sons, 2005.
- [47] MATLAB, *Robust Control Toolbox*, The MathWorks Inc.
- [48] Packard, A., Doyle, J., and Balas, G., “Linear, Multivariable Robust Control With a μ Perspective,” *Journal of Dynamic Systems, Measurement, and Control*, Vol. 115, No. 2B, 1993, pp. 426–438.

- [49] Hindi, H., Seong, C.-Y., and Boyd, S., “Computing Optimal Uncertainty Models From Frequency Domain Data,” *Decision and Control, 2002, Proceedings of The 41st IEEE Conference on*, Vol. 3, 2002, pp. 2898–2905.
- [50] Häggblom, K., “Data-Based Uncertainty Modeling by Convex Optimization Techniques,” *Proc. Int. Symp. Advanced Control of Chemical Processes*, 2006, pp. 91–96.
- [51] Häggblom, K., “Data-Based Modeling of Block-Diagonal Uncertainty by Convex Optimization,” *IEEE American Control Conference*, 2007, pp. 4637–4642.
- [52] Häggblom, K., “Convex Formulations For Data-Based Uncertainty Minimization of Linear Uncertainty Models,” *Control Automation Robotics & Vision (ICARCV), 2010 11th International Conference on*, IEEE, 2010, pp. 501–505.
- [53] Häggblom, K., “MIMO Uncertainty Model Identification of Time-Delay Systems,” *Advanced Control of Chemical Processes*, Vol. 8, 2012, pp. 385–390.
- [54] Boyd, S., El Ghaoui, L., Feron, E., and Balakrishnan, V., *Linear Matrix Inequalities in System and Control Theory*, Vol. 15 of *Studies in Applied Mathematics*, SIAM, 1994.
- [55] Boyd, S., El Ghaoui, L., Feron, E., and Balakrishnan, V., *Linear Matrix Inequalities in System and Control Theory*, Vol. 15 of *Studies in Applied Mathematics*, SIAM, 1994.
- [56] Zheng, J., Fu, M., Wang, Y., and C., D., “Nonlinear Tracking Control for a Hard Disk Drive Dual-Stage Actuator System,” *Transactions on Mechatronics*, 2008.
- [57] Shamma, J. S. and Cloutier, J. R., “Gain-Scheduled Missile Autopilot Design Using Linear Parameter Varying Transformations,” *Journal of Guidance, Control, and Dynamics*, Vol. 16, No. 2, 1993, pp. 256–263.
- [58] Esteban, A. M., *A Linear Parameter Varying Model of The Boeing 747-100/200 Longitudinal Motion*, Ph.D. thesis, University of Minnesota, 2001.
- [59] Kothare, M. V., Mettler, B., Morari, M., Bendotti, P., and Falinower, C.-M., “Linear Parameter Varying Model Predictive Controller For Steam Generator Level Control,” *Computers and Chemical engineering*, Vol. 21, 1997, pp. S861–S866.

- [60] Fialho, I. and Balas, G. J., “Road Adaptive Active Suspension Design Using Linear Parameter-Varying Gain-Scheduling,” *Control Systems Technology, IEEE Transactions on*, Vol. 10, No. 1, 2002, pp. 43–54.
- [61] Warwel, M., Wittler, G., Hirsch, M., and Reuss, H.-C., “Real-Time Coolant Temperature Monitoring in Power Electronics Using Linear Parameter-Varying Models For Variable Coolant Flow Situations,” *Thermal Investigations of ICs and Systems, 2014 20th International Workshop on*, IEEE, 2014, pp. 1–4.
- [62] Hu, J., Wang, Z., Gao, H., and Stergioulas, L. K., “Robust Sliding Mode Control For Discrete Stochastic Systems With Mixed Time Delays, Randomly Occurring Uncertainties, and Randomly Occurring Nonlinearities,” *Industrial Electronics, IEEE Transactions on*, Vol. 59, No. 7, 2012, pp. 3008–3015.
- [63] Marcos, A. and Balas, G. J., “Development of Linear-Parameter-Varying Models For Aircraft,” *Journal of Guidance, Control, and Dynamics*, Vol. 27, No. 2, 2004, pp. 218–228.
- [64] Lu, Q., Zhang, L., Shi, P., and Karimi, H. R., “Control Design For a Hypersonic Aircraft Using a Switched Linear Parameter-Varying System Approach,” *Proceedings of The Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering*, Vol. 227, No. 1, 2013, pp. 85–95.
- [65] Wang, Q. and Stengel, R. F., “Robust Nonlinear Control of a Hypersonic Aircraft,” *Journal of Guidance, Control, and Dynamics*, Vol. 23, No. 4, 2000, pp. 577–585.
- [66] Xu, H., Mirmirani, M. D., and Ioannou, P. A., “Adaptive Sliding Mode Control Design For a Hypersonic Flight Vehicle,” *Journal of Guidance, Control, and Dynamics*, Vol. 27, No. 5, 2004, pp. 829–838.
- [67] Shahruz, S. and Behtash, S., “Design of Controllers For Linear Parameter-Varying Systems by The Gain Scheduling Technique,” *Journal of Mathematical Analysis and Applications*, Vol. 168, No. 1, 1992, pp. 195–217.
- [68] Rugh, W. and Shamma, J., “Research on Gain Scheduling,” *Automatica*, Vol. 36, No. 10, 2000, pp. 1401–1425.

