

ROBUST AND GAIN-SCHEDULED CONTROL
USING LINEAR MATRIX INEQUALITIES

By
S.Z.Sayed Hassen

SUBMITTED IN PARTIAL FULFILLMENT OF THE
REQUIREMENTS FOR THE DEGREE OF
MASTER OF ENGINEERING SCIENCE
AT
MONASH UNIVERSITY
AUSTRALIA
23RD APRIL 2001

© Copyright

MONASH UNIVERSITY

Date: 23rd April 2001

Author: **S.Z.Sayed Hassen**

Title: **Robust and Gain-Scheduled Control using Linear
Matrix Inequalities**

Department: **Electrical and Computer Systems Engineering**

Degree: **MEngSc**

Permission is herewith granted to Monash University to circulate and to have copied for non-commercial purposes, at its discretion, the above title upon the request of individuals or institutions.

Signature of Author

THE AUTHOR RESERVES OTHER PUBLICATION RIGHTS, AND NEITHER THE THESIS NOR EXTENSIVE EXTRACTS FROM IT MAY BE PRINTED OR OTHERWISE REPRODUCED WITHOUT THE AUTHOR'S WRITTEN PERMISSION.

THE AUTHOR ATTESTS THAT PERMISSION HAS BEEN OBTAINED FOR THE USE OF ANY COPYRIGHTED MATERIAL APPEARING IN THIS THESIS (OTHER THAN BRIEF EXCERPTS REQUIRING ONLY PROPER ACKNOWLEDGEMENT IN SCHOLARLY WRITING) AND THAT ALL SUCH USE IS CLEARLY ACKNOWLEDGED.

To my wife Soraya

and son Tariq

Table of Contents

Table of Contents	iv
List of Tables	vii
List of Figures	viii
List of Symbols	xi
List of Acronyms	xiii
Abstract	xiv
Acknowledgements	xv
I Theoretical Background	1
1 Introduction	2
1.1 Robust and Gain-Scheduled Control	4
1.2 Outline of the Dissertation	5
2 Mathematical Background	8
2.1 Matrix Algebra	8
2.2 Norms	9
2.3 Linear-Fractional Transformations	10
2.4 Linear Matrix Inequalities	11
2.5 Nonlinear Simulation	13
3 Robust Control Theory	15
3.1 Introduction	15

3.2	LQG or \mathcal{H}_2 Controller Design	17
3.3	Model Uncertainty	20
3.4	Performance Specifications and Limitations	23
3.4.1	Weighting Functions	25
3.5	\mathcal{H}_∞ Control	28
3.5.1	The term \mathcal{H}_∞	28
3.5.2	Problem Formulation	29
3.5.3	Linear Matrix Inequality Approach to \mathcal{H}_∞ Control	31
3.5.4	Application of \mathcal{H}_∞ controllers	33
3.6	μ -Synthesis	35
3.6.1	<i>D-K</i> Iteration	36
3.6.2	Advantages and Drawbacks of μ -Synthesis	39
4	Gain-Scheduled Control Theory	41
4.1	Introduction	41
4.2	Polytopic Approach to Gain-Scheduling	44
4.3	LFT Approach to Gain Scheduling using LMI	46
4.3.1	Computation of the Gain-Scheduled Controller	50
4.4	Assessment of Gain-Scheduled Controllers	53
II	Case Studies	55
5	Two-Link Robot Manipulator	56
5.1	Introduction	56
5.2	Dynamics of the Model	58
5.3	Linearized Model of the Two-Link Manipulator	62
5.4	Position Control	63
5.4.1	Mixed-sensitivity \mathcal{H}_∞ control	64
5.4.2	μ -Synthesis using <i>D-K</i> Iteration	73
5.4.3	Gain-Scheduled Controller Design	83
5.4.4	Comparison of Discussed techniques	90
5.5	μ -Controller Reduction	92
6	RTAC Benchmark Problem	96
6.1	Introduction	96
6.2	Problem Statement	96
6.3	\mathcal{H}_∞ Controller Design on the RTAC Model	98
6.4	Gain-Scheduled Control	101

6.4.1	LFT approach to Controller Synthesis	102
7	Missile Model	111
7.1	Introduction	111
7.2	Dynamics of the Model	111
7.3	Gain-Scheduled Control	112
7.3.1	Problem Specification and Synthesis	112
7.3.2	Assessment of LPV Controller	113
8	Conclusions and Further Work	117
8.1	Conclusions	117
8.2	Future Work	119
III	Appendices	121
A	MATLAB Source File	122
B	Non-Singularity of the Manipulator	126
	Bibliography	128

List of Tables

5.1	Parametric values used for simulation	61
5.2	Bounds on parameters for synthesis	86
6.1	Parameter Values	98
6.2	Parameter bounds for controller design	104

List of Figures

2.1	Block diagrams for linear-fractional transformations	11
3.1	The Separation Theorem	19
3.2	Nyquist diagram of an uncertain model	22
3.3	Typical behaviour of multiplicative uncertainty	23
3.4	Feedback Configuration	24
3.5	Desired loop gain	26
3.6	Performance weight W_e and desired S	27
3.7	Robustness weight W_u and desired T	27
3.8	Block diagram	29
3.9	Multiplicative Uncertainty	35
3.10	Standard form for robust analysis and synthesis	37
4.1	Gain-Scheduled \mathcal{H}_∞ problem	45
4.2	LPV control structure	48
5.1	Two-link manipulator	58
5.2	Perturbed System with associated weighting functions	65
5.3	LFT Description of the System	65
5.4	Sensitivity S and W_1^{-1}	68
5.5	Complementary sensitivity T and W_3^{-1}	69
5.6	Singular value plot of the controller K	69

5.7	Step response and control signal of Linear system to a small signal at the point of design $q_2 = 180$ deg with \mathcal{H}_∞ Controller	70
5.8	Small-signal step response and control signal for different q_2 using the same controller	71
5.9	Sensitivity function S for different q_2 using the same controller	72
5.10	Complementary Sensitivity T for different q_2 using the same controller	72
5.11	Nonlinear simulation with \mathcal{H}_∞ Controller	74
5.12	Perturbed System with Multiplicative Uncertainty used for μ -Synthesis	75
5.13	Robust Stability μ plot	76
5.14	Robust Performance μ plot	77
5.15	Worst-Case Performance Degradation	78
5.16	Step response and control signal for worst-case perturbation	78
5.17	Actual Worst-Case Performance Bound at Different q_2	80
5.18	Actual Worst-Case Stability Bound	81
5.19	Sensitivity S and Weighting function W_1^{-1}	81
5.20	Nonlinear simulation of μ -controller when commanded to move from (30,60) to (45,30)	82
5.21	Pre-filtering of control signal	85
5.22	Control structure for gain-scheduled controller design	86
5.23	Augmented plant interconnection	86
5.24	Variation of parameters used for simulation	87
5.25	Step Response of the system with the gain-scheduled controller to a small signal of (0.01,0.01)	88
5.26	Nonlinear simulation with gain-scheduled controller	89
5.27	Comparison of different controllers : \mathcal{H}_∞ (dashed), Robust (dotted), LPV (solid)	91
5.28	Response of nonlinear system with 36 th order controller	93
5.29	Response of nonlinear system with 15 th order controller	94
5.30	Response of nonlinear system with 10 th order controller	95

6.1	Rotational Actuator to control a translational oscillator	97
6.2	σ -plot of the mapping $d \rightarrow [\theta \ z]^T$	99
6.3	System Interconnection for \mathcal{H}_∞ Design	99
6.4	LFT Description for Augmented Plant	100
6.5	Sensitivity S and W_3^{-1}	100
6.6	Impulse Response of the Closed-Loop System	101
6.7	Plot of γ v/s θ	102
6.8	Gain-Scheduled Control of RTAC	103
6.9	LFT Block diagram of the Augmented Plant	103
6.10	Open-loop mapping, closed-loop mapping and W_3^{-1}	105
6.11	Nonlinear simulation to a disturbance $d = \sin 2\pi t$	106
6.12	Nonlinear simulation to a disturbance $d = \sin 12t$	107
6.13	Variation of parameter p_2 to a disturbance $d = \sin 12t$	108
6.14	Nonlinear simulation to an impulse disturbance	109
6.15	Variation of parameter p_2 to an impulse disturbance	110
7.1	Missile Control Autopilot	113
7.2	Parameter trajectories 1 (solid), 2 (dashed)	114
7.3	Nonlinear simulation of missile over trajectories 1 (solid), 2 (dashed) .	115
B.1	Variation of Determinant with angle q_2	127

List of Symbols

$:=$	“is defined to be”
\mathbb{R}	the set of real numbers
\mathbb{R}^n ($\mathbb{R}^{m \times n}$)	the set of real, ordered n -tuples (real, $m \times n$ matrices)
0_n ($0_{m \times n}$)	the zero element of \mathbb{R}^n ($\mathbb{R}^{m \times n}$)
I_n	the identity matrix of size n
$[x, y]$	for vectors $x, y \in \mathbb{R}^n$, the n -dimensional cube $\Pi_{k=1}^n [x_k, y_k]$
$[x_1 \dots x_n]$	the vector $[x_1^T \dots x_n^T]^T \in \mathbb{R}^{m \times n}$ where each $x_k \in \mathbb{R}^m$
$\bar{\sigma}(M)$	largest singular value of M
$\mathcal{R}(M), \mathcal{N}(M)$	the range and nullspace of M
$M > 0$ ($M \geq 0$)	the symmetric matrix M is positive (semi-)definite
$M < 0$ ($M \leq 0$)	the symmetric matrix M is negative (semi-)definite
\mathbb{R}_+	the set of positive real numbers
$\mathbb{R}_+^{n \times n}$	the set of real, symmetric, positive definite $n \times n$ matrices
$M^{1/2}$	the Hermitian square root of M
$\text{diag}(A_1, \dots, A_n)$	a block-diagonal matrix with A_1, \dots, A_n along the diagonal
$\mathcal{F}_u(\cdot, \cdot), \mathcal{F}_l(\cdot, \cdot)$	upper, lower linear-fractional transformations
\mathcal{L}_2^n	the space of square-integrable signals in \mathbb{R}^n
l_2^n	the space of square-summable sequences in \mathbb{R}^n
$ \cdot $	Euclidean norm of a vector
$\langle \cdot, \cdot \rangle$	Euclidean inner product in \mathbb{R}^n , \mathcal{L}_2^n , or l_2^n
$\langle \cdot, \cdot \rangle_L$	inner product for finite-length signals/sequences of length L

$\ \cdot\ _2$	\mathcal{L}_2 or l_2 norm of a signal or sequence
$\ \cdot\ _{i2}$	induced- \mathcal{L}_2 or induced- l_2 norm of a linear system
$\ \cdot\ _\infty$	\mathcal{H}_∞ norm of an LTI system
$\frac{\partial F}{\partial x}$	For $x \in \mathbb{R}^n$ and differentiable $F : \mathbb{R}^n \rightarrow \mathbb{R}^m$, the $m \times n$ Jacobian with $(\frac{\partial F}{\partial x})_{ij} = \frac{\partial F_i}{\partial x_j}$
$\left[\begin{array}{c c} A & B \\ \hline C & D \end{array} \right]$	system-matrix notation for a finite-dimensional system

List of Acronyms

AMI	Affine Matrix Inequality
FDLTI	Frequency Dependent Linear Time Varying
LMI	Linear Matrix Inequality
LPV	Linear Parameter Varying
LTI	Linear Time Invariant
LTV	Linear Time Varying
LFT	Linear Fractional Transformation
LQG	Linear Quadratic Gaussian
LQR	Linear Quadratic Regulator
MIMO	Multi-Input Multi-Output
RMS	Root Mean Square
RTAC	Rotational/Translational Proof-Mass Actuator
SISO	Single-Input Single-Output
SSV	Structured Singular Value
SVD	Singular Value Decomposition

Abstract

This thesis gives an insight into the current trends in robust control theory. It considers the analysis of systems with uncertainties and the design of robust controllers for such systems. We initially overview the linear quadratic gaussian (LQG), \mathcal{H}_∞ and the μ -theories. The drawbacks and advantages of each of the methodologies are presented.

