Robust Controller Design Using H_∞-Loop Shaping Approach

In partial fulfillment of the requirements
for the degree of

BACHELOR OF TECHNOLOGY

by

AYUSH PANDEY

Department of Electrical Engineering
Indian Institute of Technology, Kharagpur
West Bengal, INDIA 721302
May 2016

Department of Electrical Engineering
Indian Institute of Technology, Kharagpur
West Bengal, INDIA 721302
November 2015

Certificate

This is to certify that the report entitled **Robust Controller Design Using H_∞ Loop Shaping Approach**, submitted by **Ayush Pandey**, an undergraduate student, in the *Department of Electrical Engineering, Indian Institute of Technology, Kharagpur, India,* for the award of the degree of Bachelor of Technology, is a record of the work carried out by him under our supervision and guidance. Neither this report nor any part of it has been submitted for any degree or academic award elsewhere.

Dr. Sourav Patra
Assistant Professor,
Department of Electrical Engineering,
Indian Institute of Technology, Kharagpur,
INDIA 721302

Acknowledgements

I would like to take this opportunity to thank and acknowledge the help and support of the people who have been directly or otherwise involved with the work presented in thesis.
I am humbled by the support of my advisor Professor Sourav Patra. It his only because of his guidance and continuous support that I have been able to work persisently on this project.
I also wish to thank the Professors who have evaluated my project progress from time to time. The reviews and the evaluation certainly motivated me to work even harder on the project. I also owe my gratitude to Ganesh who helped me during the very beginning of this project by properly presenting a layout for me to go ahead with this project.
Also, I can't imagine having survived through the tough times without the support of fellow classmates and friends. Finally, I would like to thank my family. Without their love and support, none of this would have been possible.

Ayush Pandey
IIT Kharagpur

Contents

1 Introduction

Robustness of a system is defined by its ability to be insensitive to parameter variations and unwanted exogenous signals. For a control system, these variations could either be plant disturbances and parameter variations within the plant or these could be due to the sensor noises in measurement. Also, at times it is not possible to model the system dynamics accurately. These unmodeled plant dynamics may also affect the system in undersired manner.(See the block diagram in Figure(10)).

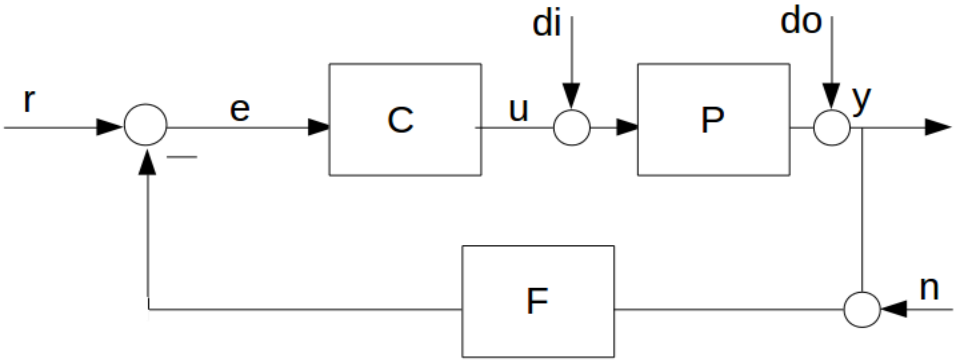

Figure 1: Control System Block Diagram
n-Sensor Noise, r-Reference Input, y-Outpt, P-Plant, C-Controller, F-Feedback, di-Input Disturbance, do-Output Disturbance, e-Error, u-Control Input

Hence, the aim of a control engineer is to design a controller in such a way that it deals with these uncertainities. The classical control theories, though, provide good performance for SISO systems but they are, in general, not replicable to multivariable systems. To overcome these difficulties and to achieve a robust performance in MIMO systems several successful design methods have been developed in the past three decades. Among them, the H_∞ control is a popular and effective robust control design technique.
The theory of robust control was first proposed in the late 1970s and early 1980s and since then there has been a huge effort in various methods to achieve specified performance of a system robustly. The most widespread and important of these methods is the H_∞ loop-shaping technique which was proposed by McFarlane and Glover of Cambridge University. The H_∞ loop-shaping technique combines classical loop-shaping concepts with the H_∞ controller design techniques.
For application of a robust controller to a particular application the constraints of the system need to be modeled as well and the controller design

has to be changed accordingly. This project aims to study in detail the H_∞ loop-shaping robust controller design technique and then model a mobile robot system in such a way that a robust controller for the mobile robot's locomotion is designed. Some challenges that might be faced in this process are those concerning the complexity of the very theory of H_∞ loop-shaping technique. Moreover, designing the controller for a mobile robot would be faced with many challenges with respect to actuator saturation and other modeling constraints.

2 Background and Motivation

The robust control theory is covered in great detail in the book by authors Kemin Zhou, Keith Glover and J C Doyle in "Robust and Optimal Control" [1]. The book was published in 1995 and hence it covers most of the research and contributions by various pioneers to modern control theory. H_∞ methods in control theory are used to design controllers which guarantee specific performance with robust stability. The H_∞ control problem is expressed as an optimization problem. Loop shaping control technique is a part of classical control that has been existent in control theory since a long time. Hence, the amalgamation of the two, known as H_∞ loop-shaping design method combines the two methods and achieves a good robust performance and stability.

The progress in robust control design techniques has been enormous and has even been applied to the industry to an aircraft. The 1995 publication breifs about the same [2]. Despite the popularity, the advantages of a robust controller are such that in future most of the controllers would be preferred to be designed using a robust algorithm and hence there remains a great scope of learning and research in this field. With regards to application of robust control theory, Gu, Petkov and Konstantinov published a book titled "Robust Control Design with MATLAB" [3]. The book elucidates the robust control design procedure using MATLAB and considers various multivariable systems and their controller designs as case studies. The second edition of this book demonstrates the use of the latest Robust Control Toolbox v3.0.

The motivation for this project comes from the fact that usually great amount of experimental, hit and trial based methods are employed for controlling a mobile robot. In this sense, if a robust control system can be designed it would provide great ease for further development of technology on robots. Robotics is finding it's way into our lives at a pace which was never expected and is more than ever before. The applications range from big industries to small household chores and beyond. In this changing

world, the control of such a mobile robot is essential and important to be achieved in a very robust manner so that more and more complex applications can be designed on top of it.

Particularly, on working with a tricycle design mobile robot driven by a BLDC hub motor for translation, we found that the PID controller being used previously for speed control was susceptible to parameter variations and the performance when the road terrain is not smooth is extremely poor.

The PID controller needs to be tuned experimentally time and again depending on the conditions but still the controller is not robust as shown in the input output responses below.

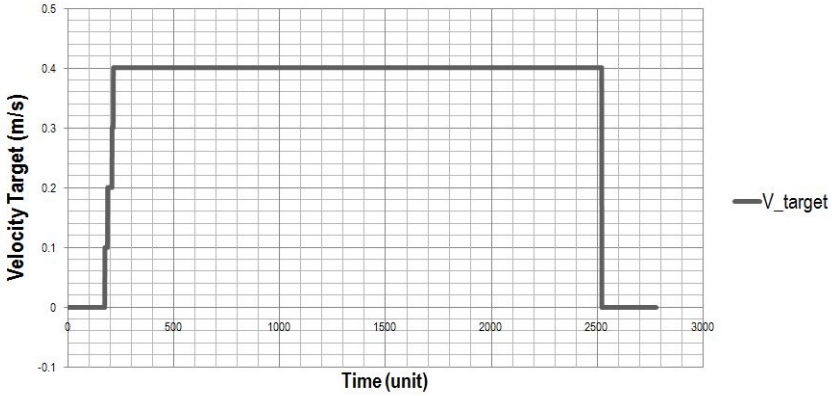

Figure 2: Step reference input given to the robot

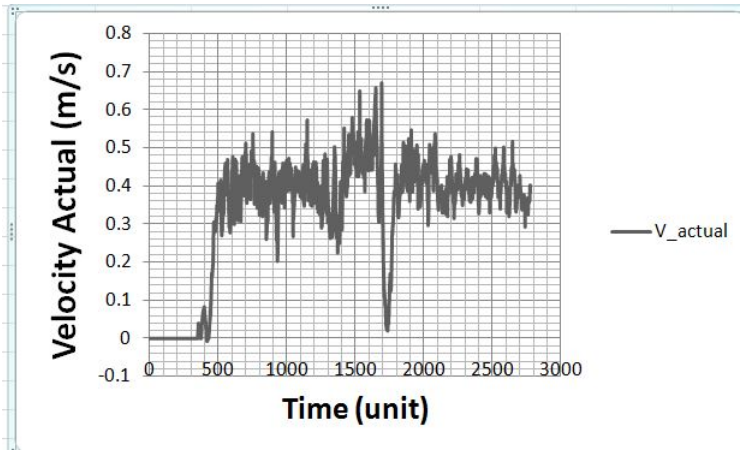

Figure 3: Output step response with PID controller

As shown in Fig.(2), a step input for speed was given to the robot. The output speed response because of the uneven terrain of the road and because of the parameter variations is shown in Fig.(3). A sudden dip in

the output response plot is because of a small pothole that the robot ran into on the road, it took almost 4 to 5 seconds for the robot to regain its reference steady state value after hitting that pothole. It is clear that a robust controller is needed here to improve the performance of this system and also to do away with the need of tuning the controller everytime with changing parameters/environmental conditions.

3 Preliminaries

As explained in Section(2), the robust controller design revloves around solving an optimization problem and hence it is obvious that various mathematical techniques need to be learnt and applied to successfully be able to learn the theory of H_∞ control technique. The following subsections go over the basics required as prerequisites for all the study on robust control that shall follow. The mathematical tools and concepts given in this section have been limited to an introduction only and proper references are given for the readers who wish to dwelve more into a particular concept.

3.1 Vector Norms and Normed Spaces

In linear algebra, norm is a function that is defined for a vector which gives a positive scalar quantity representing the vector's size. It can be understood as an analogy to the absolute value function that we have for scalar quantities. A vector space for which a particular norm is defined is known as a normed space.

3.1.1 Types of norms

p-Norm The p-norm is the most commonly used form of norm used in linear algebra. For a vector x$\in \mathbb{R}$, the p-norm is given by

$$\|x\|_p := \left(\sum_{i=1}^{n} (x_i)^p \right)^{\frac{1}{p}} \tag{1}$$

p can take any value in [1,∞) which would give different norms such as the 2-norm and so on. The two norm is also known as the Euclidean norm or the energy norm.

3.2 Inner product and Inner Product Spaces

The inner product between two vectors x and y is denoted by $\langle x, y \rangle$ and is given by x^*y where the * operator is the adjoint operator which can be

calculated by taking the transpose and then the conjugate of the complex vector.

$$\langle x, y \rangle = x^* y = \sum_{i=1}^{n} (\bar{x_i} y_i)$$

If for a vector space V, the inner product is defined in the manner as in eq. 3.2 and it exists. Then the vector space V is called an inner product space.

3.3 Hilbert Spaces

A Hilbert space is a vector space which is a complete inner product space with the norm induced by its inner product.

3.3.1 Examples

There can be many common examples for a Hilbert Space. Some of them are

1. The space $\mathbb{R}^n$, with the usual inner product defined is a finite dimensional Hilbert Space.

2. The space $\mathbb{R}^{n \times m}$ of matrix valued functions is a Hilbert Space with the inner product defined as:

$$\langle A, B \rangle := Trace A^* B = \sum_{i=1}^{n} \sum_{j=1}^{m} \bar{a_{ij}} b_{ij} \quad \forall A, B \in \mathbb{C}^{n \times m}$$

3. $l_2(-\infty, \infty)$ is an infinite dimensional Hilbert Space which consists of sequences ...., $x_{-2}, x_{-1}, x_0, x_1, ...$ (real or complex) which are square summable. The inner product is given by:

$$\langle x, y \rangle := \sum_{i=-\infty}^{\infty} \bar{x_i} y_i$$

If each component is a vector or a matrix, then the inner product shall be given by

$$\langle x, y \rangle := \sum_{i=-\infty}^{\infty} Trace\,(x_i^* y_i)$$

4. Subspaces of l_2 viz. $l_2[0, \infty], l_2(-\infty, 0]$ are defined similarly.

3.4 Hardy Spaces

A function f(t) is said to be analytic at a point t_0 if it is continuous at that point and in the neighbourhood along with being differentiable at t_0. For analytic functions over a set S, the derivatives of all orders at all points in S of the function f(t) exist. This also implies that the function will have a power series at all those points where it is analytic. The opposite of this is also true, that is, when the power series exists for a function at a point then it is differentiable and analytic at the given point as well. If a function is a matrix valued function then all elements of the matrix should be analytic at the considered point or at all points in the considered set, whatever may be the case.

$\mathbb{L}_2(j\mathbb{R})$ space can also be written simply as $\mathbb{L}_2$ space is a Hilbert space of matrix valued functions on j$\mathbb{R}$ and consists of all matrix valued functions for which the inner product is defined as follows

$$\langle F, G\rangle := \frac{1}{2\pi}\int_{-\infty}^{\infty} Trace\left[F^*(j\omega)\, G(j\omega)\right]d\omega \tag{2}$$

and the inner product induced norm is given by

$$\|F\|_2 := \sqrt{\langle F, F\rangle} \tag{3}$$

Now, if for a given transfer matrix F (real, rational, strictly proper), if there are no poles on the $j\omega$ axis, then we have a subspace of $\mathbb{L}_2$ space known as $\mathbb{RL}_2$.

Hardy Spaces are subspaces of $\mathbb{L}_2(j\mathbb{R})$ space but with an additional property of analyticity of the functions. Formally, all matrix valued functions analytic in $Re(s) > 0$ form the $\mathbb{H}_2$ space. The inner product can be shown to be equal to the one shown in Eq.(2). On similar lines as to $\mathbb{RL}_2$, we define $\mathbb{RH}_2$ which is a subspace of $\mathbb{H}_2$. The $\mathbb{RH}_2$ consists of all real rational and stable (all poles in LHP) transfer matrices. This is because if there were poles in the RHP then the functions would not be analytic anymore.
The space $\mathbb{H}_2^\perp$ is the subspace of $\mathbb{L}_2$ having analytic functions in LHP, which implies $\mathbb{RH}_2^\perp$ would contain all proper rational transfer matrices having all poles in the RHP.
Next, we have $\mathbb{H}_\infty$ spaces. The above definitions follow in the same way for $\mathbb{L}_\infty(j\mathbb{R})$, $\mathbb{H}_\infty$ and $\mathbb{H}_\infty^-$ spaces and the $\mathbb{RL}_\infty$, $\mathbb{RH}_\infty$, $\mathbb{RH}_\infty^-$ subspaces

respectively except that now the norms are given by the following set of equations,

For $\mathbb{L}_\infty(j\mathbb{R})$ space

$$\|F\|_\infty := ess \sup_{\omega \in \mathbb{R}} \bar{\sigma}[F(j\omega)]$$

For $\mathbb{H}_\infty$ space

$$\|F\|_\infty := \sup_{Re(s)>0} \bar{\sigma}[F(s)] = ess \sup_{\omega \in \mathbb{R}} \bar{\sigma}[F(j\omega)]$$

The second equality can be regarded as the generalization of maximum modulus theorem for matrix functions. Now for $\mathbb{H}_\infty^-$ space

$$\|F\|_\infty := \sup_{Re(s)<0} \bar{\sigma}[F(s)] = ess \sup_{\omega \in \mathbb{R}} \bar{\sigma}[F(j\omega)]$$

where $\bar{\sigma}$ is equal to maximum singular value of the matrix, where a singular value of a matrix T is given by $\sqrt{\lambda_i(T^*T)}$ for ith eigen value of the matrix.