- [69] Stilwell, D. and Rugh, W., “Stability Preserving Interpolation Methods For The Synthesis of Gain Scheduled Controllers,” *Automatica*, Vol. 36, No. 5, 2000, pp. 665–671.
- [70] Reichert, R., “Dynamic Scheduling of Modern-Robust-Control Autopilot Designs For Missiles,” *Control Systems, IEEE*, Vol. 12, No. 5, 1992, pp. 35–42.
- [71] Hyde, R. and Glover, K., “The Application of Scheduled H_∞ Controllers to a VSTOL Aircraft,” *Automatic Control, IEEE Transactions on*, Vol. 38, No. 7, 1993, pp. 1021–1039.
- [72] Cook, J., Grizzle, J., and Sun, J., “Engine Control,” *Control System Applications*, 1996, pp. 87–100.
- [73] Becker, G. and Packard, A., “Robust Performance of Linear Parametrically Varying Systems Using Parametrically-Dependent Linear Feedback,” *Systems and Control Letters*, Vol. 23, No. 3, 1994, pp. 205–215.
- [74] Stilwell, D. and Rugh, W., “Interpolation of Observer State Feedback Controllers For Gain Scheduling,” *Automatic Control, IEEE Transactions on*, Vol. 44, No. 6, 1999, pp. 1225–1229.
- [75] De Oliveira, M., Geromel, J., and Bernussou, J., “Extended H_2 and H_∞ Norm Characterizations and Controller Parametrizations For Discrete-Time Systems,” *International Journal of Control*, Vol. 75, No. 9, 2002, pp. 666–679.
- [76] Apkarian, P. and Adams, R., “Advanced Gain-Scheduling Techniques For Uncertain Systems,” *Control Systems Technology, IEEE Transactions on*, Vol. 6, No. 1, 1998, pp. 21–32.
- [77] Nichols, R., Reichert, R., and Rugh, W., “Gain Scheduling For H_∞ Controllers: A Flight Control Example,” *IEEE Transactions on Control Systems Technology*, Vol. 1, No. 2, 1993, pp. 69–79.
- [78] Caigny, J., Camino, J. F., and Swevers, J., “Interpolation-Based Modeling of MIMO LPV Systems,” *Control Systems Technology, IEEE Transactions on*, Vol. 19, No. 1, 2011, pp. 46–63.
- [79] Caigny, J., Camino, J. F., Oliveira, R., Peres, P., and Swevers, J., “Gain-Scheduled Dynamic Output Feedback Control For Discrete-Time LPV Systems,”

International Journal of Robust and Nonlinear Control, Vol. 22, No. 5, 2012, pp. 535–558.