Emphasis is however laid on the more recent work of gain-scheduled controller analysis and design using linear matrix inequalities (LMIs). We specifically look at two approaches to gain-scheduling which we shall call the linear fractional transformation or “LFT” approach and the “polytopic” approach.

The capabilities of each of the design methods are assessed using 3 different case studies : a two-link robot manipulator, a rotational/translational actuator (RTAC) and a missile model. By treating and modelling the nonlinearities as uncertainties, these dynamical systems are treated as linear parameter varying plants. The performance of each of the methods are assessed by exhibiting the closed-loop responses using nonlinear simulations.

Acknowledgements

I would like to thank Francesco Crusca, my supervisor, for his many suggestions and constant support during this research. I would also like to thank Naeem Al-Shamary for introducing me to *WinEdt* and the two anonymous examiners for their constructive remarks.

I am grateful to my wife for her patience and *love*, and to my family for having given me the chance to do further studies.

Finally, I am thankful to Monash University for sponsoring me with a scholarship to do postgraduate studies.

Clayton, Australia
April 23, 2001

S.Z. Sayed Hassen

Part I

Theoretical Background

Chapter 1

Introduction

More than twenty years ago, control theory began to experience a remarkable paradigm shift from optimality to robustness [51]. Theorists were disappointed by results when they applied linear quadratic gaussian (LQG) feedback design theory to realistic problems [5]. The problem was the lack of attention paid to multivariable stability margin. LQG theory, though optimal, had failed to address the fact that changes of loop gains should leave the system with an adequate stability margin [47]. Design based on “pure theory” did not work and there was a need for common sense pragmatic techniques. Even though, it was recognised that there was a lack of modelling accuracy [4], the mathematical concept of uncertainty representation was not there. But all this was to end with the formulation of the multivariable stability margin problem in terms of a matrix with simultaneously varying uncertain real gains [58]. The term “robust” was soon introduced in the control theory [49] and \mathcal{H}_∞ theory came to light in 1988 [16, 62]. Since then, numerous techniques of assessing multivariable stability margins or robustness have emerged amongst which the structured singular value (SSV) $\mu(1/k_m)$ approach is the dominant one [50]. The mixed roles of sensitivity and complementary sensitivity in robust control was also identified [52].

To control a physical system, we today have numerous techniques that we can use at our disposal with each of them having a rich history in literature. Beforehand, we know the advantages and limitations of each method and we choose the best approach for our particular problem. The linear quadratic gaussian(LQG) for example has been found to be not robust to plant variations [17]. \mathcal{H}_∞ control surpasses all the LQG based-design methods and can handle unstructured uncertainties [51]. In [33], it is shown how \mathcal{H}_∞ control can cope with parametric uncertainties and design examples to that effect are given in [46]. Furthermore, robust \mathcal{H}_∞ design for linear uncertain systems with norm-bounded uncertainty in both the state and input matrices is shown in [59]. In practice, it is necessary to include both structured and unstructured uncertainty models so that the composite uncertainty model will generally consist of a variety of structured and unstructured blocks. To cope with such general uncertainty models, other design methodologies are sought after. The μ -synthesis approach, also referred to as *D-K* iteration is able to cope with both structured and unstructured uncertainty. However, there tends to be problems when the uncertainty is real and with the actual computation of the μ value. One has to rely on lower and upper bounds of μ to assess performance and stability. These bounds tend to go further away from each other with the presence of real uncertainty and conservatism is introduced as a result of the usually large operating envelope within which the plant is expected to operate. Given the conservatism present, performance are generally not of the highest standards and controllers of very high order tend to be synthesized which are difficult to implement practically.

This lack of success in dealing with uncertain systems have led to an increasing interest in developing systematic, theoretically rigorous techniques for designing gain-scheduled control systems for time-varying and/or nonlinear plants. And this interest

has stimulated a great deal of research on linear parameter-varying (LPV) systems.

Hence the motivation for this thesis which will focus on robust, gain-scheduled control using linear matrix inequalities (LMI). However, we will also briefly go over the other traditional modern control approaches.

1.1 Robust and Gain-Scheduled Control

One of the most popular methods for applying linear time-invariant (LTI) control theory to time-varying and/or nonlinear systems is gain-scheduling [3]. One of the strategies used involved obtaining linearized dynamic models for the plant at usually finitely many operating points, designing an LTI control law (“point design”) to satisfy local performance objectives for each point, and then adjusting (“scheduling”) the controller gains in real-time as the operating conditions vary. This approach has been applied successfully for many years, particularly for aircraft and process control problems. Despite these successes of gain-scheduling in practice, little has been known about it theoretically as a time-varying and nonlinear control technique. Determining the actual scheduling routine is more of an art than a science. More rigorous design and analysis techniques are needed.

The current methods are based on and extend state-space approaches to \mathcal{H}_∞ for LTI systems [22]. The resulting parameter-dependent controllers are also scheduled automatically; the often arduous task of scheduling a complex multivariable controller *a posteriori* is avoided. These methods rely on LMIs for computing controllers, characterizing performance, and/or determining the solvability of design problems. Many problems involving LMIs have been found to arise in control theory [11]. Moreover,

efficient algorithms have been developed for using convex programming to solve feasibility and optimization problems involving LMIs [23]. One of the research paths that has developed involves LPV systems and controllers whose parameter dependence can be expressed as linear-fractional transformations (LFTs) [18]. The approach relies on scaled small-gain methods and scaled \mathcal{H}_∞ optimization to design parameter-dependent controllers that also resemble LFTs and provide closed-loop stability and induced- \mathcal{L}_2 performance [1, 2, 43]. Another approach in this same line of thought which we shall call the “polytopic” approach is based on [3]. A major part of this thesis is devoted to these two gain-scheduled approaches whereby we will fully describe these two approaches and we will investigate the capabilities of both through Case Studies.

1.2 Outline of the Dissertation

This thesis is organized as follows :

The thesis is divided into primarily two parts. Part I which consists of Chapters 2, 3 and 4 presents the theoretical background behind robust control and gain scheduling. Part II looks into the design examples and these are presented in Chapters 5, 6 and 7. Finally we have added a section for the Appendices. Let us now present each chapter in more detail.

In Chapter 2, we present some elementary mathematical definitions and results that are used in later chapters. In particular, we describe in detail the simulation method used for all the nonlinear models we looked at.

In Chapter 3, we present an overview of how robust control theory arose and we look into the main design methodologies of modern robust control theory. We briefly

describe the modelling aspect of plants and discuss in detail the various kinds of uncertainty existent in any physical model and how they are modelled for controller synthesis purposes. We also look into the limitations and the trade-off always present between performance and robustness and give brief results about how to come up with weighting functions representative of the performance specifications and of the robustness desired from the problem. The LQG methods of controller synthesis is explained quite thoroughly and we mention its potential drawbacks. We look into the \mathcal{H}_∞ design method, more specifically at the LMI approach to \mathcal{H}_∞ synthesis and the virtues and limitations of the method are assessed. Finally, we turn our attention to the μ -Synthesis method and talk about its applicability to real-life problems where structured uncertainties are common occurrence. The $D-K$ iteration method is addressed quite thoroughly and we mention the drawbacks of the method with respect to the accuracy of the results obtained with the present software tools when dealing with real and/or repeated uncertainties present in the model and also with respect to the very high order controllers achieved using this method.

In Chapter 4, we present a detailed discussion of gain-scheduling and talk about its capability to excel in terms of performance when implemented properly. Specifically, we look at the “polytopic” and “LFT” approach of gain-scheduled controller synthesis and provide simple mathematical solutions for all the computations needed for the design. Finally we make a few comments about the assessment of their performance and stability.

In Chapter 5, we look at a two-link robot manipulator model as a case-study and apply all the techniques learned before and perform a thorough analysis of their performances. Extensive nonlinear simulations are performed and finally we compare

the different methods and briefly investigate Model Reduction in the context of this example.

In Chapter 6, the RTAC model [12] is presented in detail. We design an \mathcal{H}_∞ controller for the linearized model at a fixed parameter value and finally design a gain-scheduled controller for the nonlinear model using the LFT approach. Nonlinear simulations follow for different disturbance conditions.

In Chapter 7, we look into a simple missile model that has been studied extensively. As a first though, we synthesize a gain-scheduled controller for the model using the LFT approach and we show that the controller performs extremely well even in the face of abrupt parameter changes. We also show that the conservativeness of the LFT approach pays off when the parameter changes are quick and rapid through nonlinear simulation.

Chapter 8 provides a summary of the results, some concluding remarks and avenues for future work.

Appendix A contains the MATLAB source code for the synthesis of gain-scheduled LFT controllers. These codes require the LMI TOOLBOX [23] under MATLAB to be executed.

Appendix B presents a very simple analysis showing that the two-link manipulator model discussed in Chapter 5 is non-singular.

Chapter 2

Mathematical Background

In this chapter, the mathematical tools that are used extensively in the remainder of the thesis are presented. A good reference for the results on matrix algebra (Section 2.1) and norms (Section 2.2) is [26].