3.5 Well Posedness and Internal Stability

A feedback system is said to be well posed if all closed-loop transfer function matrices are well-defined and proper.
For a linear time invariant system given by a transfer function $G(s)$, stability of the system is assured if and only if all poles of the the transfer function lie in the open left half complex plane. This is also often referred to as the BIBO stability condition, i.e. when the input is bounded the output will be bounded. For a system described by state space representation, the concept of asymptotic stability is defined. A system is asymptotically stable if, for an identically zero input, the system states will converge to zero from any initial states. Hence, an interconnected system is internally stable if the subsystems of all input–output pairs are asymptotically stable.

3.6 Coprime factorization over $\mathbb{RH}_\infty$

For a SISO linear system whose transfer function is given $b(s)/a(s)$, $b(s)$ and $a(s)$ are said to be coprime if they do not have any common zeros. Such a representation of the system is often referred to as the minimal realization of the system. For SISO systems, it is always possible to write a transfer function as a ratio of two stable transfer function with no common divisors (i.e. their greatest common divisor (gcd) is 1). This is called coprime factorization of the transfer function as the two factored transfer functions

have no common divisors. Mathematically, using Euclid's algorithm we can write transfer functions X(s) and Y(s) in $\mathbb{RH}_\infty$ such that for a given transfer function G(s), transfer fucntions M(s) and N(s) in $\mathbb{RH}_\infty$ are it's coprime factors such that:

$$G(s) = N(s)M(s)^{-1} \tag{4}$$

where X and Y exists such that

$$N(s)X(s) + M(s)Y(s) = 1 \tag{5}$$

Eq.(5) is known as the Bezout's Identity. With some variations, the above is valid for MIMO systems as well. But, for MIMO systems both left and right coprime factorization need to be defined appropriately. In Chapter 5 of [1], detailed explainations and equations for the same have been given.

4 Linear Fractional Transformation

Linear Fractional Transformation or LFT is a unified theory of transfer function and state space representations including uncertainity. All H_∞ control problems use LFT representation of systems. An LFT representation could be of two types, upper and lower LFTs. Both of these structures are shown in the Figures (4) and (5).

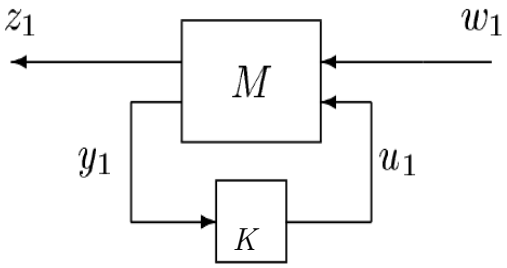

Figure 4: Lower LFT Structure

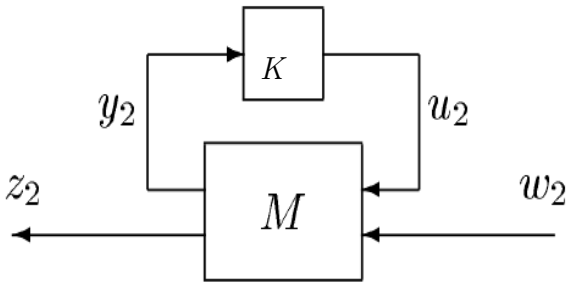

Figure 5: Upper LFT Structure

For the lower LFT structure from Figure(4) the transfer matrix from w to z is denoted by $\mathscr{F}_l(G, K)$. Now for this structure since we have,

$$\begin{bmatrix} z \\ y \end{bmatrix} = \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix} \begin{bmatrix} w \\ u \end{bmatrix}$$

we can easily work out (using $u = Ky$) that the transfer matrix T_{zw} is given by $G_{11} + G_{12}K(I - G_{22}K)^{-1}G_{21}$ which is denoted by $\mathscr{F}_l(G, K)$. Similarly for upper LFT one can write from Figure(5) the following equations:

$$\begin{bmatrix} y \\ z \end{bmatrix} = \begin{bmatrix} G_{11} & G_{12} \\ G_{21} & G_{22} \end{bmatrix} \begin{bmatrix} u \\ w \end{bmatrix}$$

The transfer function comes out as follows (the notation changes as well): $T_{zw} = \mathscr{F}_u(G, K) = G_{22} + G_{21}K(I - G_{11}K)^{-1}G_{12}$. Here the vectors z, w, u, y have special meaning as follows:

1. z: The objective signal vector. All signals which are to be penalized (or are of interest, performance wise) are kept in this vector.

2. w: The exogenous input vector. As the name suggests, all input exogenous signals (be it reference signal or disturbance) are included in this vector.

3. u: This is the signal which is given as output by the controller, called the "control signal".

4. y: The measured variables are a part of this vector.

4.1 Basic Principle

LFT framework is widely used for H_∞ control because it provides a general representation for all kinds of systems and systems with different uncertainities as well. This is one of the reasons why LFT forms the basic

working framework for robust controller synthesis and analysis. The uncertainities in a system in an LFT framework are separated out and the resulting framework, called as an M-Δ structure consists of the nominal plant M (without any uncertainities) and Δ matrix has all uncertain parameters. A closed loop system with uncertainities can hence be reduced in LFT framework as shown in Figure(6).

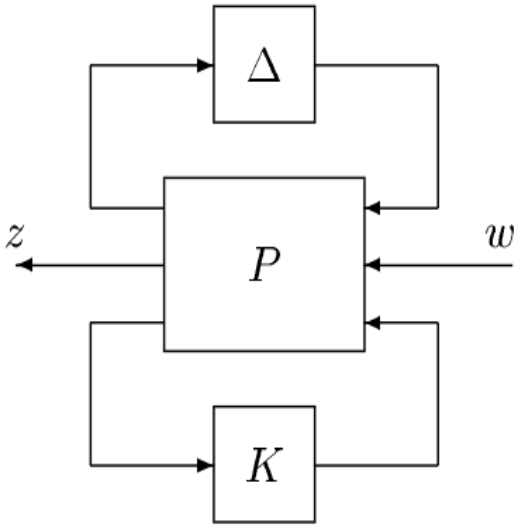

Figure 6: Closed Loop System in LFT Structure

The input output relation for the system can easily be worked out by considering the upper and lower LFT structures in the given system.
For any given system, to reduce in this M-Δ format, all uncertain parameters need to be "pulled out" of the given plant. This can be achieved after drawing the block diagram of the system such that all uncertain parameters are individual blocks in the block diagram. This technique is illustrated in detail in the following example of a spring mass damper system.

4.2 Spring Mass Damper System : LFT Configuration

The Figure(7) shows a simple spring mass damper system with external force F being applied on the mass.

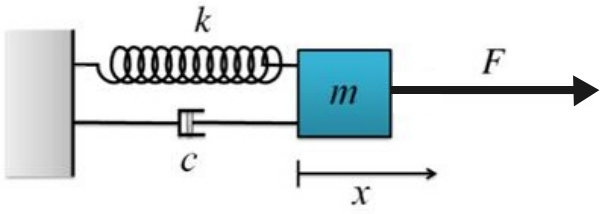

Figure 7: Spring Mass Damper System

The system parameters m (mass of the block), c (the damping constant) and k (spring constant) are uncertain parameters with given upper and lower bound on uncertainities. The equation of motion is given by :

$$\ddot{x} + \frac{c}{m}\dot{x} + \frac{k}{m}x = \frac{F}{m} \tag{6}$$

A block diagram with two integrators and parametric uncertainities can be easily drawn, as shown in Figure(8), from the above equation using the uncertainity values within 10% of mass $\tilde{m}$, within 20% of $\tilde{c}$ and within 30% for $\tilde{k}$. The corresponding uncertain parameters are given by δ_m, δ_c and δ_k respectively. Note here that $\tilde{x}$ represents the nominal value of the parameter, where x is the uncertain parameter.

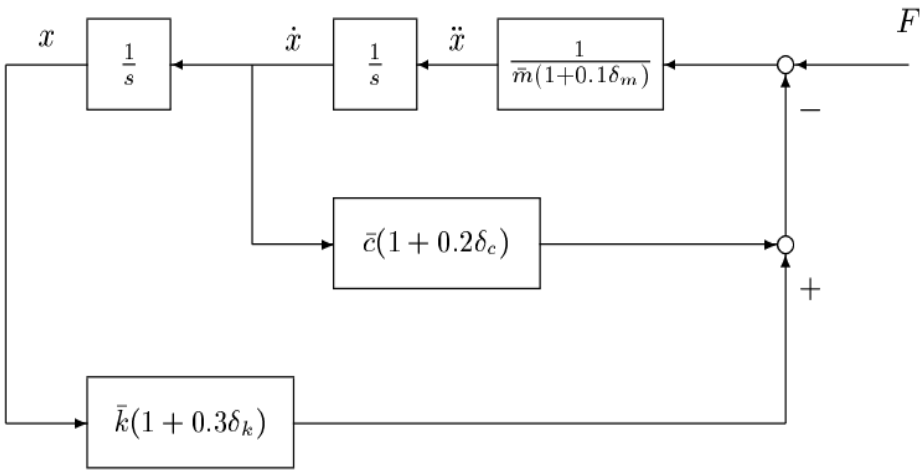

Figure 8: Spring Mass Damper Block Diagram with Uncertainities

The three uncertain parameters now need to be represented in LFT framework and for that we need to separate out the Δ's. To achieve this,

we first evaluate the $\frac{1}{m}$ block in the given block diagram. The $\frac{1}{\tilde{m}}$ block can be represented as LFT by the following manipulation:

$$\frac{1}{m} = \frac{1}{\tilde{m}(1 + 0.1\delta_m)} \tag{7}$$

on evaluating, we get

$$\frac{1}{m} = \frac{1}{\tilde{m}} - \frac{0.1\delta_m}{\tilde{m}}.(1 + 0.1\delta_m)^{-1} \tag{8}$$

The above equation can be represented in lower LFT framework where, the nominal plant M is given by $\mathscr{F}_l(M, \Delta_m)$ and $\Delta = \delta_m$.

$$M = \begin{bmatrix} \frac{1}{\tilde{m}} & \frac{-0.1}{\tilde{m}} \\ 1 & -0.1 \end{bmatrix}$$

Next, to separate out δ_c and δ_k, simple block diagram manipulations lead us to the following block diagram shown in Figure(9).

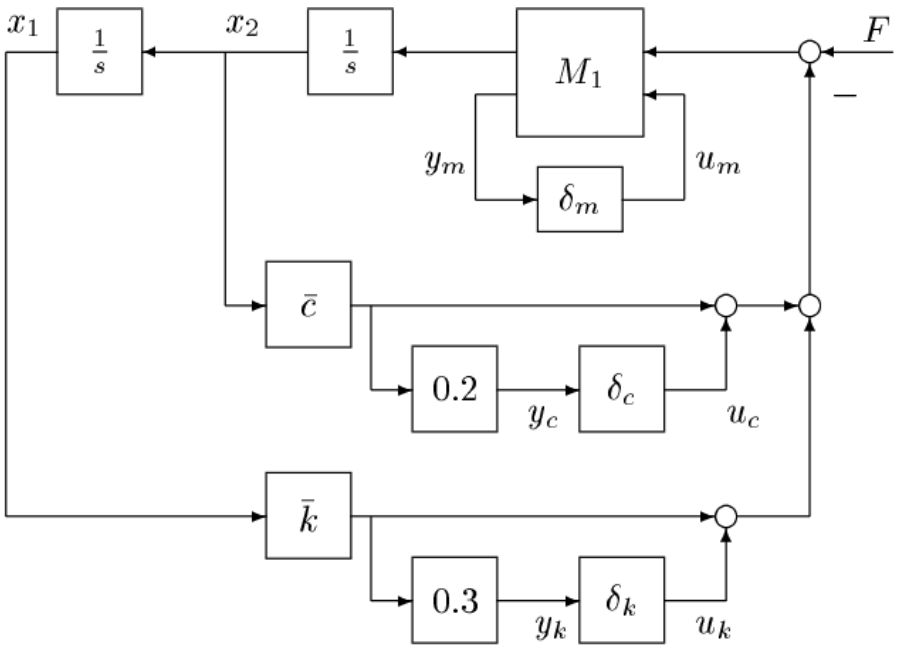

Figure 9: Spring Mass Damper Block Diagram with Uncertainities Pulled Out

Now, we can mark the inputs and outputs to these δs appropiately. y is the input to the delta and u is its output. Using these inputs and outputs

to δs along with exogenous input vector and the objective signal vector we can represent the whole spring mass damper system into M-Δ structure. The values of M and Δ can be found out by actually writing out the state space representation using the block diagram. For this system the exogenous signal vector can be chosen to comprise of F, x_1 and x_2 and the objective signal vector as $\dot{x_2}$ and $\dot{x_1}$, where x_1 and x_2 are the states of the system. The final M-Δ structure in LFT form is given as follows:

$$\left[\begin{array}{c} \dot{x}_1 \\ \dot{x}_2 \\ \hline y_k \\ y_c \\ y_m \end{array}\right] = \left[\begin{array}{ccc|ccc} 0 & 1 & 0 & 0 & 0 & 0 \\ \frac{-\tilde{k}}{\tilde{m}} & \frac{-\tilde{c}}{\tilde{m}} & \frac{1}{\tilde{m}} & \frac{-1}{\tilde{m}} & \frac{-1}{\tilde{m}} & \frac{-0.1}{\tilde{m}} \\ \hline 0.3\tilde{k} & 0 & 0 & 0 & 0 & 0 \\ 0 & 0.2\tilde{c} & 0 & 0 & 0 & 0 \\ -\tilde{k} & -\tilde{c} & 1 & -1 & -1 & -0.1 \end{array}\right] \left[\begin{array}{c} x_1 \\ x_2 \\ F \\ \hline u_k \\ u_c \\ u_m \end{array}\right]$$

,

$$\begin{bmatrix} u_k \\ u_c \\ u_m \end{bmatrix} = \Delta \begin{bmatrix} y_k \\ y_c \\ y_m \end{bmatrix}$$

. Hence, the uncertainity in parameters is all put together in the matrix Δ which is given by:

$$\Delta = \begin{bmatrix} \delta_k & 0 & 0 \\ 0 & \delta_c & 0 \\ 0 & 0 & \delta_m \end{bmatrix}$$

4.3 Generalized SISO Control System Block Diagram

A simple SISO system consisting of a reference signal, controller, disturbance and a plant. The block diagram is shown in Figure (10).