- [80] Caigny, J., Pintelon, R., Camino, J., and Swevers, J., “Interpolated Modeling of LPV Systems Based on Observability and Controllability,” *Proc. 16th IFAC Symposium on System Identification, Brussels, Belgium*, 2013, pp. 1773–1778.
- [81] Wernow, J., “On The Future of Head-Based Microactuators in Hard Disk Drives,” 2012.
- [82] Du, C., Tan, C. P., and Yang, J., “Three-Stage Control For High Servo Bandwidth and Small Skew Actuation,” *Magnetics, IEEE Transactions on*, Vol. 51, No. 1, 2015, pp. 1–7.
- [83] Felix, S., Nie, J., and Horowitz, R., “Enhanced Vibration Suppression in Hard Disk Drives Using Instrumented Suspensions,” *Magnetics, IEEE Transactions on*, Vol. 45, No. 11, 2009, pp. 5118–5122.

Appendix A

MATLAB Functions

```
% [nOpt, wOpt, Imin, Imax]=optNominalFreq(sys,freqRange);
% Finds the optimal nominal value for a set of frequency
% response data.
% The user is able to choose which frequency range to
% apply the convex
% optimization.
%
% Input:
%     sys: FRD object with set of frequency response
%         data of the system
%
%     freqRange: the frequency range to apply the method
%
% Output:
%     nOpt: the optimal value that will minimize the
%         required uncertainty weight
%
%     wOpt: the optimal uncertainty weight required to
%         cover the data.
%
%     Imin: the minimum frequency index
%
%     Imax: the maximum frequency index

function [nOpt, wOpt, Imin, Imax] = optNominalFreq(sys,
    freqRange);

szH=size(sys);
nY=szH(1);
```

```

nU=szH(2);
nModels=szH(3);

freq = sys.frequency;

if isempty(freqRange);
    minFreq = -1;
    maxFreq = 10^10;
else
    minFreq = freqRange(1);
    maxFreq = freqRange(2);
end

Imin = find(freq > minFreq, 1, 'first');
Imax = find(freq < maxFreq, 1, 'last');

dataFreq = freq(Imin:Imax);
tmp1      = sys.response;
tmp2      = reshape(tmp1, length(freq), nModels);
sysFreq   = tmp2(Imin:Imax, :);

sysData = frd(sysFreq, dataFreq);

nFreq=length(sysData.frequency);
szH=size(sysData);
nU=szH(2);
nModels=szH(3);

Pnom = zeros(nFreq,1);
Lnom = zeros(nFreq,1);
Rnom = zeros(nFreq,1);

setlmis([]);
[L] = lmivar(1,[1 0]);
[R] = lmivar(1,[1 0]);
[Pr] = lmivar(1,[1 0]);
[Pi] = lmivar(1,[1 0]);

for n=1:nModels;
    lmiterm([-n 1 1 L],1,1);
    lmiterm([-n 2 2 R],1,1);
    lmiterm([-n 3 3 L],1,1);
    lmiterm([-n 4 4 R],1,1);

```

```

        lmiterm([n 1 2 Pr],1,1);
        lmiterm([-n 1 4 Pi],1,1);
        lmiterm([n 2 3 Pi],1,1);
        lmiterm([n 3 4 Pr],1,1);
    end
    lmiterm([-nModels+1 1 1 L],1,1);
    lmiterm([-nModels+2 1 1 R],1,1);
    lmisTemplate=getlmis;

    FD = sysData.ResponseData;
    for m=1:nFreq;
        setlmis(lmisTemplate);
        for n=1:nModels;
            H = FD(1,1,m,n);
            HR = real(H);
            HI = imag(H);
            lmiterm([-n 1 2 0],HR);
            lmiterm([n 1 4 0],HI);
            lmiterm([-n 2 3 0],HI');
            lmiterm([-n 3 4 0],HR);
        end
        lmis = getlmis;
        c=[1 1 0 0];
        options = [0 0 0 0 1];
        [~,xopt] = mincx(lmis,c, options);
        Lnom(m)=xopt(1);
        Rnom(m)=xopt(2);
        P=xopt(3)+sqrt(-1)*xopt(4);
        Pnom(m)=P;
    end

    Lw=Lnom;
    Rw=Rnom;

    n0pt=frd(Pnom,sysData.Frequency);
    w0pt=frd(Lw,sysData.Frequency)/n0pt;

    end