2.1 Matrix Algebra

Definition 2.1.1. Let $x, y \in \mathbb{R}^n$ be real vectors. The *Euclidean inner product* of x and y and *Euclidean norm* of x are respectively defined as $\langle x, y \rangle := x^T y$ and $|x| := \sqrt{\langle x, x \rangle} = \sqrt{x^T x}$.

Definition 2.1.2 (Singular Value Decomposition). The singular value decomposition or SVD of a given matrix $M \in \mathbb{R}^{m \times n}$ having rank k is

$$M = U\Sigma V^T = [U_1 \quad U_2] \begin{bmatrix} \Sigma_1 & 0 \\ 0 & 0 \end{bmatrix} \begin{bmatrix} V_1^T \\ V_2^T \end{bmatrix} \quad (2.1.1)$$

where the matrices $U \in \mathbb{R}^{m \times m}$ and $V \in \mathbb{R}^{n \times n}$ are orthogonal (i.e., $U^T U = U U^T = I_m$ and $V^T V = V V^T = I_n$), $\Sigma_1 = \text{diag}(\sigma_1, \dots, \sigma_k)$ and $\sigma_1 \geq \dots \geq \sigma_k > 0$. The scalars $\sigma_1, \dots, \sigma_{\min\{m,n\}}$ (where $\sigma_{k+1} = \dots = \sigma_{\min\{m,n\}}$) are called the *singular values* of M . The largest singular value σ_1 , also denoted by $\bar{\sigma}(M)$, equals to the induced 2-norm

$$\bar{\sigma}(M) = \|M\|_{i2} := \max_{\substack{x \in \mathbb{R}^n \\ x \neq 0}} \frac{|Mx|}{|x|} \quad (2.1.2)$$

of M . The *condition number* of M is denoted by $\kappa(M) := \sigma_1/\sigma_k$. The columns of U_1, U_2, V_1 and V_2 form respective orthonormal bases for $\mathcal{R}(M)$, $\mathcal{N}(M^T)$, $\mathcal{R}(M^T)$, and $\mathcal{N}(M)$.

Definition 2.1.3. Let the matrix $M \in \mathbb{R}^{n \times n}$ be symmetric (i.e., $M = M^T$). We say that:

M is positive definite ($M > 0$) if $x^T M x > 0$ for all nonzero $x \in \mathbb{R}^n$

M is positive semi-definite ($M \geq 0$) if $x^T M x \geq 0$ for all $x \in \mathbb{R}^n$

M is negative definite ($M < 0$) if $x^T M x < 0$ for all nonzero $x \in \mathbb{R}^n$

M is negative semi-definite ($M \leq 0$) if $x^T M x \leq 0$ for all $x \in \mathbb{R}^n$

If a symmetric matrix M is block-diagonal, then it is positive/negative (semi-)definite if and only if each block is. If $M \geq 0$, then its singular value and eigenvalue decompositions are identical: the eigenvalues of M are $\sigma_1, \dots, \sigma_n$, and the columns of $U = V$ are the corresponding(orthonormal) eigenvectors; we can then define the matrix square root as $M^{1/2} := U^T \Sigma^{1/2} U$, where $\Sigma^{1/2} = \text{diag}(\sqrt{\sigma_1}, \dots, \sqrt{\sigma_n})$.

Lemma 2.1.1 (Matrix Inversion Lemma). For appropriately dimensioned matrices A, B, C and D ,

$$(A \pm BCD)^{-1} = A^{-1} \mp A^{-1}B(C^{-1} \pm DA^{-1}B)^{-1}DA^{-1}$$

(assuming that A and C are invertible).

Lemma 2.1.2. If the block-partitioned matrix $M = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}$ is invertible, then its inverse can be written as $M^{-1} = \begin{bmatrix} P_{11} & P_{12} \\ P_{21} & P_{22} \end{bmatrix}$, where

$$\begin{aligned} P_{11} &= (M_{11} - M_{12}M_{22}^{-1}M_{21})^{-1} \\ P_{12} &= -M_{11}^{-1}M_{12}(M_{22} - M_{21}M_{11}^{-1}M_{12})^{-1} \\ P_{21} &= -M_{22}^{-1}M_{21}(M_{11} - M_{12}M_{22}^{-1}M_{21})^{-1} \\ P_{22} &= (M_{22} - M_{21}M_{11}^{-1}M_{12})^{-1} \end{aligned}$$

2.2 Norms

We here present some definitions for induced matrix and system norms.

Definition 2.2.1 (Induced Matrix Norm). Using the Euclidean vector norm defined previously, the *induced matrix norm* for a matrix $M \in \mathbb{R}^{m \times n}$ is defined by

$$\|M\| \triangleq \sup_{x \neq 0} \frac{|Mx|}{|x|}$$

With this definition, it can be shown [26] that $\|M\| = \bar{\sigma}(M)$, where $\bar{\sigma}$ denotes the maximum singular value of M .

Definition 2.2.2 (L_p Space). *Given a continuous-time signal $x : \mathbb{R} \rightarrow \mathbb{R}^n$, x is said to be in L_p^n (or simply L_p) if*

$$\int_0^\infty \sum_{i=1}^n |x_i(t)|^p dt < \infty$$

and the L_p -norm of x is defined for $1 \leq p < \infty$ as

$$\|x\|_p \triangleq \left\{ \int_0^\infty \sum_{i=1}^n |x_i(t)|^p dt \right\}^{\frac{1}{p}}$$

In particular, the L_2 -norm of $x(t)$ is given by

$$\|x\|_2 = \left\{ \int_0^\infty x(t)^T x(t) dt \right\}^{\frac{1}{2}}$$

and the L_∞ -norm of x is

$$\|x\|_\infty = \max_{t \geq 0} \sup |x(t)|$$

Definition 2.2.3 (Induced L_2 -norm). *Consider a continuous-time operator $G : L_2^n \rightarrow L_2^m$. The induced L_2 -norm of G is defined as*

$$\|G\|_{i,2} \triangleq \sup_{\substack{u \in L_2^n \\ u \neq 0}} \frac{\|Gu\|_2}{\|u\|_2}$$

For a stable, rational and proper transfer matrix $G(s)$, the induced L_2 -norm of G is equal to the ∞ -norm of G , where

$$\|G\|_\infty \triangleq \sup_{w \in \mathbb{R}} \bar{\sigma}(G(jw))$$

2.3 Linear-Fractional Transformations

In this section, we introduce linear-fractional transformations (LFTs) and use them to provide elementary results regarding feedback interconnections of matrices or systems.

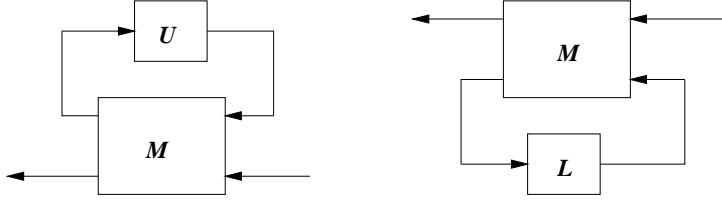


Figure 2.1: Block diagrams for linear-fractional transformations

Definition 2.3.1 (Linear-Fractional Transformation). *Appropriately partition the matrices $M \in \mathbb{R}^{(m_1+m_2) \times (n_1+n_2)}$ as*

$$M = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}$$

Let $U \in \mathbb{R}^{n_1 \times m_1}$ and $L \in \mathbb{R}^{n_2 \times m_2}$. The linear fractional transformations $\mathcal{F}_u(M, U)$ and $\mathcal{F}_l(M, L)$ are given as follows. Assuming that $(I_{m_1} - M_{11}U)$ and $(I_{m_2} - M_{22}L)$ are invertible,

$$\begin{aligned} \mathcal{F}_u(M, U) &:= M_{22} + M_{21}U(I_{m_1} - M_{11}U)^{-1}M_{12} \\ \mathcal{F}_l(M, L) &:= M_{11} + M_{12}L(I_{m_2} - M_{22}L)^{-1}M_{21} \end{aligned}$$

These operations are depicted in Figure 2.1. One of the attractive properties of linear-fractional transformations is that sums, products, inversions and block augmentations of LFTs also are LFTs [18].

2.4 Linear Matrix Inequalities

Most of the results in this thesis are centred around Linear Matrix Inequalities (LMIs).

We now give a brief introduction to LMIs. A very comprehensive study of LMIs and their use can be found in [11, 45].

Given matrices $F_0 = F_0^T \in \mathbb{R}^{n \times n}$, $F_i = F_i^T \in \mathbb{R}^{n \times n}$, for $i = 1, \dots, m$, an LMI is a constraint of the form

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

where the x_i are unknown scalar variables. The LMI feasibility problem is one of finding any x_i such that $F(x) > 0$.

An LMI defines a convex constraint on x , so that effective solution techniques such as ellipsoid, interior-point or projective algorithms can be applied with guaranteed convergence to a global minimum [11].

Several problems in control theory can be cast as LMIs. An immediate example is the stability analysis of the linear time-invariant system

$$\dot{x} = Ax$$

where $x \in \mathbb{R}^n$. It is well-known that the above system is asymptotically stable if and only if there exists a matrix $P = P^T \in \mathbb{R}^{n \times n}$ such that

$$A^T P + PA < 0 \text{ and } P > 0$$

Defining $\{P_1, \dots, P_{\frac{n(n+1)}{2}}\}$ as a basis¹ for the space of symmetric matrices in $\mathbb{R}^{n \times n}$ and representing the $\frac{n(n+1)}{2}$ independent entries in P as p_i , we can rewrite the stability condition as

$$\sum_{i=1}^{\frac{n(n+1)}{2}} p_i (A^T P_i + P_i A) < 0 \text{ and } \sum_{i=1}^{\frac{n(n+1)}{2}} p_i P_i > 0$$

which clearly conforms to the definition of the LMI feasibility problem. Thus, any matrix inequality where the unknown matrices appear affinely define LMIs in terms of the entries of the variable matrices. Software for solving LMIs are commercially available, with possibly the most widely used package being [23].

¹e.g. for $n = 2$, a sample basis $\{P_1, P_2, P_3\}$ is :

$$P_1 = \begin{bmatrix} 1 & 0 \\ 0 & 0 \end{bmatrix}, P_2 = \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix}, P_3 = \begin{bmatrix} 0 & 0 \\ 0 & 1 \end{bmatrix}$$

2.5 Nonlinear Simulation

For nonlinear simulation results throughout this thesis, the trapezoidal rule is used to ensure the convergence of the states, x , of the open-loop plant or of the plant and controller if simulating a closed-loop system. The first step for the simulation is to split the simulation time evenly into very small time frames T and then the following technique is used :

By trapezoidal rule,

$$x_{k+1} = x_k + \frac{1}{2}T(\dot{x}_k + \dot{x}_{k+1}) \quad (2.5.1)$$

where

- x_k is the present state of plant/controller or both
- x_{k+1} is the next state of plant/controller or both
- T is the iteration step size.