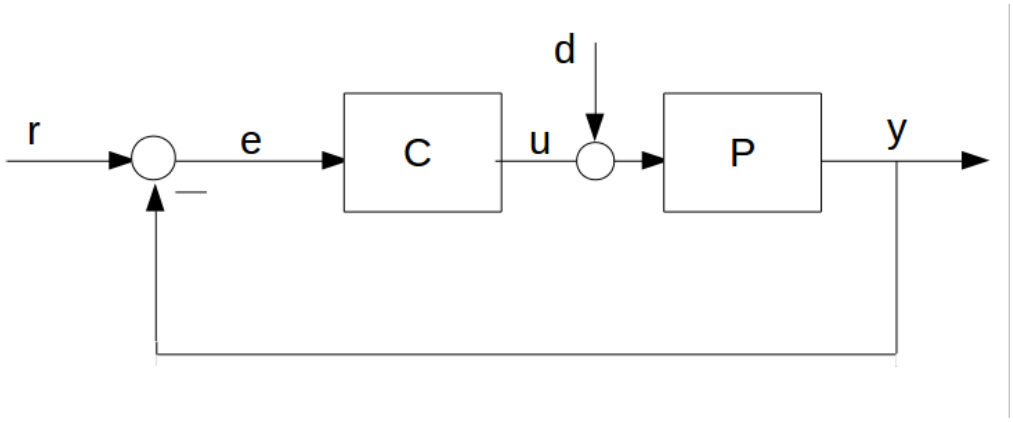

Figure 10: A SISO Control System Block Diagram

The exogenous signal vector (w) for this system will have r and d as its elements. The objective signal (if good tracking of reference is desired) can be chosen as e. The input to the controller C is equal to e, hence $y = e$. The control input in this case is given as u. Now, we can write the transfer function matrix form of the system in LFT framework.

$$e = r - y \tag{9}$$
$$e = r - Pd - Pu \tag{10}$$

We can write,

$$\left[\begin{array}{c} e \\ \hline e \end{array}\right] = \left[\begin{array}{cc|c} I & -P & -P \\ \hline I & -P & -P \end{array}\right] \left[\begin{array}{c} r \\ d \\ \hline u \end{array}\right]$$

Hence, we can write $z = \mathscr{F}_l(G, C)w$ where

$$G_{11} = \begin{bmatrix} I & -P \end{bmatrix}, G_{12} = [-P]$$
$$G_{21} = \begin{bmatrix} I & -P \end{bmatrix}, G_{22} = [-P]$$

The state space representation can also be written by following a similar procedure. The following equations describe the state variable model. It is assumed that the state space representation of the plant P and the controller C are known and their states are denoted as x_p and x_c.

$$\dot{x}_p = A_p x_p + B_p d + B_p u \tag{11}$$
$$y = C_p x_p + D_p d + D_p u \tag{12}$$
$$\dot{x}_c = A_c x_c + B_c r - B_c y \tag{13}$$
$$u = C_c x_c + D_c r - D_c y \tag{14}$$

On solving using the block diagram, we finally get,

$$\left[\begin{array}{c} \dot{x}_p \\ \dot{x}_c \\ \hline e \\ \hline e \end{array}\right] = \left[\begin{array}{cc|cc|c} A_p & 0 & 0 & B_p & B_p \\ -B_c C_p & A_c & B_c & -B_c D_p & -B_c D_p \\ \hline -C_p & 0 & I & -D_p & -D_p \\ \hline -C_p & 0 & I & -D_p & -D_p \end{array}\right] \left[\begin{array}{c} x_p \\ x_c \\ \hline r \\ d \\ \hline u \end{array}\right]$$

Hence, the generalised system matrices are

$$A = \begin{bmatrix} A_p & 0 \\ -B_c C_p & A_c \end{bmatrix}, B_1 = \begin{bmatrix} 0 & B_p \\ B_c & B_c D_p \end{bmatrix}, B_2 = \begin{bmatrix} B_p \\ -B_c D_p \end{bmatrix}$$

$$C_1 = \begin{bmatrix} -C_p & 0 \end{bmatrix}, C_2 = \begin{bmatrix} -C_p & 0 \end{bmatrix}$$

$$D_{11} = \begin{bmatrix} I & -D_p \end{bmatrix}, D_{12} = \begin{bmatrix} -D_p \end{bmatrix}, D_{21} = \begin{bmatrix} I & -D_p \end{bmatrix}, D_{22} = \begin{bmatrix} -D_p \end{bmatrix}$$

4.4 Other Worked Out Examples

Since LFT plays such an important role in H_∞ control design and analysis, some more examples were worked out as a part of this project. The following examples deal with different situations such as presence of sensor noise in measurement in a SISO control system, presence of exogenous output and input disturbances in the system simultaneously along with sensor noise (this is the most general case for SISO systems) and assigning weights to objective signals in multivariable systems (in presence of disturbance and nosie signals).
The block diagram shown in Figure (11) can be reduced in LFT framework in both state space and transfer function forms.

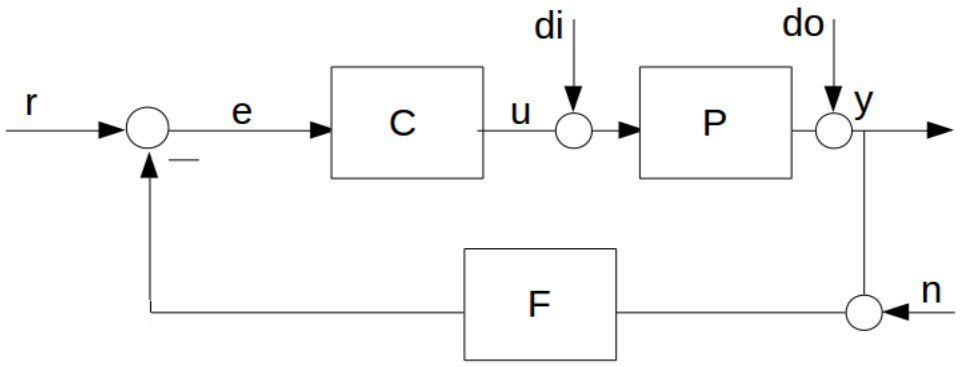

Figure 11: A SISO Control System Block Diagram with Noise and Disturbances

The exogenous input signal vector (w) for this case is given by:

$$w = \begin{bmatrix} r \\ d_i \\ d_o \\ n \end{bmatrix}$$

The objective signal vector z can be taken as equal to e for robust tracking requirement. The input to the controller C is taken as e and the control input is u. The transfer function form is given by:

$$\begin{bmatrix} e \\ \hline e \end{bmatrix} = \left[\begin{array}{cccc|c} I & -FP & -F & -F & -FP \\ \hline I & -FP & -F & -F & -FP \end{array}\right] \begin{bmatrix} r \\ d_i \\ d_o \\ n \\ \hline u \end{bmatrix}$$

Hence, z=$\mathscr{F}_l(G,C)w$ is the lower LFT representation of the system in transfer function form. For state space form, similar to the previous example we can write the state space representation of the system. The final result is given below:

$$\left[\begin{array}{c} \dot{x}_p \\ \dot{x}_c \\ \dot{x}_f \\ \hline e \\ \hline e \end{array}\right] = \left[\begin{array}{ccc|c|ccc|c} A_p & 0 & 0 & 0 & B_p & 0 & 0 & B_p \\ -B_cB_fC_p & A_c & -B_cC_f & B_c & -B_cD_fD_p & -B_cD_f & -B_cD_f & -B_cD_fD_p \\ B_fC_p & 0 & A_f & 0 & B_fD_p & B_f & B_f & B_fD_p \\ \hline -D_fC_p & 0 & -C_f & I & -D_fD_p & -D_f & -D_f & -D_fD_p \\ \hline -D_fC_p & 0 & -C_f & I & -D_fD_p & -D_f & -D_f & -D_fD_p \end{array}\right] \left[\begin{array}{c} x_p \\ x_e \\ x_f \\ \hline r \\ d_i \\ d_o \\ n \\ \hline u \end{array}\right]$$

Hence, the final system matrices are

$$A = \begin{bmatrix} A_p & 0 & 0 \\ -B_cB_fC_p & A_c & -B_cC_f \\ B_fC_p & 0 & A_f \end{bmatrix}$$

$$B_1 = \begin{bmatrix} 0 & B_p & 0 & 0 \\ B_c & -B_cD_fD_p & -B_cD_f & -B_cD_f \\ 0 & B_fD_p & B_f & B_f \end{bmatrix}, B_2 = \begin{bmatrix} B_p \\ -B_cD_fD_p \\ B_fD_p \end{bmatrix}$$

$$C_1 = \begin{bmatrix} -D_fC_p & 0 & -C_f \end{bmatrix}, C_2 = \begin{bmatrix} -D_fC_p & 0 & -C_f \end{bmatrix}$$

$$D_{11} = \begin{bmatrix} I & -D_fD_p & -D_f & -D_f \end{bmatrix}, D_{12} = \begin{bmatrix} -D_fD_p \end{bmatrix}$$

$$D_{21} = \begin{bmatrix} I & -D_fD_p & -D_f & -D_f \end{bmatrix}, D_{22} = \begin{bmatrix} -D_fD_p \end{bmatrix}$$

As one last example, we take a multivariable system with weights W_1 and W_2 assigned to the control input and the objective signal vectors. These weights help in better formulation of cost function in the H_∞ optimization problem. Assigning weight to control input u helps is achieving the performance criteria of having less control energy along with achieving the required performance by assigning weight to the objective signal. The block diagram for the system is shown in Figure(12).

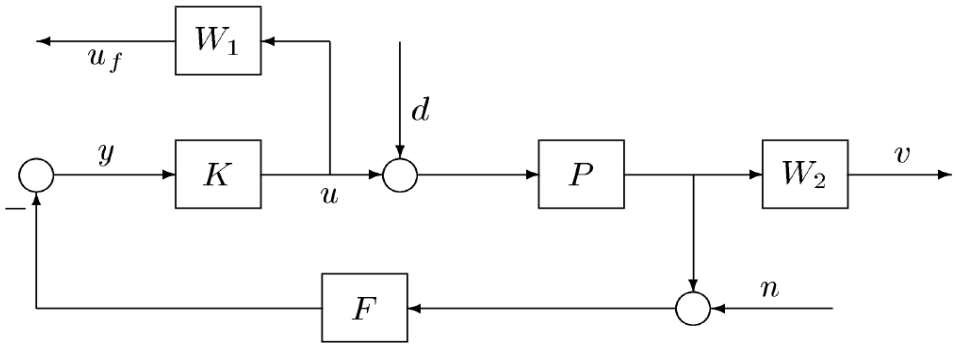

Figure 12: A MIMO Control System Block Diagram with Weights Assigned

The objective signal vector here will have both u_f and v (See that $u_f = W_1 u$, where u is the control input signal). Other vectors are chosen in similar fashion as previous systems.

$$\begin{bmatrix} v \\ u_f \\ \hline y \end{bmatrix} \left[\begin{array}{cc|c} W_2P & 0 & W_2P \\ 0 & 0 & W_1 \\ \hline -FP & -F & -FP \end{array}\right] \begin{bmatrix} d \\ n \\ \hline u \end{bmatrix}$$

The LFT structure is $z = \mathscr{F}_l(G, K)w$. The elements of G are clear from the partition shown above.

5 Uncertainities

The previous section dealt with uncertainities in various systems in detail and it was shown that using LFT framework these uncertainities can be handled very efficiently. However, we did not look much into the modeling of Δ, the uncertainity. This section deals briefly on the types of uncertainities that exist in a system and gives a short introduction on modeling two such kinds of uncertainities.
Broadly, there may be three kinds of uncertainities in a system:

1. Unstructured Uncertainity
2. Parametric Uncertainity
3. Structured Uncertainity

At the moment, the structured uncertainities are out of scope of this project and hence the other two kinds of uncertainities were studied and are briefly described below:

5.1 Unstructured Uncertainities

Many dynamic perturbations occuring throughout the system may be lumped into one single block, Δ. This is referred to as unstructured uncertainity. To model these uncertainities in a system either transfer function matrix approach or LFT framework could be used. Some examples of unstructured uncertainities are additive uncertainity, multiplicative uncertainity etc. These and many others have been elaborately presented in [1] and [3].

For a system with unstructured uncertainities simulation was performed in MATLAB using the Robust Control Toolbox. The system shown in Figure (11), with $G(s)$ and $C(s)$ transfer function matrices as follows, was chosen. The feedback transfer function matrix $H(s) = 1$.

$$G(s) = \begin{bmatrix} \frac{1}{s+1} & \frac{0.5s+0.2}{(s+0.3)(s+1)} \\ \frac{-0.7}{0.2s+1} & \frac{1}{s+0.3} \end{bmatrix} K(s) = \begin{bmatrix} \frac{7(s+1)}{0.3s+1} & 0 \\ 0 & \frac{18(s+2)}{s+1} \end{bmatrix}$$

The Figures (13), (14),(15),(16) show the transient responses of the two input two output system.

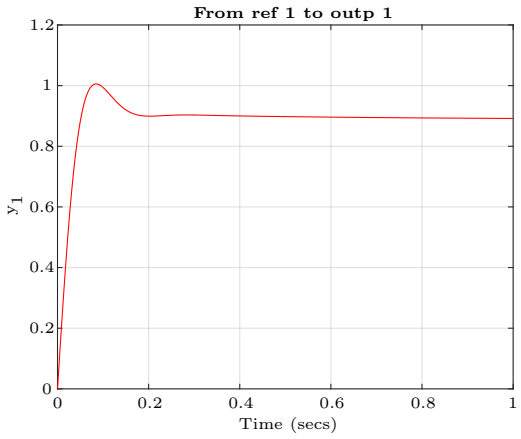

Figure 13: Ref1 to Out1

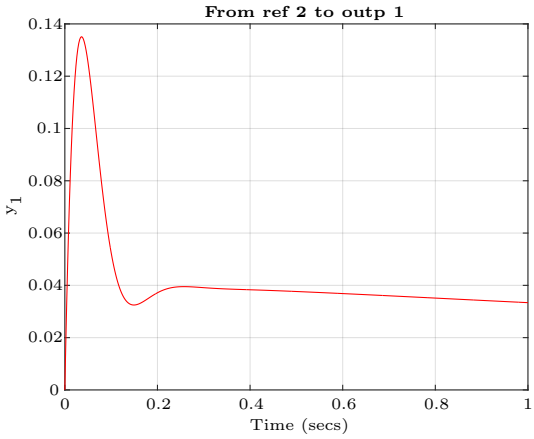

Figure 14: Ref2 to Out1

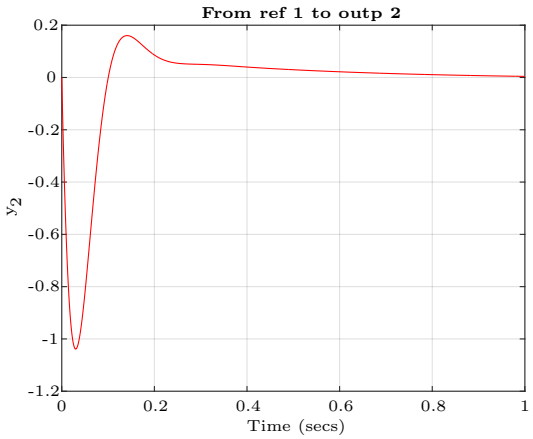

Figure 15: Ref1 to Out2

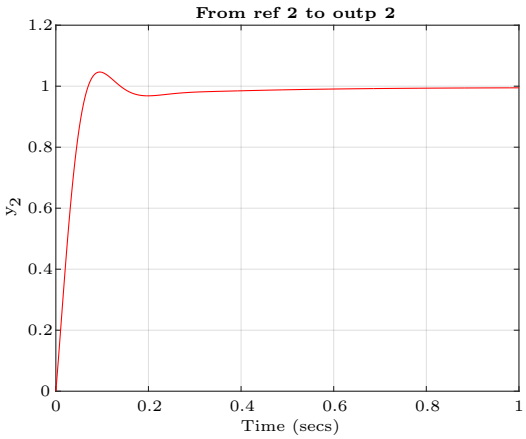

Figure 16: Ref2 to Out2

Also, the Figures (17),(18), (19), (20) show the singular value plots of input and output sensitivity and complimentary sensitivity functions plotted using MATLAB. Along with this, many other simulations were done such as responses to disturbance signal etc.