```

```

% [W,info]=FrdTrace(Data,ord,varargin);
% This function creates a FRD object based on the user
% input of frequency points and the the magnitude (in dB)
% at those points.
%
% Points between the frequency points are interpolated.
% Then fitmagfrdIter function is used to fit a
% transfer function model to those points
%
% Input:
%     Data: a matrix that defines the frequencies and
%           magnitude of the weight, has the form
%           [freq(rad) mag(dB); freq(rad) mag(dB);... ]
%           Connects the dot between points
%
%     ord: order of the weight
%
%     varargin(Wt): can add weight to the fitmagfrd fit.
%
% Output:
%     W: The weight model created from the fit
%
%     info: Contains various data such as raw freq
%           and magnitude data.

function [W,info]=FrdTrace(Data,ord,varargin);
N=length(Data(:,1));
M=40;
InitFreq=Data(:,1);
InitDB=Data(:,2);

freq=zeros(1,(N-1)*M);
magDB=freq;

for i=1:N-1
    F1Log=log10(InitFreq(i));
    F2Log=log10(InitFreq(i+1)*.99);

    freqtmp=logspace(F1Log,F2Log,M);
    freq(1+(i-1)*M:M*i)=freqtmp;
    slptmp=(InitDB(i+1)-InitDB(i))/ ...
        (log10(InitFreq(i+1)/InitFreq(i)));
    if slptmp==0;

```

```

        b=InitDB(i);
        magtmpDB=ones(1,M)*InitDB(i);
    else
        b=InitFreq(i)/(10.^(InitDB(i)/slptmp));
        magtmpDB=slptmp*log10(freqtmp./b);
    end
    magDB(1+(i-1)*M:M*i)=magtmpDB;
end

mag=10.^(magDB/20);

info.f=freq;
info.m=mag;

wFrd=frd(mag,freq);

info.wFrd=wFrd;

if ~isempty(varargin)
    wt=varargin{1};
    [W,info.sysIter]=LOCALfitmagIter(wFrd,ord,'low',wt);
else
    [W,info.sysIter]=LOCALfitmagIter(wFrd,ord,'low');
end
end

% This function uses fitmagfrd in iteration to fit a ss
% model to have a magnitude greater than or less than
% that of the FRD object.
%
% Method (for fitting a model with greater magnitude than\
% the FRD): Fits a 1st order model with magnitude
% greater than the FRD object. Then it will fit a
% second order model to have magnitude less than the
% 1st order model with 20% increase in magnitude and
% greater than the FRD object. Then it will repeat for
% N number of times until the desired order for the
% model is reached.
%
% Input:
%     sysfrd: The FRD object to fit a ss model.
%
%     order: Desired number of orders for the model.

```

```

%
%     opt: 'high' fit a model with magnitude greater
%           than the FRD's
%           'low' fit a model with magnitude less than
%           that FRD's
%
%     varargin: If there is a desire to weight the model
%                fit, this is the input to put it in.
% Output:
%     sys: The output with number of order specified in
%           the input.
%     sysIter: A cell with collection of all models that
%              were fit onto the FRD
function [sys,sysIter]=LOCALfitmagIter(sysfrd,order,opt,
    varargin)

if ~isempty(varargin)
    wt=varargin{1};
else
    wt=[];
end
freq=sysfrd.frequency;

sysIter=cell(1,order);
switch opt
    case 'high'
        sysIter{1}=fitmagfrd(sysfrd,1,[],wt,1);
        for i=2:1:order;
            sysg=abs(frd(sysIter{i-1},freq));
            upperFrd=1.1*sysg;
            C2.LowerBound=abs(sysfrd);
            C2.UpperBound=upperFrd;
            sysIter{i}=fitmagfrd(sysfrd,i,[],wt,C2);
        end

    case 'low'
        sysIter{1}=fitmagfrd(sysfrd,1,[],wt,-1);
        for i=2:1:order;
            sysg=abs(frd(sysIter{i-1},freq));
            lowerFrd=0.8*sysg;
            C2.UpperBound=abs(sysfrd);
            C2.LowerBound=lowerFrd;
            sysIter{i}=fitmagfrd(sysfrd,i,[],wt,C2);
        end
end

```

```
        end
    end
    sys=sysIter{end};

end
```