For any general plant, the state space equations are

$$\dot{x} = Ax + Bu \quad (2.5.2)$$

Substituting (2.5.2) into (2.5.1) leads to

$$\begin{aligned} x_{k+1} &= x_k + \frac{1}{2}T \{ A(x_k) x_k + B_k u_k \\ &\quad + A(x_{k+1}) x_{k+1} + B_{k+1} u_{k+1} \} \end{aligned} \quad (2.5.3)$$

Equation 2.5.3 is then implemented in a computer program to solve for x_{k+1} . The first step is to assume that x_k and x_{k+1} are equal. Then the program is run for a new value of x_{k+1} . This new value of x_{k+1} is then compared with the previous x_{k+1} . If the new x_{k+1} is greater or less than the last x_{k+1} by more than a certain desired convergence limit (usually 10^{-8}), then this new value of x_{k+1} is used together with

the previous x_k to compute a new x_{k+1} . This process is then repeated until the last and the new x_{k+1} are within the convergence limit. Then the new state x_{k+1} is stored. The process is then repeated taking the recently stored x_{k+1} obtained as the present state. In other words, x_{k+1} is assigned to x_k . And the whole process begins again. This is done until the final time is reached at which the iteration process stops. This completes the simulation.

Chapter 3

Robust Control Theory

3.1 Introduction

Feedback control is well understood for large classes of nonlinear systems with single inputs. For general multi-input nonlinear systems, however, feedback control and especially robustness issues are still research topics, the urgency of which has been rendered more acute by the recent development of machines with challenging nonlinear dynamics, such as robot manipulators, high-performance aircraft, or advanced underwater and space vehicles.

The basis for control design and stability analysis is a dynamical model that captures prominent features of the system under consideration. To account for unnoticeable and unknown aspects of the real system in the mathematical model, one often uses the notion of uncertainty. Uncertainty denotes any obscure element in the dynamics of the real system. Possible uncertainties include unknown parameters, unknown functions, disturbances, and unmodeled dynamics. In general uncertainties can be either stochastic or deterministic and control design and performance analysis must be done accordingly. Uncertainties can also be classified as either “structured” or “unstructured”.

Structured Uncertainty represents parametric variation in the plant dynamics, for example :

- Uncertainties in certain entries of state-space matrices (A, B, C, D).
- Uncertainties in specific poles and/or zeros of the plant transfer function
- Uncertainties in specific loop gains/phases.

Unstructured uncertainty may be used to represent frequency-dependent elements such as actuator saturations and unmodeled structural modes in the high frequency range or plant disturbances in the low frequency range.

For a robotic system for example, system parameters and payload can be viewed as structured uncertainties; unstructured uncertainties include friction, disturbances, and unmodeled dynamics.

The challenge in robust multi-variable feedback control system design is to synthesize a control law which maintains system response and error signals to within pre-specified tolerances despite the effects of uncertainty on the system. Depending on the nature of the uncertainties, different designs can be used to achieve effective control.

Before uncertainties were recognized as important enough to be part of the design process, the main multivariable method was LQG. With the birth of robust control in the late seventies though, \mathcal{H}_∞ and μ proved to be superior in terms of their robustness [17, 51]. We talk about these in Section 3.5 and Section 3.6 respectively. In the next section, we shall briefly look at the LQG approach.

3.2 LQG or \mathcal{H}_2 Controller Design

Optimal Control, building on the optimal filtering work of Weiner in the 1940's, reached maturity in the 1960's with what we now call Linear Quadratic Gaussian or LQG control [51, 54, 60]. In traditional LQG control, it is assumed that the plant dynamics are linear and known, and that the measurement noise and disturbance signals (process noise) are stochastic with known statistical properties. That is, we have a plant model

$$\dot{x} = Ax + Bu + w_d \quad (3.2.1)$$

$$y = Cx + w_n \quad (3.2.2)$$

where w_d and w_n are the disturbance(process) and measurement noise inputs respectively, which are usually assumed to be uncorrelated zero-mean Gaussian stochastic processes with constant power spectral density matrices W and V respectively. That is, w_d and w_n are white noise processes with covariances

$$Ew_d(t)w_d(\tau)^T = W\delta(t - \tau) \quad (3.2.3)$$

$$Ew_n(t)w_n(\tau)^T = V\delta(t - \tau) \quad (3.2.4)$$

and

$$Ew_d(t)w_n(\tau)^T = 0, \quad Ew_n(t)w_d(\tau)^T = 0$$

where E is the expectation operator and $\delta(t)$ is the delta function.

The LQG control problem is to find the optimal control $u(t)$ which minimizes

$$J = E \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T [x^T Q x + u^T R u] dt$$

where Q and R are appropriately chosen constant weighting matrices (design parameters) such that $Q = Q^T \geq 0$ and $R = R^T > 0$. The weighting matrices are selected on

the basis of several sets of criteria. The physics of the problem at hand may in some cases suggest terms in the cost function. For example, we might want to minimize the kinetic energy or potential energy of a pendulum while at the same time minimize the energy used by the controller in doing so. Q would then select certain states to represent the kinetic/potential energy and R would take care of the control energy.

The name LQG comes from the use of a linear model, an integral Quadratic cost function, and Gaussian white noise processes to model disturbance signals and noise. The solution to the LQG problem, known as the Separation Theorem or Certainty Equivalence Principle, is surprisingly simple and elegant. It consists of first determining the optimal control to a deterministic linear quadratic regulator (LQR) problem. LQR forms a component within the LQG controller, its major limitations being that the entire state must be measured. The LQR problem is then the LQG problem without w_d and w_n and with $y = x$. It happens that the solution to this problem can be written in terms of the simple state feedback law

$$u(t) = -K_r x(t)$$

where K_r is a constant matrix which is easy to compute and is independent of W and V , the statistical properties of the plant noise. The next step is then to find a way of getting around measuring all the states as required by LQR. This can be very expensive especially with high order MIMO systems. A logical step is to find an optimal estimate \hat{x} of the state x , that minimizes the mean square estimation error or $E[x - \hat{x}]^T[x - \hat{x}]$. The optimal state estimate is given by a Kalman filter and is independent of Q and R . The required solution to the LQG problem is then found by replacing x by \hat{x} , to give $u(t) = -K_r \hat{x}(t)$. The LQG problem and its solution can be separated into two distinct parts as illustrated in Figure 3.1.

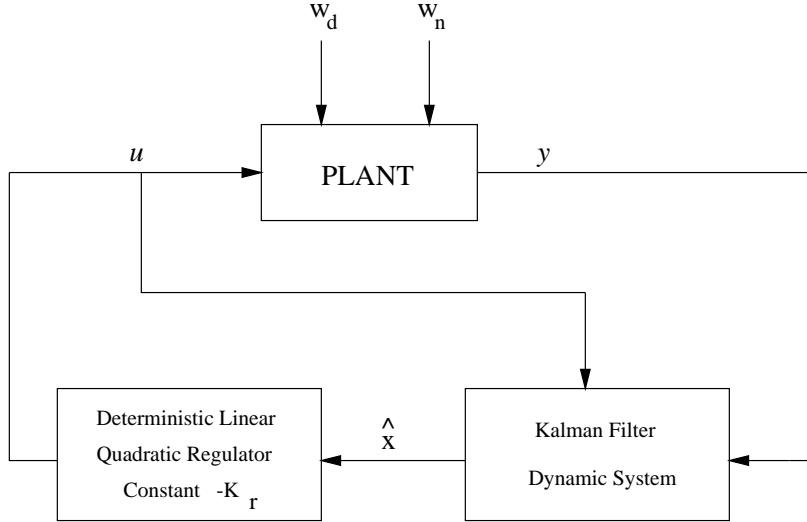


Figure 3.1: The Separation Theorem

The robustness of each of the LQR and the Kalman filter subject to multiplicative perturbations, respectively, is very good. Unfortunately, this robustness may be destroyed when combining the LQR and the Kalman filter. In general, there are no guarantees on the robustness of the LQG optimal control, and the robustness of each design should be carefully checked [15, 17, 63].

The LQG optimal control problem is equivalent to an \mathcal{H}_2 optimal control problem. This \mathcal{H}_2 optimization problem is formulated with the output equal to the combination of $Q^{1/2}$ times the state and $R^{1/2}$ times the control. The \mathcal{H}_2 optimal control problem is as follows: Find a feedback controller that internally stabilizes the closed-loop system and also minimizes the closed-loop system \mathcal{H}_2 -norm. Thinking of the LQG in terms of \mathcal{H}_2 optimization is useful in generalizing the LQG to include frequency-domain performance specifications. The \mathcal{H}_2 formulation of the LQG problem also indicates the possibility of generating alternative controller designs by using different system norms. The idea is pursued in the Section 3.5, where \mathcal{H}_∞ -norm optimization is presented.

3.3 Model Uncertainty

Most control designs are based on the use of a design model. The relationship between models and the reality they represent is subtle and complex. A mathematical model provides a map from inputs to responses. The quality of a model depends on how closely its responses match those of a true plant. Since no fixed model can respond exactly like the true plant, we need, at the very least, a set of maps. However, the modeling problem is much deeper - the universe of mathematical models from which a model set is chosen is distinct from the universe of physical systems. Therefore, a model set that includes the true physical plant can rarely be constructed. It is necessary for the engineer to make a leap of faith regarding the applicability of a particular design based on a mathematical model. To be practical, a design technique must help make this leap small by accounting for the inevitable inadequacy of models. A good model should be simple enough to facilitate design, yet complex enough to give the engineer confidence that designs based on the model will work on the true plant.

The term “uncertainty” refers to the differences or errors between models and reality. Representations of uncertainty vary primarily in terms of the amount of uncertainty they contain. This reflects both our knowledge of the physical mechanisms that cause differences between the model and the plant and our ability to represent these mechanisms in a way that facilitates convenient manipulation. For example, a set membership statement for the parameters of an otherwise known frequency dependent linear time invariant (FDLTI) model is a highly structured representation of uncertainty. It typically arises from the use of linear incremental models at various operating points (e.g., aerodynamical coefficients in flight control vary with flight

environment and aircraft configurations). The amounts of variation and any known relationships between parameters can be expressed by confining the parameters to appropriately defined subsets of parameter space. However, for certain classes of signals (e.g., high-frequency), the parameterized FDLTI model fails to describe the plant because the plant will always have dynamics that are not represented in the fixed order model. In general, we are forced to use not just a single parameterized model but model sets that allow for plant dynamics that are not explicitly represented in the model structure. A simple example of this involves using frequency domain bounds on transfer functions to describe a model set. To use such sets to describe physical systems, the bounds must roughly grow with frequency. This gives a less structured representation of uncertainty.