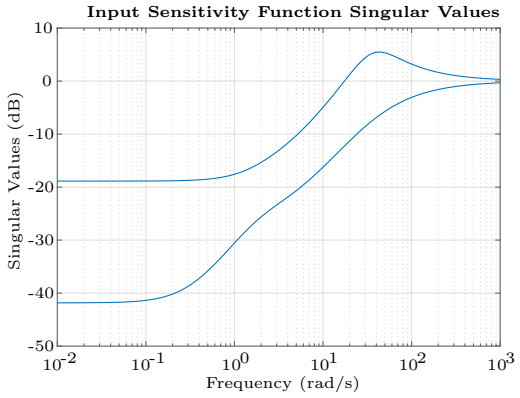

Figure 17: Input Sensitivity Singular Value

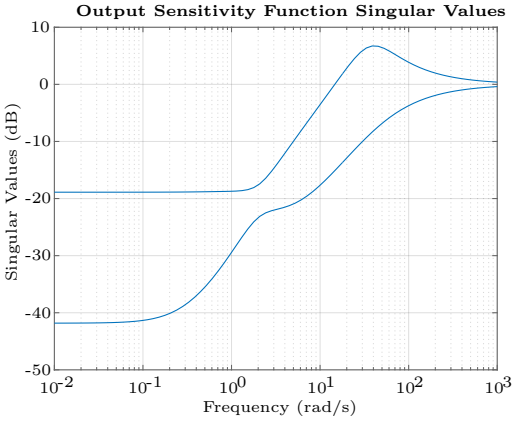

Figure 18: Output Sensitivity Singular Value

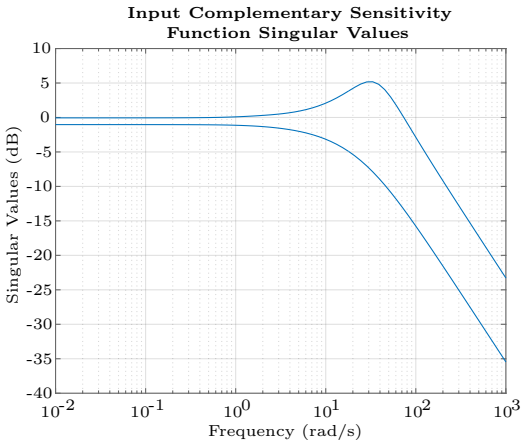

Figure 19: Input Complementary Sensitivity Singular Value

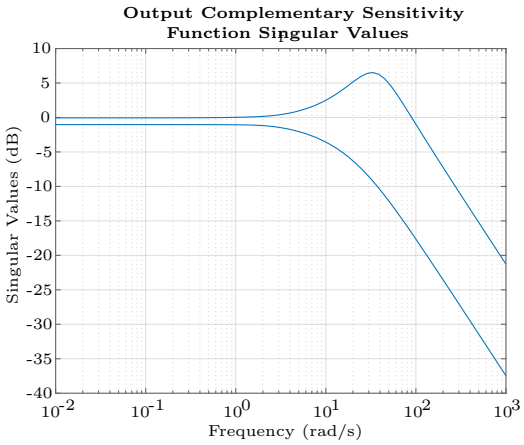

Figure 20: Output Complementary Sensitivity Singular Value

5.2 Parametric Uncertainities

The example of spring mass damper was a perfect example to demonstrate parametric uncertainities in a physical system. These uncertainities are due to unknown exact values of certain parameters within the plant. These are often useful in defining unmodeled or neglected system dynamics.

For spring mass damper system, the parametric uncertainities were modeled in MATLAB and the responses of the system in presence of these uncertainities was studied in detail. The bode plot of the uncertain system is shown in Fig.(21). Also, the step response of the system in presence of these parametric uncertainities of given level is shown in Figure (22).

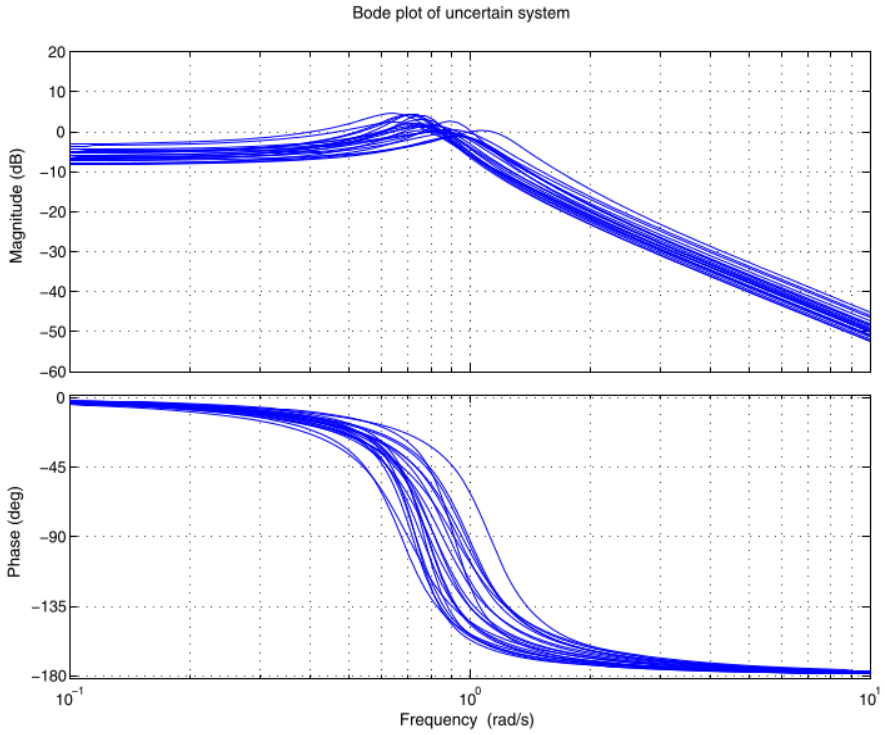

Figure 21: Bode Plot for Spring Mass Damper Uncertain System

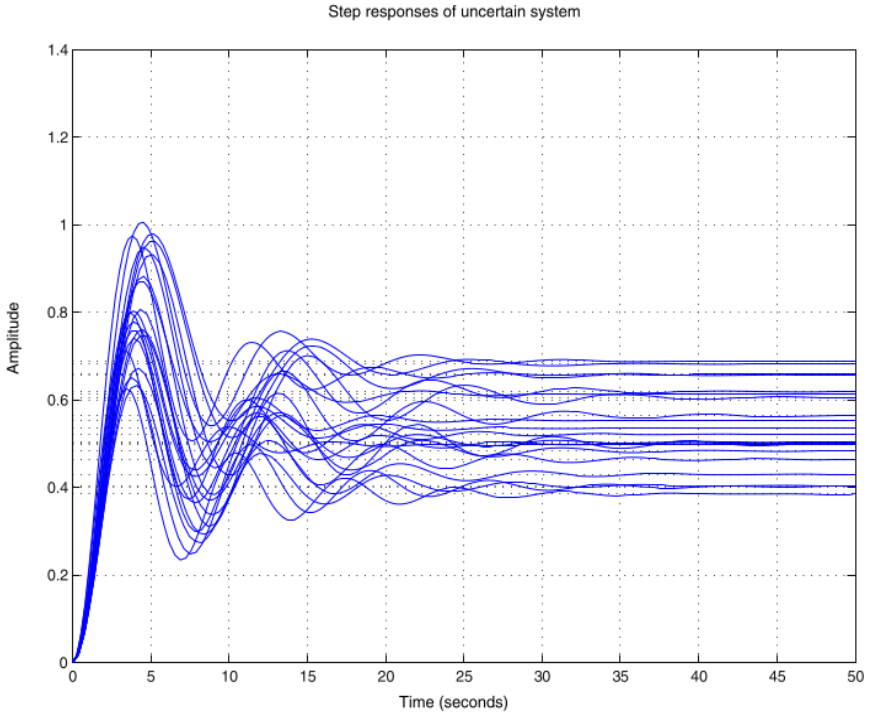

Figure 22: Step Response for Spring Mass Damper Uncertain System

Apart from the spring mass damper system, two other systems with parametric uncertainities was simulated on MATLAB.

6 Small-Gain Theorem

In general, the small-gain theorem provides a sufficient condition for stability of a given system. To ensure internal stability, it is widely used in robust control system designs. For a given M-Δ structure, the theorem states that the system is internally stable if,

$$\|M\|_\infty \|\Delta\|_\infty < 1 \tag{15}$$

The infinity norm can be calculated as shown in Section 3.4. The small-gain theorem can then be formulated as follows:

$$\sup_\omega(M(j\omega)) \sup_\omega(\Delta(j\omega)) = \|M\|_\infty \|\Delta\|_\infty < 1 \ \ \forall\omega \in \mathbb{R} \tag{16}$$

where $\Delta, M \in \mathbb{RH}_\infty$.
The small-gain theorem is used in analysis problem of a given closed loop control system. The analysis of a given system involves checking for robust

stability of a system given a controller for the nominal plant. To analyse a closed loop system, first, it needs to be reduced to LFT framework. This can be done by methods described in the previous section. Next, to comment about the internal stability of the closed loop system the small-gain theorem statement above needs to be verified. From Equation 16, we see that first we need to find out the norm of the M block. A later section in this report gives the details on calculation of the norm and its simulations on MATLAB. However, to analyse systems where the controller is dependent on a few unknown parameters, we need to calculate the norm using other methods which involve the use of Bounded Real Lemma and the Linear Matrix Inequalities.

7 H_∞ Control

Aim of any robust controller synthesis would be to allow for as much uncertainities in the system as possible. Using the small-gain theorem, this controller design criteria can be fulfilled by minimizing the infinity norm of nominal plant model (M, see section 6). This is the very idea that forms the basis of H_∞ control.

7.1 Optimization Problem and Cost Functions

The objective of H_∞ controller design is to find a stabilizing controller K to minimize the output z, in the sense of energy (i.e. the induced 2-norm or the transfer matrix matrix operator norm), over all w with energy less than or equal to 1. This is equivalent to minimizing the H_∞ norm of the transfer function from w to z. This transfer function has been shown to be represented in terms of LFT structure and hence, the design objective is stated as follows:

$$\min_{K stabilizing} \|\mathscr{F}_l(P, K)\|_\infty \tag{17}$$

where the generalized plant is given by P.

7.2 H_∞ Suboptimal Solutions

At times, in practical designs it suffices to achieve the above minimization upto a certain extent, i.e. the H_∞ norm above can be minimized such that it is less than a given γ where $\gamma > \gamma_{min}$. The solution so obtained is called a suboptimal solution.

7.3 Calculation of H_∞ norm

As explained in Section(3.4), the H_∞ norm for a transfer matrix $G(s) \in \mathbb{RH}_\infty$ is given by maximum singular value of the matrix. The H_∞ norm is defined as follows:

$$\|G(s)\|_\infty = \sup_w \bar{\sigma}(G(s)) \tag{18}$$

which implies the following should be satisfied for all frequencies

$$\underline{\sigma}(G) \leqslant \|G\|_\infty \leqslant \bar{\sigma}(G) \tag{19}$$

Some examples were worked out regarding the above. For a constant matrix G given as,

$$G = \begin{bmatrix} 5 & 4 \\ 3 & 2 \end{bmatrix}$$

Using,

$$\sigma_i = \sqrt{\lambda_i(G^*G)} \tag{20}$$

The singular values were calculated and maximum of the values was found. This was verified using the MATLAB function *hinfnorm* of the Robust Control Toolbox. Also, the singular values of this matrix were plotted on logarithmic scale. The plot is shown in Figure (23).

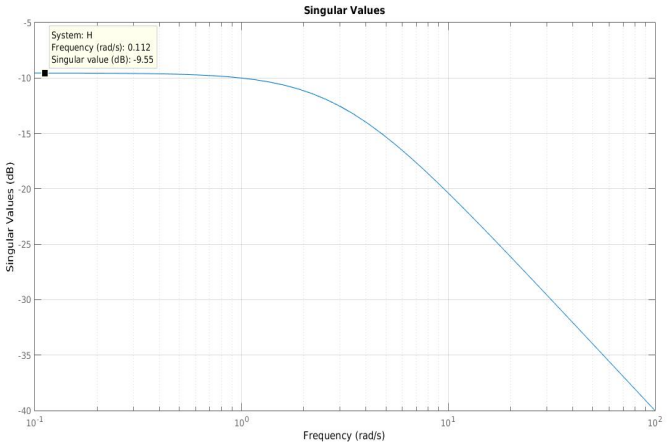

Figure 23: Singular Value Plot for SISO system

The singular value plot shows the system gain and the peak of the plot corresponds to the H_∞ norm of the given matrix. This peak value represents the maximum gain the system can produce to a given input. For

SISO systems, this is equal to the peak value of the frequency response plot. This fact was verified by calculating the H_∞ norm of a given SISO system. The result was verified using MATLAB code as well. The system $H(s) = \frac{1}{s+3}$ was chosen and the peak value of the magnitude frequency response was obtained to be equal to 0.33. The MATLAB plot displayed the corresponding value in dB to be equal to -9.54, hence verifying the stated fact.
Extending the above to multivariable system where the H_∞ norm is equal to the maximum singular value of the system, the following system was evaluated:

$$A = \begin{bmatrix} -1 & 0 \\ 0 & -3 \end{bmatrix} B = \begin{bmatrix} 0 & 1 \\ 2 & 1 \end{bmatrix}$$

$$C = \begin{bmatrix} 1 & 2 \\ 1 & 0 \end{bmatrix} D = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}$$

The maximum singular value for this system is equal to 2.2836. The MATLAB simulation of this system was used to verify the result. The concerned plot is shown in Figure(24).