Examples of less structured representations of uncertainty are direct set membership statements for the transfer function matrix of the model. For instance, the statement

$$P_\Delta(s) = P(s) + W_1(s)\Delta(s)W_2(s), \quad \bar{\sigma}[\Delta(j\omega)] < 1, \quad \forall \omega \geq 0, \quad (3.3.1)$$

where W_1 and W_2 are stable transfer matrices that characterize the spatial and frequency structure of the uncertainty, confines the P_Δ to a neighbourhood of the nominal model P . In particular, if $W_1 = I$ and $W_2 = w(s)I$, where $w(s)$ is a scalar function, then P_Δ describes a disk centered at P with radius $w(j\omega)$ at each frequency as shown in Figure 3.2. The uncertainty may be caused by parameter changes, as mentioned previously or by neglected dynamics, or by a host of other unspecified effects. An alternative statement to (3.3.1) is the multiplicative form:

$$P_\Delta(s) = (I + W_1(s)\Delta(s)W_2(s))P(s). \quad (3.3.2)$$

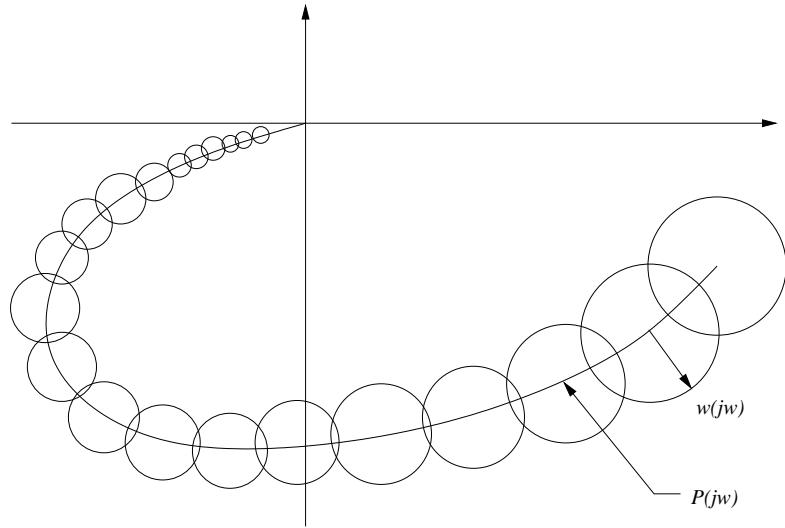


Figure 3.2: Nyquist diagram of an uncertain model

This statement confines P_Δ to a normalized neighbourhood of the nominal model P . When used to represent the various high-frequency mechanisms, the weighting function in (3.3.2) commonly has the properties illustrated in Figure 3.3. They are small ($\ll 1$) at low frequencies and increase to unity and above at higher frequencies. The best choice of uncertainty representation for a specific FDLTI model depends, of course, on the errors in the model. It is generally possible to represent some of these errors in a highly structured parameterized form. These are usually the low-frequency error components. There are always higher-frequency errors, however, which cannot be covered this way. The less structured representations, such as (3.3.1) and (3.3.2), are well suited to represent this latter class of errors. Consequently, (3.3.1) and (3.3.2) have become widely used “generic” uncertainty representations for FDLTI models. An important point is that the construction of the weighting functions W_1 and W_2 for multi-variable systems is not trivial and this is what we are going to look into in the next section.

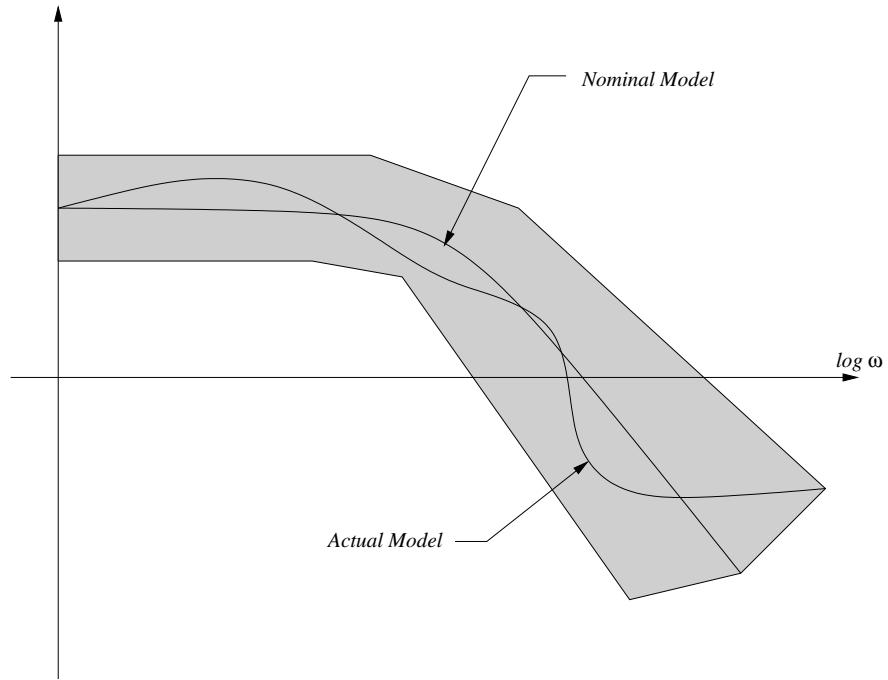


Figure 3.3: Typical behaviour of multiplicative uncertainty

3.4 Performance Specifications and Limitations

Consider the feedback system shown in Figure 3.4 where

- r : reference input,
- d : disturbance,
- n : noise,
- u : control signal, and
- y : measurements.

Clearly,

$$\begin{aligned} y &= G u + G_d d \\ u &= K (r - y - n) \\ \therefore y &= GK (r - y - n) + G_d d \end{aligned}$$

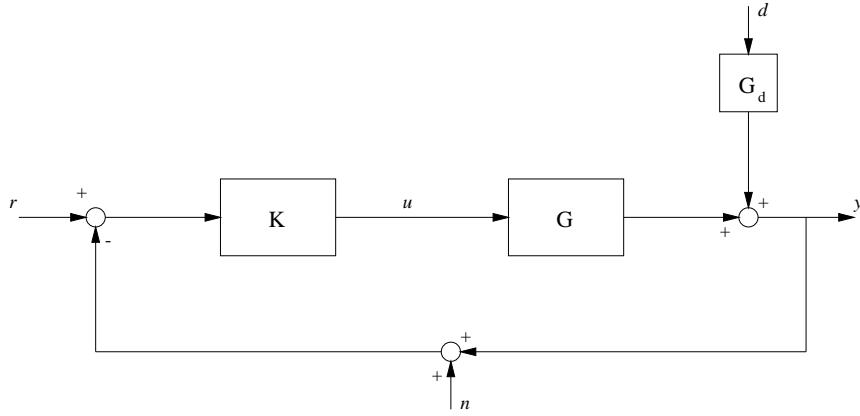


Figure 3.4: Feedback Configuration

and hence the closed-loop response is

$$y = \underbrace{(I + GK)^{-1}GK}_T r + \underbrace{(I + GK)^{-1}G_d}_S d - \underbrace{(I + GK)^{-1}GK}_T n \quad (3.4.1)$$

For perfect control, we want

$$\begin{aligned} e &= y - r = 0 \\ e &\approx 0 \cdot d + 0 \cdot r + 0 \cdot n \end{aligned} \quad (3.4.2)$$

The first two requirements in (3.4.2), namely disturbance rejection and command tracking, are obtained with $|S| \approx 0$, or equivalently, $|T| \approx 1$. Since $S = (I + L)^{-1}$, where $L = GK$, this implies that the loop transfer function L must be large in magnitude. On the other hand, the requirement for zero noise transmission implies that $|T| \approx 0$, or equivalently, $|S| \approx 1$, which is obtained with $|L| \approx 0$. This illustrates the fundamental nature of feedback design which always involves a trade-off between conflicting objectives; in this case between large loop gains for disturbance rejection and tracking, and small loop gains to reduce the effect of noise. Large $\bar{\sigma}(L)$ values over a large frequency range makes errors due to d small. However, they also make

errors due to n large because this noise is passed through over the same frequency range. Fortunately, d and n are typically significant in different frequency ranges.

Also, a large loop gain outside the bandwidth of G , that is, $\bar{\sigma}(L) \gg 1$ while $\bar{\sigma}(G) \ll 1$ can make the control activity u quite unacceptable, causing the saturation of the actuators. For good performance, we must also consider the speed of the response and this leads to considering the bandwidth of the system. In general, a large bandwidth corresponds to a faster rise time, since high frequency signals are more easily passed onto the outputs. A high bandwidth also indicates a system which is sensitive to noise and to parameter variations. Conversely, if the bandwidth is small, the time response will generally be slow, and the system will usually be more robust.

Another major performance tradeoff concerns commands and disturbance error rejection versus stability under the model uncertainty. Let us say that the plant model is perturbed to $(I + \Delta)G$ with Δ stable, and that the system is nominally stable (i.e., the closed-loop system with $\Delta = 0$ is stable). The perturbed closed-loop system is stable if

$$\det(I + (I + \Delta)GK) = \det(I + GK) \det(I + \Delta T)$$

has no right-half plane zeros [60]. This would generally require that $\bar{\sigma}(T)$ be small at those frequencies where Δ is significant, typically at high frequency range, which in turn implies that the loop gain, $\bar{\sigma}(L)$, should be small at those frequencies. These design requirements are shown graphically in Figure 3.5.