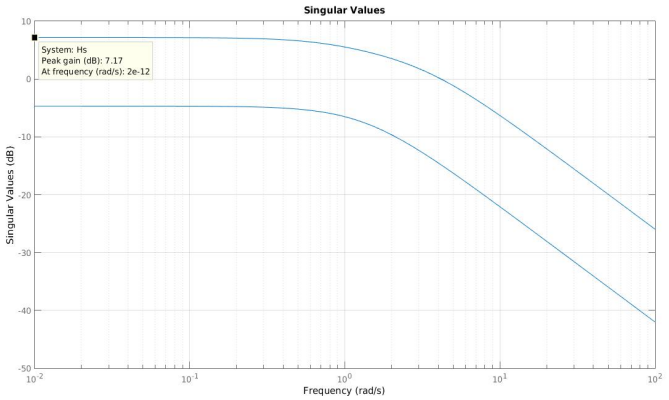

Figure 24: Singular Value Plot for the MIMO system

8 LMI Based Approach to H_∞ Control

8.1 Preliminaries

This section gives an introduction to convex optimization based technique to solve the H_∞ control optimization problem. Before 1995, two algebraic riccati equation based methods were used to solve the H_∞ control optimization problem, the disadvantage with these methods is that

they require various assumptions to be satisfied which are usually not satisfied for many plant models. A convex optimization based approach was proposed in the mid 90s which was based on solving a few linear matrix inequalities (LMIs) inorder to result in a stabilizing controller for the H_∞ control problem. The LMI based approach is easily applicable to a range of systems and is covered in detail in the following sections.

8.1.1 Linear Matrix Inequality

An optimization problem with convex constraints can be formulated in the LMI framework. An LMI has the following form.

$$F(x) = F_0 + \sum_{i=1}^{m} x_i F_i > 0 \tag{21}$$

where, $x = [x_1 x_2 ... x_m]$ is the design variable and F_i, $i = 0, 1...m$ are symmetric matrices. The constraint shown in Eq.(21) indicates the positive definiteness of the matrix $F(x)$. Formulation of an optimization problem in the LMI framework helps the designer to make use of the available LMI solvers in softwares such as MATLAB to efficiently solve the optimization problem.

8.1.2 Bounded Real Lemma

As previously seen in Section(7), closed loop performance and robust stability requirements for a system expressed in LFT framework can be expressed as H_∞ norm minimization problem of certain stable closed loop transfer function matrix of the form $\|T_{zw}\|_\infty < \gamma$. The bounded real lemma gives equivalent conditions to this minimization in LMI framework which can be solved to effectively solve the H_∞ control problem. Consider a system in state space representation as follows.

$$\dot{\mathbf{x}} = \mathbf{Ax} + \mathbf{Bu}, x(0) = 0 \tag{22}$$

$$\mathbf{y} = \mathbf{Cx} + \mathbf{Du} \tag{23}$$

In compact matrix notation, we can write (denoting the system TFM as G(s)):

$$G(s) = \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right]$$

The bounded real lemma states that $\|G(s)\|_\infty < \gamma \Leftrightarrow \exists\, Y = Y^T > 0$ such that:

$$\begin{bmatrix} YA + A^TY & YB & C^T \\ B^TY & -\gamma I & D^T \\ C & D & -\gamma I \end{bmatrix} < 0 \tag{24}$$

The above LMI can be solved for Y and γ using convex optimization techniques. The value of γ corresponds to the H_∞ norm of the TFM $G(s)$ which can be used to design a stabilizing controller.

Proof We can write

$$\|G(s)\|_\infty = \sup_{u \neq 0} \frac{\|y\|_2^2}{\|u\|_2^2} \tag{25}$$

We have, $\|G(s)\|_\infty < \gamma$ and we need to derive the relation in Eq.(24). We define a supply rate function as $(\gamma^2 \|u\|_2^2 - \|y\|_2^2)$. From Eq.(25), we have that the supply rate function is positive. Physically, this is the energy supply rate to a system whose input is u and output, y. Let, $V(x)$ be the energy storage function for this system. Then, we directly have that

$$\dot{V}(x) < (\gamma^2 \|u\|_2^2 - \|y\|_2^2) \tag{26}$$

That is, the rate of supply of energy to the system would always be more than the rate of storage of energy in the system. For a linear time invariant systems with minimal realization, this function $V(x)$ can be considered to be the Lyapunov function of the system. Now, let $V(x)$ be a quadratic Lyapunov function $x^T P x$. For the system in Eq.(22), we evaluate Eq.(26) and on simple manipulation we get,

$$\Leftrightarrow \begin{bmatrix} x^T & u^T \end{bmatrix} \begin{bmatrix} A^T P + PA + C^T C & PB + C^T D \\ B^T P + D^T C & -\gamma^2 I + D^T D \end{bmatrix} \begin{bmatrix} x \\ u \end{bmatrix} < 0$$

$$\Leftrightarrow \begin{bmatrix} A^T P + PA + C^T C & PB + C^T D \\ B^T P + D^T C & -\gamma^2 I + D^T D \end{bmatrix} < 0$$

Now, we know that for a function to be a Lyapunov function candidate, the following properties are satisfied:

1. $V(0) = 0$
2. $V(x) > 0 \, \forall \, x \neq 0$

Using this and the fact that for an LTI system stability doesn't depend on the input, we can see from Eq.(26) that the system is stable as $\dot{V}(x) < 0$. On integrating Eq.(26) for time 0 to time t, we obtain

$$\Leftrightarrow V(x(t)) - V(x(0)) < \int_0^t (\gamma^2 \|u_2\|^2 - \|y_2\|^2) dt \tag{27}$$

Since $x(0) = 0$, We have $V(x(0)) = 0$ and $V(x(t))$ is always positive. Hence,

$$(\gamma^2 \|u\|_2^2 - \|y\|_2^2) > 0 \tag{28}$$

$$\Leftrightarrow \frac{\|y\|_2^2}{\|u\|_2^2} < \gamma^2 \tag{29}$$

$$\Leftrightarrow \|G(s)\|_\infty < \gamma \tag{30}$$

We have proved that $\|G(s)\|_\infty < \gamma$ is equivalent to, when $\exists Y = Y^T > 0$ such that

$$\begin{bmatrix} A^TP + PA + C^TC & PB + C^TD \\ B^TP + D^TC & -\gamma^2 I + D^TD \end{bmatrix} < 0$$

Now, using the Schur's complement lemma, we will prove the statement of the bounded real lemma. The Schur's complement lemma for a symmetric matrix states that

$$\begin{bmatrix} A & B \\ B^T & D \end{bmatrix} < 0 \Leftrightarrow A - BD^{-1}B^T < 0\,;\; D < 0 \Leftrightarrow D - B^TA^{-1}B < 0\,;\; A < 0$$

For the LMI in the bounded real lemma statement, using Schur's complement taking

$$A = \begin{bmatrix} YA + A^TY & YB \\ B^TY & -\gamma I \end{bmatrix},\; B = \begin{bmatrix} C^T \\ D^T \end{bmatrix},\; D = \begin{bmatrix} -\gamma I \end{bmatrix}$$

Using the above matrices in the Schur's complement's first form, we can prove the equivalence relation in the bounded real lemma, which completes the proof.

8.1.3 LMI Toolbox

The *mincx* function in MATLAB helps in minimizing linear objective under LMI constraint. To use this, first an LMI needs to be defined using *setlmis*, *lmivar* and *lmiterm* commands. The details for these commands are available in MATLAB documentation. After an LMI has been defined in MATLAB, the linear objective function which needs to be minimized is defined using *defcx* function. Once this is done, the *mincx* command yields the minimized objective function value as well as the value at which this minimum occurs. This completes the H_∞ norm calculation. The MATLAB code written for the above system is avaliable at [6].

8.1.4 YALMIP Toolbox

YALMIP is a modelling language for advanced modeling and solution of convex and nonconvex optimization problems. Its function *optimize* can

be used to solve a convex optimization problem. The YALMIP toolbox code works on symbolic decision variables which are defined by the user. These decision variables can be then used to define the LMI in a very user friendly manner. YALMIP toolbox can be used to solve both strict and non-strict constraint LMIs while MATLAB's LMI toolbox only handles strict constraints. For the system shown above, the MATLAB program which solves the LMI given by the bounded real lemma is available at [6]. As expected, the result for the H_∞ norm is same using all the three methods.

8.2 LMI Approach

As mentioned in the Introduction section, there are three methods to solve a H_∞ control problem. The first two methods, viz. the DGKF method and the Doyle-Glover method are based on finding out a stabilizing solution to certain algebraic riccati equation. The design procedure in these methods works only when the system satisfies a set of assumptions which are usually difficult to satisfy for a given plant model. On the other hand, the LMI based convex optimization approach only requires two assumptions, which are satisfied easily for a range of different systems. Also, it has been proved that the controller structure found in the two methods can be similarly derived in the LMI framework. Hence, the LMI based approach to H_∞ control is widely used due to its ease of design on the part of the designer. Using the bounded real lemma stated and proved in the previous section, the H_∞ norm of a TFM can be found out by solving the LMI. This can be used to design a stabilizing controller $K(s)$ for a system by minimizing the H_∞ norm of the closed loop TFM. The LMI approach to H_∞ control requires two assumptions to be satisfied for the generalized plant P. Let a generalized plant be expressed in state space representation as follows.

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

The following two assumptions should be satisfied to apply LMI based approach to solve the H_∞ control problem:

1. (A,B_1,C_1) stabilizable and detectable.
2. $D_{22} = 0$

The controller design to minimize the H_∞ norm of the generalized plant using LMI approach has been covered in detail in [4]. A simple numerical example is shown in the next section which demonstrates the use of LMI toolbox and YALMIP toolbox to calculate the H_∞ norm of a given system in MATLAB using bounded real lemma.

8.2.1 Numerical Example

In Section(7), we took a simple stable system and calculated its infinity norm.

$$A = \begin{bmatrix} -1 & 0 \\ 0 & -3 \end{bmatrix}, \; B = \begin{bmatrix} 0 & 1 \\ 2 & 1 \end{bmatrix}, \; C = \begin{bmatrix} 1 & 2 \\ 1 & 0 \end{bmatrix}, D = 0_{2\times 2}$$

Using MATLAB's Robust Control Toolbox *hinfnorm* function, the value obtained was 2.2836. The same result was obtained by solving the LMI that the bounded real lemma gives using the LMI tooblox and the YALMIP toolbox [5].

9 H_∞ Loop-Shaping Control

The H_∞ loop-shaping design technique incorporates classical loop shaping method to obtain performance and robust stability tradeoffs along with H_∞ control optimization technique to gurantee a level of closed loop robust stability. This method ensures robustness against unstructured uncertainty which is described as perturbations to normalized coprime factors of the shaped plant.
The H_∞ controller design optimization methods face a few disadvantages with respect to model perturbations being limited by number of poles in the RHP and the possibility of undesirable pole-zero cancellation between the nominal plant and the H_∞ controller. In H_∞ loop shaping design technique, the perturbations are directly described on the coprime factors of the nominal model. This helps in relaxing the restriction on the number of RHP poles and also doesn't produce any pole-zero cancellation. More importantly, this design technique is computationally more efficient as it doesn't require an iterative procedure for finding γ. Finally, since this design technique inherits concepts of loop shaping which is a classical controller design technique, hence it is much more user friendly method to design a robust controller.

9.1 Coprime Factor Uncertainty Description

A brief introduction to coprime factorization was given in Section(3.6). We consider left coprime facotrization of a plant P given by $\tilde{M}^{-1}\tilde{N}$, where $\tilde{M}$ and $\tilde{N} \in \mathbb{RH}_\infty$. Unstructured uncertainties can be described in a more general sense by perturbations on the coprime factors of a plant ($\tilde{\Delta}_m$ and $\tilde{\Delta}_n \in \mathbb{RH}_\infty$). The Figure(25) shows perturbations on the left coprime factors of the plant P considered above.

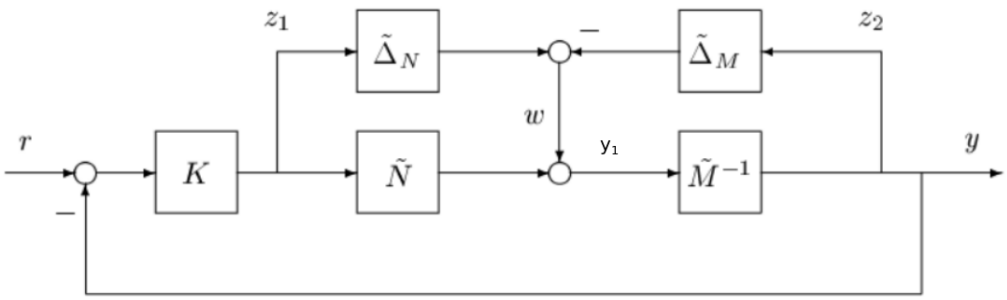

Figure 25: Perturbations to Coprime Factors : Block Diagram

From the block diagram we have,

$$y_1 = \tilde{M}y \tag{31}$$
$$y_1 = \tilde{N}u + \tilde{\Delta}_n u - \tilde{\Delta}_m y \tag{32}$$

We have from these two equations, eliminating y_1

$$y = (\tilde{M} + \tilde{\Delta}_m)^{-1}(\tilde{N} + \tilde{\Delta}_n)u \tag{33}$$

Hence, the perturbed plant in this case is given by

$$P = (\tilde{M} + \tilde{\Delta}_m)^{-1}(\tilde{N} + \tilde{\Delta}_n) \tag{34}$$

Now with this coprime factor perturbed plant, we will study the robust stabilization and the H_∞ loop shaping controller design technique.

9.2 Robust Stabilization of Coprime Factors

In Eq.(34), we assume that $\left\|[\tilde{\Delta}_m \ \tilde{\Delta}_n]\right\|_\infty < \epsilon$. We also assume that in Fig.(25) the controller K internally stabilizes the nominal plant. The robust stability analysis for this system can be done by reducing the block diagram in Fig.(25) to LFT framework and applying the small gain theorem. For the equivalent $M - \Delta$ structure, we can obtain M as follows.

$$
\begin{aligned}
z_2 = y\ ;\ z_1 &= -ky \\
y &= \tilde{M}^{-1}(\tilde{N}u + w) \\
\Rightarrow z_1 &= -KPu - K\tilde{M}^{-1}w \\
\Rightarrow z_2 &= Gu + \tilde{M}^{-1}w \\
y &= -PKy + \tilde{M}^{-1}w \\
\Rightarrow y &= (I + PK)^{-1}\tilde{M}^{-1}w \\
\Rightarrow M &= \begin{bmatrix} K \\ I \end{bmatrix} (I + PK)^{-1}\tilde{M}^{-1}
\end{aligned}
$$

Hence, on using small gain theorem, we obtain the following condition for robust stability of the system.

$$
\left\| \begin{bmatrix} K \\ I \end{bmatrix} (I + PK)^{-1}\tilde{M}^{-1} \right\|_{\infty} \leq \frac{1}{\epsilon} \tag{35}
$$

Now, the following theorem from coprime factorization theory could be used to derive the generalized plant expression for the system shown in Fig.(25). The theorem gives the left and right coprime factorizations for a proper real-rational TFM. Let a system P have the following stabilizable and detectable minimal state space realization.

$$
P = \left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] \tag{36}
$$

Now, let B and L be matrices such that $A + LC$ and $A + BF$ are both stable. Then we can define:

$$
\begin{bmatrix} M & -Y_l \\ N & X_l \end{bmatrix} = \left[\begin{array}{c|cc} A + BF & B & -L \\ \hline F & I & 0 \\ C + DF & D & I \end{array}\right] \tag{37}
$$

$$
\begin{bmatrix} X_r & Y_r \\ -\tilde{N} & \tilde{M} \end{bmatrix} = \left[\begin{array}{c|cc} A + LC & -(B + LD) & L \\ \hline F & I & 0 \\ C & -D & I \end{array}\right] \tag{38}
$$

Then $P = NM^{-1} = \tilde{M}^{-1}\tilde{N}$ are rcf and lcf respectively. The above theorem can be proved by verifying the Bezout's identity for double coprime facotrization of plant P. Now, using Eq.(38), we get the following result for normalized LCF of P.

$$
\begin{bmatrix} \tilde{N} & \tilde{M} \end{bmatrix} = \left[\begin{array}{c|cc} A + LC & B + LD & L \\ \hline C & D & I \end{array}\right] \tag{39}
$$

Using Eq.(39) and denoting the controller $\hat{K} = -K$ in Fig.(25), we have for $z = [z_1\ z_2]^T$

$$\begin{aligned} z_1 &= u \\ z_2 &= \tilde{M}^{-1}w + \tilde{M}^{-1}\tilde{N}u \\ y &= \tilde{M}^{-1}w + \tilde{M}^{-1}\tilde{N}u \end{aligned}$$

Hence, the generalized plant in the equivalent closed loop LFT structure (as shown in Fig.(26)) is as follows (in the transfer function form).

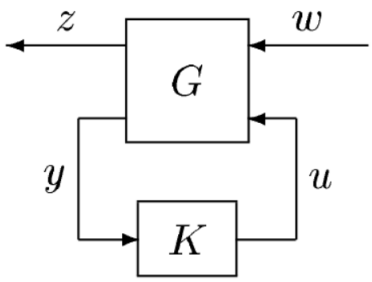

Figure 26: Lower LFT Representation

$$G(s) = \begin{bmatrix} \begin{bmatrix} 0 \\ \tilde{M}^{-1} \end{bmatrix} & \begin{bmatrix} I \\ P \end{bmatrix} \\ \tilde{M}^{-1} & P \end{bmatrix} \tag{40}$$

And in state space form is given by

$$G = \left[\begin{array}{c|cc} A & -L & B \\ \hline \begin{bmatrix} 0 \\ C \end{bmatrix} & \begin{bmatrix} 0 \\ I \end{bmatrix} & \begin{bmatrix} I \\ D \end{bmatrix} \\ C & I & D \end{array}\right] \tag{41}$$

Using $G(s) = C(sI - A)^{-1}B + D$ we can verify that Eq.(41) is indeed the state space representation of the TF representation given in Eq.(40).

$$C = \begin{bmatrix} \begin{bmatrix} 0 \\ C \end{bmatrix} \\ C \end{bmatrix}; B = \begin{bmatrix} -L & B \end{bmatrix}; D = \begin{bmatrix} \begin{bmatrix} 0 \\ I \end{bmatrix} & \begin{bmatrix} I \\ D \end{bmatrix} \\ I & D \end{bmatrix}$$

This is an important result which would help in the design of H_∞ loop-shaping design based controller using LMI approach.

9.3 Design Steps

H_∞ loop shaping controller design steps were studied as presented in [1]. Using pre-compensator and/or post-compensator the singular values of the nominal plant are shaped first to get a desired open loop shape. This shaped plant is represented as G_s, which is equal to W_2GW_1. Next, the maximum robust stability margin is calculated. It is given by

$$\epsilon_{max} = \sqrt{1 - \left\| \begin{bmatrix} \tilde{N}_s & \tilde{M}_s \end{bmatrix} \right\|_H^2} < 1 \tag{42}$$

where $G_s = \tilde{M}_s^{-1}\tilde{N}_s$ and $\tilde{M}_s\tilde{M}_s^* + \tilde{N}_s\tilde{N}_s^* = I$, i.e. $\tilde{M}_s$ and $\tilde{N}_s$ are normalized left coprime factors of the shaped plant G_s.
Now, selecting $\epsilon < \epsilon_{max}$, we go ahead with the controller design based on H_∞ optimization as follows.

$$\left\| \begin{bmatrix} I \\ K_\infty \end{bmatrix} (I + G_sK_\infty)^{-1}\tilde{M}_s^{-1} \right\|_\infty \leq \epsilon^{-1} \tag{43}$$

The final feedback controller is given by $K = W_1K_\infty W_2$. The calculation of ϵ_{max}, ϵ and the weights W_1 and W_2 are the choices that the designer has to make judiciously.

9.4 Solvability Conditions

The H_∞ loop-shaping controller design problem using LMI approach will be described in detail in this and the next section. First, in this section, conditions for the exsitence of a suboptimal controller for the generalized plant obtained for coprime factor uncertainity description in previous section (Eq.(41)) would be presented as LMIs. These existence conditions are LMIs which can be easily solved using convex optimization techniques.
The key idea in deriving these existence conditions is to use the LMI based existence conditions presented in [4] for H_∞ control problem by simply replacing the matrices A,B_1,B_2,C_1,C_2,D_{11},D_{12},D_{21} and D_{22} with those obtained for coprime factor uncertainity description (See Eq.(41))used in the design of H_∞ loop-shaping controller. The following subsections give a detailed insight into the Theorems that would be used to derive the existence conditions. The theorems statements are as described in [4].

9.4.1 Generalized Plant Description

For the LFT structure shown in Fig.(27), it has been shown in Section(4) that $T_{zw} = \mathscr{F}_l(P, K)$, where $\mathbb{F}_l(P, K) = p_{11} + p_{12}K(I - p_{22}K)^{-1}p_{21}$ for

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

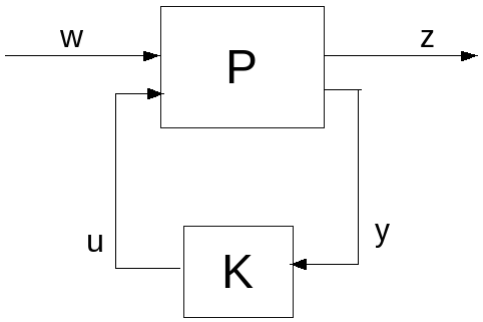

Figure 27: Lower LFT Representation of Plant P and Controller K

The above can be represented in state space form. Let $x \in \mathbb{R}^{n\times 1}$ be the state vector. For outputs $z \in \mathbb{R}^{p_1\times 1}$ and $y \in \mathbb{R}^{p_2\times 1}$ and inputs $w \in \mathbb{R}^{m_1\times 1}$ and $u \in \mathbb{R}^{m_2\times 1}$, we can write the following equations:

$$\dot{x} = Ax + B_1 w + B_2 u \tag{44}$$
$$z = C_1 x + D_{11} w + D_{12} u \tag{45}$$
$$y = C_2 x + D_{21} w + D_{22} u \tag{46}$$

In packed matrix notation, the above can be written as follows.

$$\left[\begin{array}{c} \dot{x} \\ \hline z \\ y \end{array}\right] = \left[\begin{array}{c|cc} A & B_1 & B_2 \\ \hline C_1 & D_{11} & D_{12} \\ C_2 & D_{21} & D_{22} \end{array}\right] \left[\begin{array}{c} x \\ \hline w \\ u \end{array}\right]$$

It is straightforward from the above discussion that $A \in \mathbb{R}^{n\times n}$, $B_1 \in \mathbb{R}^{n\times m_1}$, $B_2 \in \mathbb{R}^{n\times m_2}$, $C_1 \in \mathbb{R}^{p_1\times n}$, $C_2 \in \mathbb{R}^{p_2\times n}$, $D_{11} \in \mathbb{R}^{p_1\times m_1}$, $D_{12} \in \mathbb{R}^{p_1\times m_2}$, $D_{21} \in \mathbb{R}^{p_2\times m_1}$ and $D_{22} \in \mathbb{R}^{p_2\times m_2}$. Next, the controller can be written as follows:

$$K = \left[\begin{array}{c|c} A_k & B_k \\ \hline C_k & D_k \end{array}\right]$$

where $A_k \in \mathbb{R}^{k\times k}$, $B_k \in \mathbb{R}^{k\times p_2}$, $C_k \in \mathbb{R}^{m_2\times k}$ and $D_k \in \mathbb{R}^{m_2\times p_2}$ are the state matrices for the controller K, where $u = K(s)y$. We can also write,

$$\dot{x}_k = A_k x_k + B_k y \tag{47}$$
$$u = C_k x_k + D_k y \tag{48}$$

We begin by writing the closed loop structure shown in Fig. (4) in state space form. This can be done by absorbing the controller K into the state space representation to obtain the following:

$$\left[\begin{array}{c} \dot{x} \\ \dot{x}_k \\ \hline z \end{array}\right] = \left[\begin{array}{c|c} A_{cl} & B_{cl} \\ \hline C_{cl} & D_{cl} \end{array}\right] \left[\begin{array}{c} x \\ x_k \\ \hline w \end{array}\right] \tag{49}$$

where the closed loop transfer function from w to z is given by $\mathscr{F}(P,K)(s) = C_{cl}(sI - A_{cl})^{-1}B_{cl} + D_{cl}$. We will next show the derivation of these closed loop matrices.

$$u = Ky$$

From Eq.(46) and (48), we have

$$\begin{aligned}
&y = C_2x + D_{21}w + D_{22}C_kx_k + D_{22}D_ky \\
\Rightarrow\ &y = (I - D_{22}D_k)^{-1}C_2x + (I - D_{22}D_k)^{-1}D_{21}w + (I - D_{22}D_k)^{-1}D_{22}C_kx_k \\
&z = C_1x + D_{11}w + D_{12}C_kx_k + D_{12}D_ky \\
\Rightarrow\ &z = (C_1 + D_{12}D_k(I - D_{22}D_k)^{-1}C_2)x + (D_{12}C_k + D_{12}D_k(I - D_{22}D_k)^{-1}D_{22}C_k)x_k + \\
&\quad (D_{11}w + D_{12}D_k(I - D_{22}D_k)^{-1}D_{21})w \\
&\dot{x} = Ax + B_1w + B_2C_kx_k + B_2D_k(I - D_{22}D_k)^{-1}C_2x + B_2D_k(I - D_{22}D_k)^{-1}D_{21}w + \\
&\quad B_2D_k(I - D_{22}D_k)^{-1}D_{22}C_kx_k \\
\Rightarrow\ &\dot{x} = (A + B_2D_k(I - D_{22}D_k)^{-1}C_2)x + (B_2C_k + B_2D_k(I - D_{22}D_k)^{-1}D_{22}C_k)x_k + \\
&\quad (B_1 + B_2D_k(I - D_{22}D_k)^{-1}D_{21})w \\
&\dot{x}_k = A_kx_k + B_k(I - D_{22}D_k)^{-1}C_2x + B_k(I - D_{22}D_k)^{-1}D_{12}C_kx_k + \\
&\quad B_k(I - D_{22}D_k)^{-1}D_{21}w
\end{aligned}$$

Using the above equations, we can find out A_{cl}, B_{cl}, C_{cl} and D_{cl} to satisfy Eq.(49).

$$A_{cl} = \begin{bmatrix} A + B_2D_k(I - D_{22}D_k)^{-1}C_2 & B_2C_k + B_2D_k(I - D_{22}D_k)^{-1}D_{22}C_k \\ B_k(I - D_{22}D_k)^{-1}C_2 & A_k + B_k(I - D_{22}D_k)^{-1}D_{22}C_k \end{bmatrix}$$

$$B_{cl} = \begin{bmatrix} B_1 + B_2D_k(I - D_{22}D_k)^{-1}D_{21} \\ B_k(I - D_{22}D_k)^{-1}D_{21} \end{bmatrix}$$

$$C_{cl} = \begin{bmatrix} C_1 + D_{12}D_k(I - D_{22}D_k)^{-1}C_2 & D_{12}C_k + D_{12}D_k(I - D_{22}D_k)^{-1}D_{22}C_k \end{bmatrix}$$

$$D_{cl} = D_{11} + D_{12}D_k(I - D_{22}D_k)^{-1}D_{21}$$

Also, as it is clear that if D_{22} is chosen as a null matrix the above equations are simplified to a great extent. So, taking $D_{22} = 0$, we get:

$$A_{cl} = \begin{bmatrix} A + B_2D_kC_2 & B_2C_k \\ B_kC_2 & A_k \end{bmatrix}, B_{cl} = \begin{bmatrix} B_1 + B_2D_kD_{21} \\ B_kD_{21} \end{bmatrix}$$

$$C_{cl} = \begin{bmatrix} C_1 + D_{12}D_kC_2 & D_{12}C_k \end{bmatrix}, D_{cl} = D_{11} + D_{12}D_kD_{21}$$

Hence, $D_{22} = 0$ is one of the assumptions in the LMI approach to H_∞ controller design. Also, we can simplify the above matrices by using the

following notation:

$$A_0(\in \mathbb{R}^{(n+k)\times(n+k)}) = \begin{bmatrix} A & 0 \\ 0 & 0_{k\times k} \end{bmatrix}; B_0(\in \mathbb{R}^{(n+k)\times(m_1)}) = \begin{bmatrix} B_1 \\ 0_{k\times m_1} \end{bmatrix} \tag{50}$$

$$C_0(\in \mathbb{R}^{(p_1)\times(n+k)}) = \begin{bmatrix} C_1 & 0_{p_1\times k} \end{bmatrix}; \mathscr{B}(\in \mathbb{R}^{(n+k)\times(m_2+k)}) = \begin{bmatrix} 0_{n\times k} & B_2 \\ I_k & 0_{k\times m_2} \end{bmatrix} \tag{51}$$

$$\mathscr{C}(\in \mathbb{R}^{(p_2+k)\times(n+k)}) = \begin{bmatrix} 0_{k\times n} & I_k \\ C_2 & 0_{p_2\times k} \end{bmatrix}; \mathscr{D}_{12}(\in \mathbb{R}^{(p_1)\times(m_2+k)}) = \begin{bmatrix} 0_{p_1\times k} & D_{12} \end{bmatrix} \tag{52}$$

$$\mathscr{D}_{21}(\in \mathbb{R}^{(p_2+k)\times(m_1)}) = \begin{bmatrix} 0_{k\times m_1} \\ D_{21} \end{bmatrix} \tag{53}$$

This notation is helpful because it only depends on plant data and as we will see in the following expression the closed loop matrices will depend affinely on controller data if the above notation is used.

$$A_{cl}(\in \mathbb{R}^{(n+k)\times(n+k)}) = A_0 + \mathscr{B}\theta\mathscr{C}; B_{cl}(\in \mathbb{R}^{(n+k)\times m_1}) = B_0 + \mathscr{B}\theta\mathscr{D}_{21}$$

$$C_{cl}(\in \mathbb{R}^{p_1\times(n+k)}) = C_0 + \mathscr{D}_{12}\theta\mathscr{C}; D_{cl}(\in \mathbb{R}^{p_1\times m_1}) = D_{11} + \mathscr{D}_{12}\theta\mathscr{D}_{21};$$

where

$$\theta(\in \mathbb{R}^{(m_2+k)\times(p_2+k)}) = \begin{bmatrix} A_k & B_k \\ C_k & D_k \end{bmatrix}$$

The above can be verified by simple substitution.
Other than the assumption that $D_{22} = 0$, we also assume that the plant is stabilizable and detectable. Hence, using LMI approach to design H_∞ controller only these two assumptions are needed. This is one of the advantages of the LMI approach compared to the DGKF method as mentioned in previous sections. Now, we present the solvability conditions for the existence of a suboptimal controller for the generalized plant described above.