3.4.1 Weighting Functions

The selection of weighting functions is clearly a very important step in any control design. They are the carriers of information about the performance and robustness

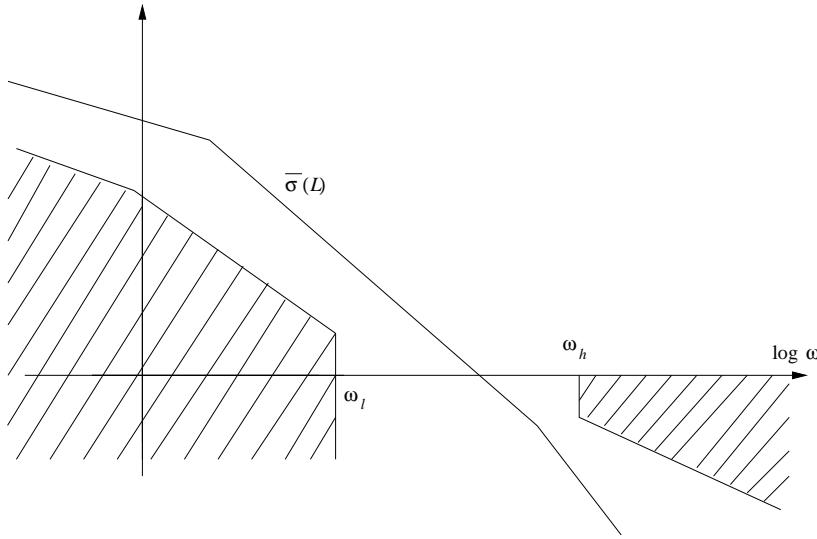


Figure 3.5: Desired loop gain

desired in the design. The selection of weighting functions for a specific design problem is not a very straightforward exercise. It often involves ad hoc fixing, many iterations, and fine tuning. It is very hard to give a general formula for the weighting functions that will work in every case and we will only give a final general formula here. For a detailed explanation of the reasoning behind it, the reader is referred to [60]. Referring to Figure 3.6, weight W_e can be selected as:

$$W_e = \left(\frac{s/\sqrt[k]{M_s} + \omega_b}{s + \omega_b \sqrt[k]{\varepsilon}} \right)^k$$

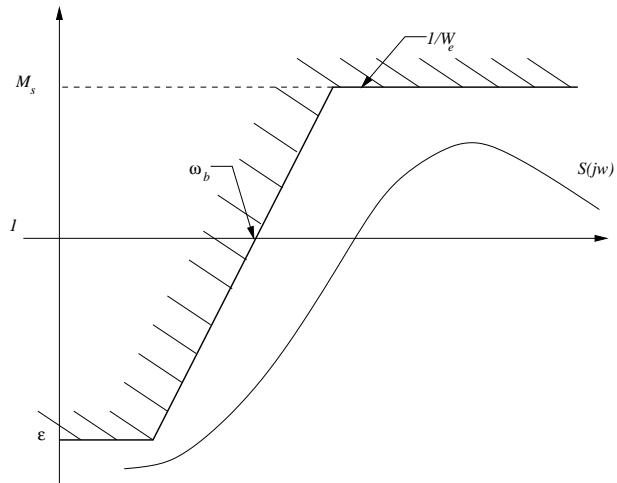
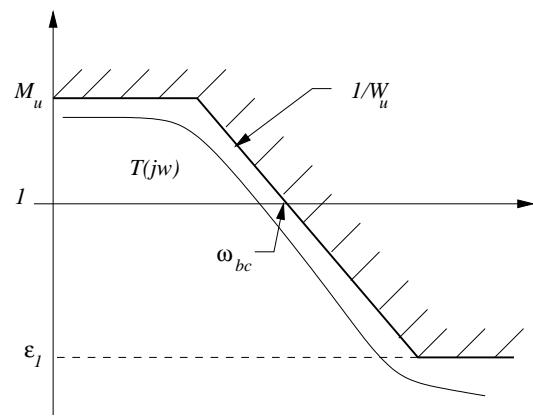
where $k \geq 1$ and controls the steepness of the transition between low-frequency and high-frequency.

Similarly for the complementary sensitivity T , an appropriate W_u is chosen (refer to Figure 3.7) as:

$$W_u = \left(\frac{s + \omega_{bc}/\sqrt[k]{M_u}}{\sqrt[k]{\varepsilon_1} s + \omega_{bc}} \right)^k$$

where $k \geq 1$.

For the multivariable case, the weights can of course be chosen as diagonal matrices with each diagonal term chosen as above.

Figure 3.6: Performance weight W_e and desired S Figure 3.7: Robustness weight W_u and desired T

3.5 \mathcal{H}_∞ Control

The linear quadratic regulator, Kalman filter, and linear quadratic gaussian problems can all be posed as 2-norm optimization problems. These optimization problems can be alternatively posed using the system \mathcal{H}_∞ -norm as a cost function. The \mathcal{H}_∞ -norm is the worst-case gain of the system and therefore provides a good match to engineering specifications, which are typically given in terms of bounds on errors and controls [51].

3.5.1 The term \mathcal{H}_∞

The terms \mathcal{H}_∞ norm and \mathcal{H}_∞ control are not terms which convey a lot of engineering significance. When we talk about \mathcal{H}_∞ , we are talking about a design method which aims to minimize the peak(s) of one or more selected transfer functions. The \mathcal{H}_∞ norm of a stable scalar transfer function $F(s)$ is the peak value of $|F(j\omega)|$ as a function of frequency, that is

$$\|F(s)\|_\infty \triangleq \max_{\omega} |F(j\omega)|$$

Strictly speaking, “max” (the maximum value) should be replaced by “sup” (supremum, the least upper bound) because the maximum may only be approached as $\omega \rightarrow \infty$ and may therefore not actually be achieved.

The symbol ∞ comes from the fact that the maximum magnitude over frequency may be written as

$$\max_{\omega} |F(j\omega)| = \lim_{p \rightarrow \infty} \left(\int_{-\infty}^{\infty} |F(j\omega)|^p d\omega \right)^{\frac{1}{p}}$$

Essentially, by raising $|F|$ to an infinite power, we pick out its peak value. \mathcal{H}_∞ is the set of transfer functions with bounded ∞ -norm, which is the set of stable and proper transfer functions [60].

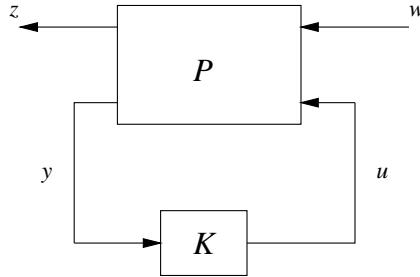


Figure 3.8: Block diagram

3.5.2 Problem Formulation

Given a proper continuous time linear time-invariant plant P mapping exogeneous inputs w and control inputs u to controlled outputs z and measured outputs y . That is,

$$\begin{pmatrix} z(s) \\ y(s) \end{pmatrix} = P(s) \begin{pmatrix} w(s) \\ u(s) \end{pmatrix}$$

and given some dynamic output feedback law $u = K(s)y$ and with the partitioning

$$P(s) = \begin{pmatrix} P_{11}(s) & P_{12}(s) \\ P_{21}(s) & P_{22}(s) \end{pmatrix}$$

the closed-loop transfer function from disturbance w to controlled output z is :

$$\mathcal{F}(P, K) = P_{11} + P_{12}K(I - P_{22}K)^{-1}P_{21}$$

The overall control objective is to minimize the H_∞ norm of the transfer function from w to z . This is done by finding a controller K which, based on the information in y , generates a control signal u which counteracts the influence of w on z , thereby minimizing the closed-loop norm from w to z .

In practice, we calculate the suboptimal rather than optimal solution. The suboptimal \mathcal{H}_∞ control problem of parameter γ consists of finding a controller $K(s)$ such that:

- the closed-loop system is internally stable
- the \mathcal{H}_∞ norm of $\mathcal{F}(P, K)$ (the maximum gain from w to z) is strictly less than γ , where γ is some prescribed performance level [21, 22, 27, 38].

It might be noticed here that the term “suboptimal” is used rather than “optimal”.

The reason for that is that it is often not necessary and sometimes even undesirable to design an optimal controller. A suboptimal controller may also have nice properties (e.g., lower bandwidth) over the optimal ones. However, knowing the achievable optimal (minimum) \mathcal{H}_∞ norm may be useful theoretically since it sets a limit on what can be achieved.

We shall now introduce some minimal realization of the plant P as is usual in state-space approaches to \mathcal{H}_∞ control:

$$P(s) = \begin{pmatrix} D_{11} & D_{12} \\ D_{21} & D_{22} \end{pmatrix} + \begin{pmatrix} C_1 \\ C_2 \end{pmatrix} (sI - A)^{-1} (B_1 \ B_2)$$

This realisation corresponds to the state-space equations :

$$\begin{aligned} \dot{x} &= Ax + B_1 w + B_2 u \\ z &= C_1 x + D_{11} w + D_{12} u \\ y &= C_2 x + D_{21} w + D_{22} u \end{aligned}$$

The problem dimensions are summarised by:

$$A \in \mathbb{R}^{n \times n}; \quad D_{11} \in \mathbb{R}^{p_1 \times m_1}; \quad D_{22} \in \mathbb{R}^{p_2 \times m_2}$$

The assumptions on the plant parameters are :

- (A, B_2, C_2) is stabilizable and detectable,
- $D_{22} = 0$

Note that the assumption that $D_{22} = 0$ is a temporary assumption leading to a simplified form of solution which can always be reversed. For more details on this, refer to [53].

3.5.3 Linear Matrix Inequality Approach to \mathcal{H}_∞ Control

An \mathcal{H}_∞ controller minimizes the worst case gain of the system. This problem can be thought of as a dynamic game with two participants: the designer, who is seeking a control that minimizes the gain; and nature, which is seeking a disturbance that maximizes the gain (worst-case input). Game theory is one of the approaches to solving \mathcal{H}_∞ problems [13]. Operator theory is another method [8].

In this section however, we will review the most recent approach of designing \mathcal{H}_∞ controllers using LMIs [22]. Given any proper real-rational controller $K(s)$ of realisation

$$K(s) = D_K + C_K(sI - A_K)^{-1}B_K \quad A_K \in \mathbb{R}^{k \times k}$$

a realisation of the closed-loop transfer function from w to z is obtained as:

$$\mathcal{F}(P, K)(s) = D_{cl} + C_{cl}(sI - A_{cl})^{-1}B_{cl}$$

where

$$\begin{aligned} A_{cl} &= \begin{pmatrix} A + B_2 D_K C_2 & B_2 C_K \\ B_K C_2 & A_K \end{pmatrix} & B_{cl} &= \begin{pmatrix} B_1 + B_2 D_K D_{21} \\ B_K D_{21} \end{pmatrix} \\ C_{cl} &= (C_1 + D_{12} D_K C_2, \quad D_{12} C_K) & D_{cl} &= D_{11} + D_{12} D_K D_{21} \end{aligned}$$

Then it can be stated that the continuous-time γ -suboptimal \mathcal{H}_∞ problem is solvable if and only if there exist symmetric matrices R, S satisfying the following