9.4.2 Controller Existence Conditions

Theorem 1 Consider the proper plant $P(s)$ and assume that the two assumptions mentioned above hold true. Define

$$\mathscr{P} := \begin{bmatrix} \mathscr{B}^T & 0_{(k+m_2)\times m_1} & \mathscr{D}_{12}^T \end{bmatrix} \tag{54}$$

$$\mathscr{D} := \begin{bmatrix} \mathscr{C} & \mathscr{D}_{21} & 0_{(k+p_2)\times p_1} \end{bmatrix} \tag{55}$$

and let $W_{\mathscr{P}}$ and $W_{\mathscr{D}}$ be the basis of null spaces of $\mathscr{P}$ and $\mathscr{D}$, respectively. Then the set of $\gamma-$suboptimal controllers of order k is nonempty iff there exists some $(n+k)\times(n+k)$ positive definite matrix X_{cl} such that:

$$W_{\mathscr{P}}^T \Phi_{X_{cl}} W_{\mathscr{P}} < 0 \tag{56}$$

$$W_{\mathscr{D}}^T \Psi_{X_{cl}} W_{\mathscr{D}} < 0 \tag{57}$$

where

$$\Phi_{X_{cl}} := \begin{bmatrix} A_0 X_{cl}^{-1} + X_{cl}^{-1} A_0^T & B_0 & X_{cl}^{-1} C_0^T \\ B_0^T & -\gamma I & D_{11}^T \\ C_0 X_{cl}^{-1} & D_{11} & -\gamma I \end{bmatrix} \tag{58}$$

$$\Psi_{X_{cl}} := \begin{bmatrix} A_0^T X_{cl} + X_{cl} A_0 & X_{cl} B_0 & C_0^T \\ B_0^T X_{cl} & -\gamma I & D_{11}^T \\ C_0 & D_{11} & -\gamma I \end{bmatrix} \tag{59}$$

Proof:
From the Bounded Real Lemma described in Section(8.1.2), for some matrix $X_{cl} = X_{cl}^T > 0 \in \mathbb{R}^{(n+k)\times(n+k)}$, we can write that $\|G_{cl}\|_\infty < \gamma$ is equivalent to

$$\begin{bmatrix} A_{cl}^T X_{cl} + X_{cl} A_{cl} & X_{cl} B_{cl} & C_{cl}^T \\ B_{cl}^T X_{cl} & -\gamma I & D_{cl}^T \\ C_{cl} & D_{cl} & -\gamma I \end{bmatrix} < 0 \tag{60}$$

where G_{cl} in packed matrix notation can be written as,

$$G_{cl} = \left[\begin{array}{c|c} A_{cl} & B_{cl} \\ \hline C_{cl} & D_{cl} \end{array}\right].$$

Now, using Eq.(59) and notation given in (50), we can write Eq.(60) as follows:

$$\begin{bmatrix} A_0^T X_{cl} + \mathscr{C}^T \theta^T \mathscr{B}^T X_{cl} + X_{cl} A_0 + X_{cl} \mathscr{B} \theta \mathscr{C} & X_{cl} B_0 + X_{cl} \mathscr{B} \theta \mathscr{D}_{21} & C_0^T + \mathscr{C}^T \theta^T \mathscr{D}_{12}^T \\ B_0^T X_{cl} + \mathscr{D}_{21}^T \theta^T \mathscr{B}^T X_{cl} & -\gamma I & D_{11}^T + \mathscr{D}_{21}^T \theta^T \mathscr{D}_{12}^T \\ C_0 + \mathscr{D}_{12} \theta \mathscr{C} & D_{11} + \mathscr{D}_{12} \theta \mathscr{D}_{21} & -\gamma I \end{bmatrix} < 0$$

Now observing that,

$$\mathscr{D}^T \theta^T \mathscr{P}_{X_{cl}} = \begin{bmatrix} \mathscr{C}^T \theta^T \\ \mathscr{D}_{21} \theta^T \\ 0 \end{bmatrix} \begin{bmatrix} \mathscr{B}^T X_{cl} & 0 & \mathscr{D}_{12}^T \end{bmatrix} = \begin{bmatrix} \mathscr{C}^T \theta^T \mathscr{B}^T X_{cl} & 0 & \mathscr{C}^T \theta^T \mathscr{D}_{12}^T \\ \mathscr{D}_{21} \theta^T \mathscr{B}^T X_{cl} & 0 & \mathscr{D}_{21} \theta^T \mathscr{D}_{12}^T \\ 0 & 0 & 0 \end{bmatrix}$$

We have,

$$\Psi_{X_{cl}} + \mathscr{D}^T \theta^T \mathscr{P}_{X_{cl}} + \mathscr{P}_{X_{cl}}^T \theta \mathscr{D} < 0 \tag{61}$$

This equation is solvable if and only if,

$$W_{\mathscr{P}_{X_{cl}}}^T \Psi_{X_{cl}} W_{\mathscr{P}_{X_{cl}}} < 0 W_{\mathscr{D}}^T \Psi_{X_{cl}} W_{\mathscr{D}} < 0 \tag{62}$$

where $W_{\mathscr{P}_{X_{cl}}}$ and $W_{\mathscr{D}}$ are the basis of the null spaces of $\mathscr{P}_{X_{cl}}$ and $\mathscr{D}$ respectively. For the proof of this lemma, the reader is referred to Section 3 of [4]. Now, we will reduce the Eq. (62) as follows to prove the statement of the theorem.

$$I = \begin{bmatrix} X_{cl} & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix} \begin{bmatrix} X_{cl}^{-1} & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix}$$

So, we can write,

$$\mathscr{P} = \mathscr{P}_{X_{cl}} \begin{bmatrix} X_{cl}^{-1} & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix}$$

Using the above and finding basis of the null spaces, we have,

$$W_{\mathscr{P}_{X_{cl}}} = \begin{bmatrix} X_{cl}^{-1} & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix} W_{\mathscr{P}}$$

where $W_{\mathscr{P}_{X_{cl}}}$ and $W_{\mathscr{P}}$ are the basis of the null space of $\mathscr{P}_{X_{cl}}$ and $\mathscr{P}$ respectively. Now, rewriting Eq.(62), we get

$$W_{\mathscr{P}}^T \left(\begin{bmatrix} X_{cl}^{-1} & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix} \Psi_{X_{cl}} \begin{bmatrix} X_{cl}^{-1} & 0 & 0 \\ 0 & I & 0 \\ 0 & 0 & I \end{bmatrix} \right) W_{\mathscr{P}} < 0$$

Simplifying, we finally get,

$$W_{\mathscr{P}}^T \Phi_{X_{cl}} W_{\mathscr{P}} < 0$$

Hence, the theorem is proved. The next theorem eradicates the need to compute X_{cl}^{-1} and also shows the role of plant parameters in the solvability conditions.

Theorem 2 Consider a continuous time plant $P(s)$ of order n and minimal realization given in Eq.(36). Let $\mathscr{N}_R$ and $\mathscr{N}_S$ be bases of null space of (B_2^T, D_{12}^T) and (C_2, D_{21}) respectively. The suboptimal H_∞ problem of parameter γ is solvable if and only if the following three conditions (formulated in LMI) are satisfied.

$$\begin{bmatrix} \mathscr{N}_R & 0 \\ 0 & I \end{bmatrix}^T \begin{bmatrix} AR + RA^T & RC_1^T & B_1 \\ C_1 R & -\gamma I & D_{11} \\ B_1^T & D_{11}^T & -\gamma I \end{bmatrix} \begin{bmatrix} \mathscr{N}_R & 0 \\ 0 & I \end{bmatrix} < 0 \tag{63}$$

$$\begin{bmatrix} \mathscr{N}_S & 0 \\ 0 & I \end{bmatrix}^T \begin{bmatrix} A^T S + SA & SB_1 & C_1^T \\ B_1^T S & -\gamma I & D_{11}^T \\ C_1 & D_{11} & -\gamma I \end{bmatrix} \begin{bmatrix} \mathscr{N}_S & 0 \\ 0 & I \end{bmatrix} < 0 \tag{64}$$

$$\begin{bmatrix} R & I \\ I & S \end{bmatrix} \geq 0 \tag{65}$$

Proof: Our main aim with this Theorem is to express the existence conditions derived in Theorem 1 in terms of plant parameters. We express X_{cl} and X_{cl}^{-1} as follows

$$X_{cl} := \begin{bmatrix} S & N \\ N^T & * \end{bmatrix}; \; X_{cl}^{-1} := \begin{bmatrix} R & M \\ M^T & * \end{bmatrix} \tag{66}$$

where $R, S \in \mathbb{R}^{n\times n}$ and $M, N \in \mathbb{R}^{n\times k}$. Consider Eq.(56) and substitute X_{cl} from Eq.(66). We have,

$$\Phi_{X_{cl}} = \begin{bmatrix} AR + RA^T & AM & B_1 & RC_1^T \\ M^T A^T & 0 & 0 & M^T C_1^T \\ B_1^T & 0 & -\gamma I & D_{11}^T \\ C_1 R & C_1 M & D_{11} & -\gamma I \end{bmatrix} \tag{67}$$

From Eq.(54), we can find out the basis of null space of $\mathscr{P}$ in the following form

$$W_{\mathscr{P}} = \begin{bmatrix} W_1 & 0 \\ 0 & 0 \\ 0 & I_{m_1} \\ W_2 & 0 \end{bmatrix}$$

where

$$\mathscr{N}_R := \begin{bmatrix} W_1 \\ W_2 \end{bmatrix}$$

is any basis of null space of $\begin{bmatrix} B_2^T & D_{12}^T \end{bmatrix}$.
Above can be found out be finding indiviudal elements of $W_{\mathscr{P}}$ that satisfy the following which make $W_{\mathscr{P}}$ the basis of null space of $\mathscr{P}$.

$$\mathscr{P} W_{\mathscr{P}} = 0 \tag{68}$$

Now, the Eq.(56) can be reduced using the above and Eq.(67) to read as given below:

$$\begin{bmatrix} W_1 & 0 \\ 0 & I_{m_1} \\ W_2 & 0 \end{bmatrix}^T \begin{bmatrix} AR + RA^T & B_1 & RC_1^T \\ B_1^T & -\gamma I & D_{11}^T \\ C_1 R & D_{11} & -\gamma I \end{bmatrix} \begin{bmatrix} W_1 & 0 \\ 0 & I_{m_1} \\ W_2 & 0 \end{bmatrix} < 0 \tag{69}$$

Substituting for W_1 and W_2 from above, we can write

$$\begin{bmatrix} \mathscr{N}_R & 0 \\ 0 & I \end{bmatrix}^T \begin{bmatrix} AR + RA^T & RC_1^T & B_1 \\ C_1 R & -\gamma I & D_{11} \\ B_1^T & D_{11}^T & -\gamma I \end{bmatrix} \begin{bmatrix} \mathscr{N}_R & 0 \\ 0 & I \end{bmatrix} < 0 \tag{70}$$

which proves the first statement of the Theorem given in Eq.(63).
Using a similar approach we will next prove the second statement of the Theorem in Eq.(64). Consider Eq.(57), i.e. $W_{\mathscr{D}} \Psi_{X_{cl}} W_{\mathscr{D}} < 0$. We will reduce this equation of Theorem 1 in the terms of plant parameters. From Eq.(66), we can write $\Psi_{X_{cl}}$ as follows:

$$\Psi_{X_{cl}} = \begin{bmatrix} A^T S + SA & A^T N & SB_1 & C_1^T \\ N^T A & 0 & N^T B_1 & 0 \\ B_1^T S & B_1^T N & -\gamma I & D_{11}^T \\ C_1 & 0 & D_{11} & -\gamma I \end{bmatrix} \tag{71}$$

From Eq.(55), we have

$$\mathscr{D} = \begin{bmatrix} 0_{k\times n} & I_{k\times k} & 0_{k\times m_1} & 0_{k\times p_1} \\ C_{2_{p_2\times n}} & 0_{p_2\times k} & D_{21_{p_2\times m_1}} & 0_{p_2\times p_1} \end{bmatrix} \tag{72}$$

Using the above, and using the proper dimensions as mentioned above to solve $\mathscr{D}W_{\mathscr{D}} = 0$ to get $\mathscr{W}_{\mathscr{D}}$, the basis of null space of $\mathscr{D}$. We get

$$W_{\mathscr{D}} = \begin{bmatrix} W_3 & 0 \\ 0 & 0 \\ W_4 & 0 \\ 0 & I \end{bmatrix} \tag{73}$$

where

$$\mathscr{N}_S = \begin{bmatrix} W_3 \\ W_4 \end{bmatrix} \tag{74}$$

is the basis of null space of $\begin{bmatrix} C_2 & D_{21} \end{bmatrix}$. Substituting, we get,

$$\begin{bmatrix} \mathscr{N}_S & 0 \\ 0 & I \end{bmatrix}^T \begin{bmatrix} A^TS + SA & SB_1 & C_1^T \\ B_1^TS & -\gamma I & D_{11}^T \\ C_1 & D_{11} & -\gamma I \end{bmatrix} \begin{bmatrix} \mathscr{N}_S & 0 \\ 0 & I \end{bmatrix} < 0 \tag{75}$$

The third statement in Eq.(65) is clearly seen as $X_{cl} > 0$ and from Eq.(66), we get R and S satisfying Eq.(65). Hence, we have proved the Theorem to obtain the existence conditions for a γ-suboptimal H_∞ controller formulated in LMI. Next, we will use these conditions to derive the conditions for existence for H_∞ loop-shaping design.

9.4.3 Conditions for H_∞ loop-shaping design

For the generalized plant expression obtained for coprime factor uncertainity description to design an H_∞ loop-shaping controller, we can obtain the solvability conditions for existence of a γ-suboptimal controller as follows. From the generalized plant expression in Eq.(41) we have.

$$B_2 = B_{n\times p}; D_{12} = \begin{bmatrix} I_p \\ 0_{m\times p} \end{bmatrix} \tag{76}$$

To find $\mathscr{N}_R$, i.e. the basis of null space of $\begin{bmatrix} B_2^T & D_{12}^T \end{bmatrix}$ we have the rank of null space $= (n+m)$ and $\mathscr{N}_R \in \mathbb{R}^{(n+p+m)\times(n+m)}$. We get

$$\mathscr{N}_R = \begin{bmatrix} I_n & 0_{n\times m} \\ -B_{p\times n}^T & 0_{p\times m} \\ 0_{m\times n} & I_m \end{bmatrix} \tag{77}$$

Similarly, we see that for $\mathscr{N}_S$, the basis of null space of $\begin{bmatrix} C_2 & D_{21} \end{bmatrix}$, we have for H_∞ loop-shaping design from Eq.(41)

$$C_2 = C_{m\times n} \; ; \; D_{21} = I_m \tag{78}$$

The rank of $\mathscr{N}_S \in \mathbb{R}^{(m+n)\times n}$ hence will be equal to n. We can then write

$$\mathscr{N}_S = \begin{bmatrix} I_n \\ -C_{m\times n} \end{bmatrix} \tag{79}$$

9.4.4 LMI Formulation

From $\mathscr{N}_R$ and $\mathscr{N}_S$ obtained above, we can reduce the solvability conditions for the existence of a suboptimal H_∞ loop-shaping controller as three LMIs given below. These three LMIs can be solved to find out the value of γ, R and S which can then be used to design the controller as shown in the following section. The three LMIs which need to be solved for R, S and γ are:

$$\begin{bmatrix} AR + RA^T - \gamma BB^T & RC^T & -L \\ CR & -\gamma I & I \\ -L^T & I & -\gamma I \end{bmatrix} < 0 \tag{80}$$

$$A^T S + SA + C^T L^T S + SLC - \gamma C^T C < 0 \tag{81}$$

$$\begin{bmatrix} R & I \\ I & S \end{bmatrix} \geq 0 \tag{82}$$

Other than the three convex conditions mentioned above, if we want to ensure that the obtained controller is of order k where $k < n$, then another conditions should be satisfied.

$$\text{Rank}(I - RS) \leq k$$

Some examples simulating a few physical systems on MATLAB to solve the above equation are presented in the next section.

9.5 Examples and Simulations

The controller solvability conditions LMIs for a few numerical examples and physical systems have been solved and presented in this section.

9.5.1 A BLDC Hub Motor Driving a Three Wheeled Mobile Robot

Consider a tricycle model of a mobile robot driven by a BLDC hub motor. As an over-simplistic model of the robot, we consider modeling the velocity

control of the BLDC motor only.
Using system identification technique in MATLAB with the data of the input voltage to the motor and the output velocity measurements, the following estimated TF was obtained. We will show that an H_∞ loop-shaping controller design can be done by showing the existence of the γ-suboptimal controller by solving the LMIs developed in previous section. The plant model is given by,

$$G(s) = \frac{0.001011s^2 + 0.0006967s + 1.801 \times 10^{-6}}{s^3 + 1.16s^2 + 0.3351s + 0.007859} \tag{83}$$

To obtain the shaped plant, we choose a first order weight W_1 as follows and $W_2 = 1$ is chosen

$$W_1(s) = \frac{10s + 3}{s(s + 8)} \tag{84}$$

Using the above, we find the following minimal state space representation of the shaped plant. Using this, we solve the LMIs formulated in the previous section

$$A = \begin{bmatrix} -9.1600 & -2.4038 & -0.6722 & -0.0157 & 0 \\ 4.0000 & 0 & 0 & 0 & 0 \\ 0 & 1.0000 & 0 & 0 & 0 \\ 0 & 0 & 1.0000 & 0 & 0 \\ 0 & 0 & 0 & 1.0000 & 0 \end{bmatrix}; \; B = \begin{bmatrix} 0.0625 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \tag{85}$$

$$C = \begin{bmatrix} 0 & 0.0404 & 0.0400 & 0.0084 & 0.0000 \end{bmatrix}; \; D = 0 \tag{86}$$

The valus of γ obtained on solving the LMIs is 1.3937. We take another similar example next.

9.5.2 A Wheeled Mobile Robot System

A second order SISO model (simplified) of a wheeled mobile robot has been derived in [7]. We use the TF model given to check if a γ-suboptimal H_∞ loop-shaping controller exists for the system or not.

$$G(s) = \frac{0.3961}{0.2256s^2 + 0.3645s + 1.469} \tag{87}$$

With the same weights as in previous example, we get the minimal state space realization of the shaped plant as follows

$$A = \begin{bmatrix} -1.6157 & -3.2558 & 2.5000 & 0.7500 \\ 2.0000 & 0 & 0 & 0 \\ 0 & 0 & -8.0000 & 0 \\ 0 & 0 & 1.0000 & 0 \end{bmatrix}; \; B = \begin{bmatrix} 0 \\ 0 \\ 4 \\ 0 \end{bmatrix} \tag{88}$$

$$C = \begin{bmatrix} 0 & 0.8779 & 0 & 0 \end{bmatrix}; \; D = 0 \tag{89}$$

Solving the LMIs we get the value of $\gamma = 1.2761$.

9.5.3 F-16 Fighter Aircraft Model

Finally, we show a MIMO system example. The longitudinal model of a F-16 fighter aircraft is shown below. The state space representation as shown below has been taken from [8] Pages 370-371. For details on definition of states, the equilibrium point about which linearization is done and other details the reader is referred to [8].

$$A = \begin{bmatrix} -0.3220 & 0.0640 & 0.0364 & -0.9917 & 0.0003 & 0.0008 & 0 \\ 0 & 0 & 1 & -0.0037 & 0 & 0 & 0 \\ -30.6492 & 0 & -3.6784 & 0.6646 & -0.7333 & 0.1315 & 0 \\ 8.5396 & 0 & -0.0254 & -0.4764 & -0.0319 & -0.0620 & 0 \\ 0 & 0 & 0 & 0 & -20.2 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -20.2 & 0 \\ 0 & 0 & 0 & 57.2958 & 0 & 0 & -1 \end{bmatrix} \tag{90}$$

$$B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 20.2 & 0 \\ 0 & 20.2 \\ 0 & 0 \end{bmatrix} \tag{91}$$

$$C = \begin{bmatrix} 0 & 0 & 0 & 57.2958 & 0 & 0 & -1 \\ 0 & 0 & 57.2958 & 0 & 0 & 0 & 0 \\ 57.2958 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 57.2958 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}; \; D = [0]_{4\times 2} \tag{92}$$

Solving the LMIs we obtain $\gamma = 3.1587$. With this we move on to controller design at the end of which we will show H_∞ controller design for these physical systems.

9.6 Controller Design

Using the Bounded Real Lemma developed in Section 8.1.2 and solving the corresponding LMI, we can find out the controller by using the values of R and S obtained by solving the controller existence conditions. The steps to reconstruct the controller are given below.

1. Compute two full column rank matrices M and N such that

$$MN^T = I - RS \tag{93}$$

2. From the matrices M and N calculated in the first step, X_{cl} can be found out as follows. Recall that, as defined previously, X_{cl} is a positive definite matrix $\in \mathbb{R}^{(n+k)\times(n+k)}$ which satisfies Eq. (66). Solving the following equation, X_{cl} can be computed.

$$\begin{bmatrix} S & I \\ N^T & 0 \end{bmatrix} = X_{cl} \begin{bmatrix} I & R \\ 0 & M^T \end{bmatrix} \tag{94}$$

Note that Eq.(94) is always solvable when $S > 0$ and M has full column rank.

3. As seen in Theorem 1, from Eq.(60) we can solve the following Bounded Real Lemma inequality:

$$\begin{bmatrix} A_{cl}^T X_{cl} + X_{cl} A_{cl} & X_{cl} B_{cl} & C_{cl}^T \\ B_{cl}^T X_{cl} & -\gamma I & D_{cl}^T \\ C_{cl} & D_{cl} & -\gamma I \end{bmatrix} = \Psi_{X_{cl}} + \mathscr{D}^T \theta^T \mathscr{P}_{X_{cl}} + \mathscr{P}_{X_{cl}}^T \theta \mathscr{D} < 0 \tag{95}$$

4. The above LMI can be solved for θ to obtain the controller parameters.

$$\theta = \begin{bmatrix} A_k & B_k \\ C_k & B_k \end{bmatrix}. \tag{96}$$

Using the method proposed above, controller was reconstructed for the examples of physical systems given above. The step responses for the plant with H_∞ loop-shaping controller design were plotted. Also, bode plots for the controller designed were plotted. Some of the results and plots are shown here.

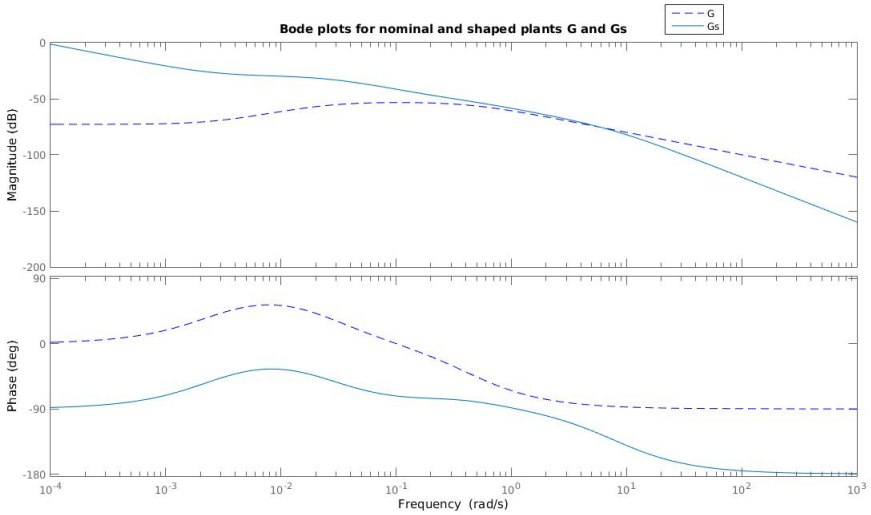

Figure 28: Bode plots for nominal (G) and shaped plants G_s

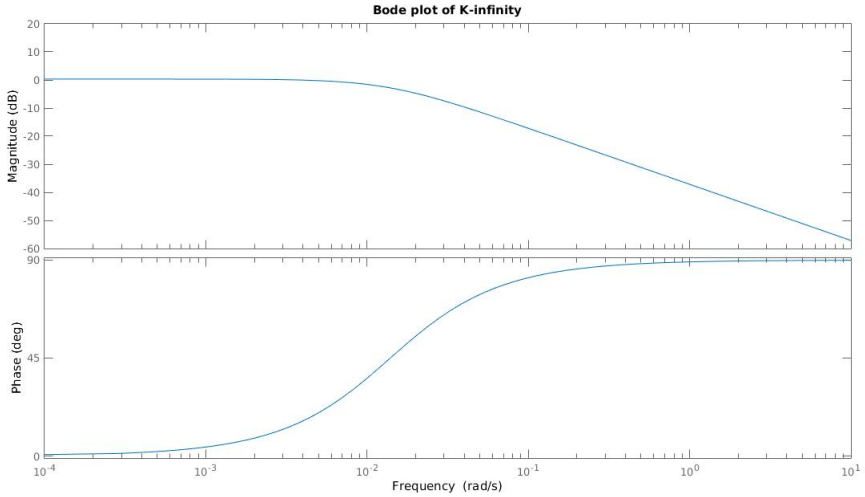

Figure 29: Bode plot for the H_∞ loop-shaping controller designed K_∞

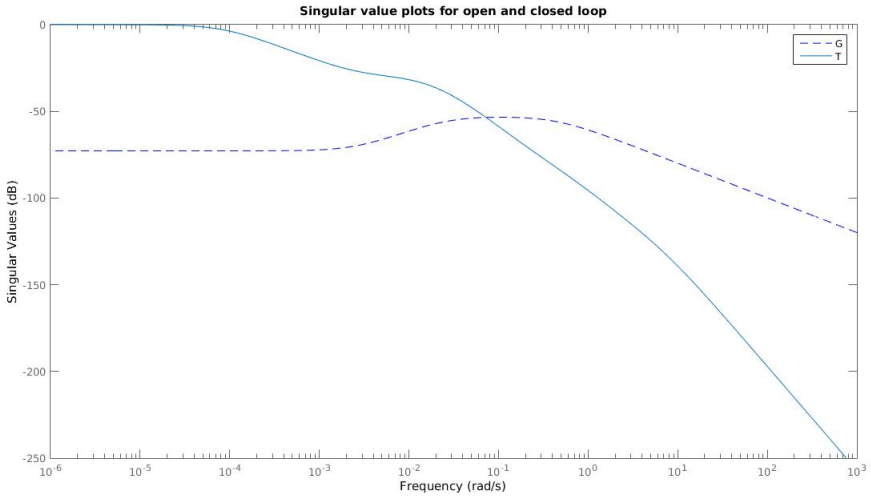

Figure 30: Singular value plot for G and complimentary sensitivity T

10 Conclusion and Future Work

In this project we covered the basics of robust control theory and studied the theory towards general H_∞ control framework. Using which, H_∞ loop-shaping design procedure was studied.
In brief, the work done in this project can be concluded as follows:

1. Various uncertain systems were analyzed for robustness. Simulations for the same on MATLAB were presented.

2. The linear fractional transformation theory which is widely used to represent all systems with uncertainities was studied. Using the LFT framework, many systems were modeled. A common physical system, the spring mass damper, with parametric uncertainities was expressed as an LFT and robustness analysis for the same was done. Subsequently, based on the performance and robust stability specifications, general H_∞ problem was formulated for the spring mass damper system.

3. Controller design was considered using LMI approach which is computationally more efficient than conventional H_∞ controller design methods.

4. An H_∞ loop-shaping controller design procedure using LMI approach was demonstrated. The controller existence conditions were derived and formulated as solvable LMIs. This approach was simulated on MATLAB for various physical systems and the robust stability margins achievable were calculated.

5. A simplified model for speed control of an autonomous robot was derived using MATLAB's system identification technique. H_∞ loop-shaping control design method using LMI approach presented in this project was applied to the simplified TF model. A controller was also designed for this system using the methods proposed by solving LMIs on MATLAB. Simulation results for the same were presented.

The future work in this project would be to analyze the controller designs presented for robust stability and performance. Moreover, the simulations need to be verified by implementation on the real mobile robotic system. Also, as previously mentioned an overly simplified model was considered for the mobile robot by only identifying the TF between voltage input and speed of the BLDC motor. Certainly, there are many more factors that need to be taken care of while modeling the system, which is inherently nonlinear. The controller developed up till now is an LTI robust controller. Hence, to successfully apply this controller to a non-linear system various challenges would need to be overcome.

References

[1] Kemin Zhou, John C. Doyle, and Keith Glover. *Robust and Optimal Control*. Englewood Cliffs, New Jersey: Prentice Hall, 1995.

[2] R. Hyde. "VSTOL first flight on an H_∞ control law". In: (1995).

[3] Da-Wei Gu, Petko H. Petkov, and Mihail M. Konstantinov. *Robust Control Design with MATLAB Second Edition*. London: Springer, 2012.

[4] Pascal Gahinet and Pierre Apkarian. "A linear matrix inequality approach to H_∞ control". In: *International Journal of Robust and Nonlinear Control* 4 (4 1994). DOI: 10.1002/rnc.4590040403. URL: http://gen.lib.rus.ec/scimag/index.php?s=10.1002/rnc.4590040403.

[5] J. Lofberg. "YALMIP : A Toolbox for Modeling and Optimization in MATLAB." In: (2015).

[6] Ayush Pandey. "MATLAB source files". In: (2015). URL: http://github.com/ayush9pandey/robustControl/.

[7] Ahmad A. Mahfouz, Ayman A. Aly, and Farhan A. Salem. "Mechatronics Design of a Mobile Robot System". In: *International Journal of Intelligent Systems and Applications* (2013).

[8] Brian L. Stevens and Frank L. Lewis. *Aircraft Control and Simulation, 2nd Edition*. Wiley, 2003.