LMI system :

$$\left(\begin{array}{c|c} \mathcal{N}_R & 0 \\ \hline 0 & I \end{array} \right)^T \left(\begin{array}{cc|c} AR + RA^T & RC_1^T & B_1 \\ C_1 R & -\gamma I & D_{11} \\ \hline B_1^T & D_{11}^T & -\gamma I \end{array} \right) \left(\begin{array}{c|c} \mathcal{N}_R & 0 \\ \hline 0 & I \end{array} \right) < 0 \quad (3.5.1)$$

$$\left(\begin{array}{c|c} \mathcal{N}_S & 0 \\ \hline 0 & I \end{array} \right)^T \left(\begin{array}{cc|c} A^T S + SA & SB_1 & C_1^T \\ B_1^T S & -\gamma I & D_{11}^T \\ \hline C_1 & D_{11} & -\gamma I \end{array} \right) \left(\begin{array}{c|c} \mathcal{N}_S & 0 \\ \hline 0 & I \end{array} \right) < 0 \quad (3.5.2)$$

$$\begin{pmatrix} R & I \\ I & S \end{pmatrix} \geq 0 \quad (3.5.3)$$

where \mathcal{N}_R and \mathcal{N}_S denote bases of the null spaces of (B_2^T, D_{12}^T) and (C_2, D_{21}) , respectively. In addition, there exist γ -suboptimal controllers of order $k < n$ (reduced order) if and only if (3.5.1)-(3.5.3) hold for some R, S that further satisfy :

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

After computing (R, S) of the system of LMI (3.5.1)-(3.5.3), we then seek a positive definite matrix $X_{cl} \in \mathbb{R}^{(n+k) \times (n+k)}$. And to this end, it is necessary to compute two full-column-rank matrices $M, N \in \mathbb{R}^{n \times k}$ such that

$$MN^T = I - RS$$

An adequate X_{cl} is then obtained as the unique solution of the linear equation:

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

Then the Bounded Real Lemma Inequality for this X_{cl} is :

$$\begin{pmatrix} A_{cl}^T X_{cl} + X_{cl} A_{cl} & X_{cl} B_{cl} & C_{cl}^T \\ B_{cl}^T X_{cl} & -\gamma I & D_{cl}^T \\ C_{cl} & D_{cl} & -\gamma I \end{pmatrix} = \Psi_{X_{cl}} + \mathcal{Q}^T \Theta^T \mathcal{P}_{X_{cl}} + \mathcal{P}_{X_{cl}}^T \Theta \mathcal{Q} < 0 \quad (3.5.4)$$

and this inequality is solved for the controller parameters

$$\Theta = \begin{pmatrix} A_K & B_K \\ C_K & D_K \end{pmatrix} \quad (3.5.5)$$

where

$$\Psi_{X_{cl}} = \begin{pmatrix} A_0^T X_{cl} + X_{cl} A_0 & X_{cl} B_0 & C_0^T \\ B_0^T X_{cl} & -\gamma I & D_{11}^T \\ C_0 & D_{11} & -\gamma I \end{pmatrix} \quad (3.5.6)$$

$$\mathcal{P}_{X_{cl}} = (\beta^T X_{cl}, 0_{(k+m_2) \times m_1}, \mathcal{D}_{12}^T) \quad (3.5.7)$$

$$\mathcal{Q} = (\mathcal{C}, \mathcal{D}_{12}, 0_{(k+p_2) \times p_1}) \quad (3.5.8)$$

and

$$A_0 = \begin{pmatrix} A & 0 \\ 0 & 0_k \end{pmatrix}; \quad B_0 = \begin{pmatrix} B_1 \\ 0 \end{pmatrix}; \quad C_0 = (C_1, 0)$$

$$\beta = \begin{pmatrix} 0 & B_2 \\ I_k & 0 \end{pmatrix}; \quad \mathcal{C} = \begin{pmatrix} 0 & I_k \\ C_2 & 0 \end{pmatrix}; \quad \mathcal{D}_{12} = (0, D_{12}); \quad \mathcal{D}_{21} = \begin{pmatrix} 0 \\ D_{21} \end{pmatrix}$$

The computation of adequate (R, S) and the controller reconstruction Θ reduce to solving LMIs, and hence to convex optimization programs. Moreover, its solutions parameterize the set of \mathcal{H}_∞ controllers and bear important connections with the controller order. Also, this approach can be used with plants in which P_{12} and P_{21} have transmission zeros on the imaginary axis [22]. This routine is implemented and is used for future synthesis of controllers on the Case Studies.

3.5.4 Application of \mathcal{H}_∞ controllers

\mathcal{H}_∞ control can be used as an alternative to LQG optimal control. Both approaches are reasonable for a wide range of problems, and in many applications the choice of

a quadratic versus an \mathcal{H}_∞ -norm cost function is arbitrary. In these applications, the LQG controller is typically selected, since controller optimization is simpler and yields a unique solution. But the LQG control system may have undesirable properties; that is, it may not be robust, it may have an undesirable frequency response and so on [22, 51]. In these cases, it is preferable to try an \mathcal{H}_∞ controller.

\mathcal{H}_∞ control is a natural for applications where the specifications are given in terms of frequency dependent bounds on the outputs (both output errors and controls) [27]. Requiring the input to output gain to remain below prescribed levels is typical of engineering design specifications. The existence conditions for a suboptimal \mathcal{H}_∞ controller are very useful when performing trade-offs between competing control objectives. These existence conditions can be used to determine when a given set of specifications are consistent with a reasonable design and when the system is over-specified.

In this context, we introduce the Mixed sensitivity \mathcal{H}_∞ design approach which does not necessarily look for an optimum solution but rather looks for a solution which satisfies many requirements or specifications at once. Shaping the sensitivity function $S = (I + GK)^{-1}$ along with one or more other closed-loop transfer functions such as KS or the complementary sensitivity function $T = I - S$ provides a direct and effective way of achieving multi-variable loop shaping.

In the problem formulation, disturbance attenuation specifications, stability margin specifications as well as other specifications can be combined into a single infinity norm specification of the form :

$$\left\| \begin{bmatrix} W_1S \\ W_2KS \\ W_3T \end{bmatrix} \right\|_\infty$$

and this is usually called the mixed-sensitivity cost function which is to be minimized.

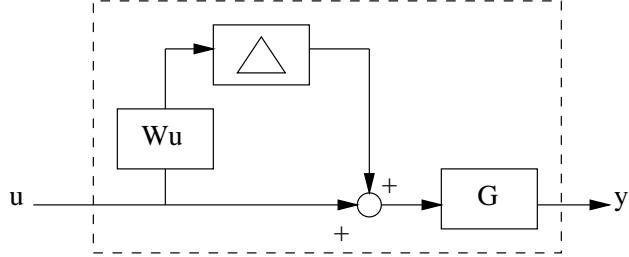


Figure 3.9: Multiplicative Uncertainty

3.6 μ -Synthesis

Most of the research on robust control has focused on \mathcal{H}_∞ like problems. However, it turns out that many practical problems do not readily fit the standard H_∞ problem setup since the involved model uncertainty tend to be a combination of both structured and unstructured uncertainty. We can only handle full complex perturbation structures non-conservatively in an \mathcal{H}_∞ robustness test. The conservatism introduced depends on the applied perturbation model and the condition number of the plant. In this section, it will be shown how these limitations are overcome using the structured singular value (SSV) μ [19, 28, 37, 51].

The structured singular value proves to be useful when the models of uncertainty are not limited to parametric uncertainty. Often, a low order, nominal model, which suitably describes the low-mid frequency range behaviour of the plant is available, but the high frequency plant behaviour is uncertain. In this situation, even the dynamic order of the actual plant is not known, and something broader than parametric uncertainty is needed to represent this uncertainty. One common approach for this type of uncertainty is to use a “multiplicative uncertainty model” as shown in Figure 3.9. This allows the specification of a frequency-dependent percentage uncertainty in the actual plant behaviour and to do that, we need to specify two things:

1. A nominal model, $G(s)$
2. A multiplicative uncertainty weighting function, $W_u(s)$

Given these, the precise definition of the multiplicative uncertainty set is

$$\mathcal{M}(G, W_u) := \left\{ \tilde{G} : \left| G(j\omega)^{-1}(\tilde{G}(j\omega) - G(j\omega)) \right| \leq |W_u(j\omega)| \right\}$$

where \tilde{G} represents the perturbed model.

At each frequency, $|W_u(j\omega)|$ represents the maximum potential percentage difference between all of the plants represented by $\mathcal{M}(G, W_u)$ and the nominal plant model G . In that sense, $\mathcal{M}(G, W_u)$ represents a ball of possible plants, centered at G . By defining $\Delta := (GW_u)^{-1}(\tilde{G} - G)$, each \tilde{G} can be represented by Figure 3.9.

In order to satisfy the constraint $\tilde{G} \in \mathcal{M}(G, W_u)$, Δ must be a transfer function that satisfies $\max_{\omega} |\Delta(j\omega)| \leq 1$.

3.6.1 D-K Iteration

The approach to controller synthesis using μ for complex perturbations [44] is frequently denoted *D-K* iteration and can be accomplished with the aid of μ -Toolbox [6]. We will here seek to design a controller that minimizes a given μ -condition. The *D-K* iteration method combines \mathcal{H}_{∞} -synthesis and μ -analysis. Given a system in standard form as shown in Figure 3.10, the starting point is the upper bound on μ in terms of the scaled singular value

$$\mu(N) \leq \min_{D \in \mathcal{D}} \bar{\sigma}(DND^{-1})$$

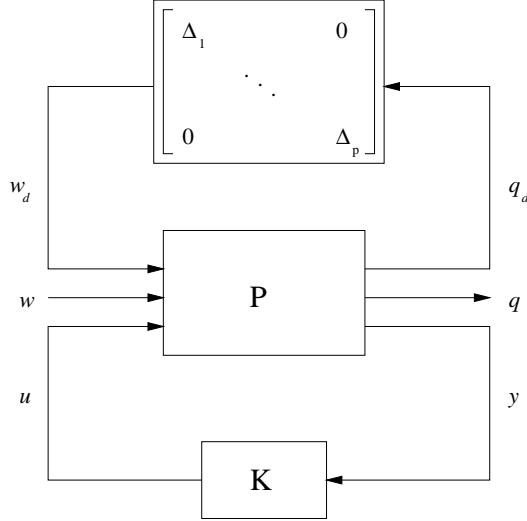


Figure 3.10: Standard form for robust analysis and synthesis

where N is the nominal closed-loop system formed by combining P and K . The matrix

$$D = \begin{bmatrix} d_1 I_1 & 0 & \dots & 0 \\ 0 & \ddots & & \\ & & d_r I_r & \vdots \\ \vdots & & D_1 & \vdots \\ \vdots & & \ddots & \\ 0 & \dots & & D_\Delta \end{bmatrix}$$

is referred to as D -scaling matrix. \mathcal{D} is the set of scalar terms d_i which are referred to as D -scales but it can also consist of full-blocks D_i which arise when repeated scalars are present in the uncertainty block.

The idea is to find the controller that minimizes the peak value over frequency of this upper bound, namely

$$\min_K (\min_{D \in \mathcal{D}} \|DND^{-1}\|_\infty)$$

by alternating between minimizing $\|DND^{-1}\|_\infty$ with respect to either K or D (while holding the other fixed).

The iteration may continue until satisfactory performance is achieved, $\|DND^{-1}\|_\infty < 1$, or until the H_∞ norm no longer decreases.

The μ -synthesis design method can be summarized as follows [13]:

1. Model the plant. The plant model should include disturbance inputs, control inputs, reference outputs, measured outputs, and perturbations.
2. Generate a control system to minimize the ∞ -norm of the transfer function from the augmented perturbation input to the augmented perturbation output.
3. Compute the structured singular values for the closed-loop system. Save the D -scales in computing the structured singular value.
4. Fit a low-order transfer function to each frequency dependent D -scale.
5. Append these transfer functions to the plant. The rational transfer function approximations for the D -scales and the inverse D -scales are appended to the nominal closed-loop system.
6. For this augmented plant, generate a controller to minimize the ∞ -norm of the transfer function from the augmented perturbation input to the augmented perturbation output.
7. Return to step 4, until the structured singular value of the closed-loop system fails to improve.

The order of the controller resulting from each iteration is equal to the number of states in the plant $G(s)$ plus the number of states in the weights plus twice the number of states in $D(s)$. One fundamental problem with this approach is that although

each of the minimization steps(K -step and D -step) are convex, joint convexity is not guaranteed [14, 18, 42]. Therefore, the iterations may converge to a local optimum rather than to a global optimum. Also for most cases, the true μ -optimal controller is not rational, and will thus be of infinite order, but because we use a finite order $D(s)$ to approximate the D -scales, we get a controller of finite (but often high) order.

With the help of the μ -Toolbox [6], this process is automated with a graphical user interface using the ‘`dkitgui`’ command. However in the current release of the μ -Toolbox (Version 3.0.3), repeated scalar complex blocks are not supported. For repeated scalar blocks the corresponding D scaling is a full matrix. Thus the number of SISO transfer function approximations needed in connection with D - K iteration grows quadratically with the number of repeated scalar perturbations. An approximation to D - K iteration with repeated scalar blocks can be made by considering them as uncorrelated. However, the design will then not be optimal. By analyzing the final design with the true perturbation structure, the amount of conservatism introduced can be assessed. Furthermore, an approximation to mixed real and complex μ synthesis can be achieved with D - K iteration by approximating all real perturbations with complex ones. Again the design will not be optimal but through μ analysis the design may be tested against the true perturbation structure.

3.6.2 Advantages and Drawbacks of μ -Synthesis

One of the powerful features of μ is that it can handle real perturbations (parametric uncertainty) as well. This uncertainty description is superior to the assumption in connection with \mathcal{H}_∞ control where only full complex blocks or parametric uncertainty can be considered at one time. Using μ , we can consider robust performance non-conservatively for a very large class of perturbation structures including parametric

uncertainty and mixed dynamic and parametric uncertainty.

Unfortunately, μ cannot in general be precisely computed since the implied optimization problem is non-convex. This fact has naturally hampered the application of μ theory to real control systems design [40, 54, 63]. Rather than trying to compute μ itself, it is then customary to look for upper and lower bounds on μ . For purely complex perturbation sets, the complex μ upper bound is generally very close to the true μ value [19]. However, in general the bounds on mixed μ seem to be much more conservative than the purely complex μ bounds. In particular, for a large number of real perturbations, the gap between the upper and lower bound may be quite large.

Also, usually the approach tends to lead to the synthesis of very high order controllers. It is often worthwhile to consider applying Model Reduction typically to the controller synthesized after the synthesis.

Chapter 4

Gain-Scheduled Control Theory

4.1 Introduction

After decades of silence, the academic research community in the 1990s addressed the topic of gain-scheduling. There is a long history of gain-scheduling in applications, but there are few citations to the control literature before 1990. The theoretical treatment of gain-scheduling as a worthy design methodology is rare until the 1990s [48]. Gain-scheduling was considered an “applied” topic, but because of its superior performance capability, it was quickly adopted in military applications even though it came at a higher cost. It is very much an old idea, but before digital implementation of controllers, it was expensive and difficult to realize in hardware.

Gain-Scheduling is now a widely used technique for controlling certain classes of nonlinear or linear time-varying systems [3, 7, 20, 29, 56]. Rather than seeking a single robust LTI controller for the entire operating range (which tends to be conservative if at all feasible), gain scheduling generally consists in designing an LTI controller for each operating point and in switching controllers when the operating conditions change. The controller coefficients are continuously varied according to the current value of the “scheduling variables”, that may be either exogenous signals

or endogenous signals with respect to the plant [1, 2, 43].

According to [48], the design of a gain-scheduled controller for a nonlinear plant can be described as a four-step procedure, which we shall re-iterate :

1. Gain scheduling is based on linear parameter-varying plant models. The first step is then to compute a linear parameter-varying model for the plant. There are two approaches to this:
 - (a) The most common one is based on Jacobian linearization of the nonlinear plant about a family of equilibrium points, yielding a parametrized family of linearized plants. With this approach, stability can be assured only locally and in a “slow-variation” setting, and there are no performance guarantees. Extensive simulation is needed for evaluation of stability and performance.
 - (b) The other approach is what is known as “quasi-LPV” scheduling, in which the plant dynamics are rewritten to hide nonlinearities as time-varying parameters that are then used as scheduling variables.
2. The second step is to use linear design controller techniques for the LPV plant model that arises from either the linearization or quasi-LPV approach. This may result directly in a family of linear controllers corresponding to the linear parameter-dependent plant, or there may be an interpolation process to arrive at a family of linear controllers from a set of controller designs at isolated values of the scheduling variables.
3. The third step involves implementing the family of linear controllers such that the controller coefficients(gains) are varied(scheduled) according to the current value of the scheduling variables.

4. The final step is performance assessment. This may be relatively simple where analytical performance guarantees are part of the design process. More typically, the local stability and performance properties of the gain-scheduled controller might be subject to analytical investigation, while the nonlocal performance evaluation is based on simulation studies.

Let us now look at how gain-scheduled controllers are represented. The traditional exogenously gain-scheduled controller, which is adjusted with reference to an externally measured variable, $\rho(t)$, has the form

$$\begin{aligned}\dot{x} &= A(\rho(t))x + B(\rho(t))y \\ u &= C(\rho(t))x + D(\rho(t))y\end{aligned}\tag{4.1.1}$$

The dynamic properties change with $\rho(t)$ and provided that the rate of change is not too rapid, then the dynamic properties of the time-varying controller, (4.1.1), are similar to those of the linear controllers obtained by “freezing” the value of ρ ; that is, the nonlinear controller inherits the dynamic properties of the family of linear controllers. It should be noted that there are no direct restrictions on the state, x , or the input, u . The only restriction is on the rate of change of the scheduling variable; indeed, when the scheduling variable is constant, the controller is linear [34, 36].

4.2 Polytopic Approach to Gain-Scheduling

The synthesis technique discussed below follows from [3] and is applicable to affine parameter-dependent plants with equations

$$\begin{aligned}\dot{x} &= A(p)x + B_1(p)w + B_2u \\ z &= C_1(p)x + D_{11}(p)w + D_{12}u \\ y &= C_2x + D_{21}w + D_{22}u\end{aligned}\tag{4.2.1}$$

where

$$p(t) = (p_1(t), \dots, p_n(t)), \quad \underline{p}_i \leq p_i(t) \leq \bar{p}_i$$

is a time-varying vector of physical parameters and $A(\cdot), B_1(\cdot), C_1(\cdot), D_{11}(\cdot)$ are affine functions of $p(t)$ [43]. The parameter vector, $p(t)$ may include part of the state vector x itself provided that the corresponding states are available for measurement.

The plant system matrix for (4.2.1) is:

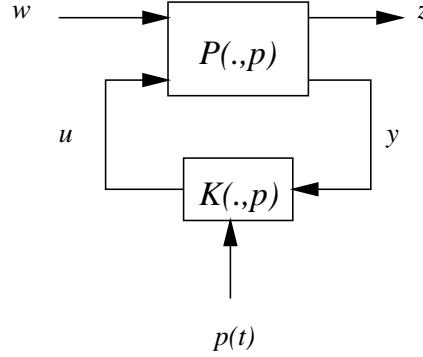
$$S(p) := \left(\begin{array}{c|cc} A(p) & B_1(p) & B_2 \\ \hline C_1(p) & D_{11}(p) & D_{12} \\ \hline C_2 & D_{21} & D_{22} \end{array} \right)$$

If the parameter vector $p(t)$ takes values in a box of \mathbb{R}^n with corners $\{\Pi_i\}_{i=1}^N (N = 2^n)$, then $S(p)$ ranges in a matrix polytope with vertices $S(\Pi_i)$. Specifically, given any convex decomposition

$$p(t) = \alpha_1\Pi_1 + \dots + \alpha_N\Pi_N, \quad \alpha_i \geq 0, \quad \sum_{i=1}^N \alpha_i = 1$$

of p over the corners of the parameter box, the system matrix $S(p)$ is given by

$$S(p) = \alpha_1S(\Pi_1) + \dots + \alpha_NS(\Pi_N).$$

Figure 4.1: Gain-Scheduled \mathcal{H}_∞ problem

With reference to the feedback system arrangement shown in Figure 4.1, we seek a parameter-dependent controller with equation

$$\begin{aligned}\dot{x}_K &= A_K(p)x_K + B_K(p)y \\ u &= C_K(p)x_K + D_K(p)y\end{aligned}\tag{4.2.2}$$

and having the following vertex property:

$$\begin{pmatrix} A_K(p) & B_K(p) \\ C_K(p) & D_K(p) \end{pmatrix} = \sum_{i=1}^N \alpha_i \begin{pmatrix} A_K(\Pi_i) & B_K(\Pi_i) \\ C_K(\Pi_i) & D_K(\Pi_i) \end{pmatrix}$$

That is, the controller state-space matrices at the operating point $p(t)$ are obtained by convex interpolation of the LTI vertex controllers

$$K_i = \begin{pmatrix} A_K(\Pi_i) & B_K(\Pi_i) \\ C_K(\Pi_i) & D_K(\Pi_i) \end{pmatrix}$$

yielding a smooth scheduling of the controller matrices by the parameter measurements $p(t)$.

For this class of controllers, consider the following \mathcal{H}_∞ -like synthesis problem relative to the interconnection of Figure 4.1

The aim is to design a gain-scheduled controller $K(., p)$ satisfying the vertex property and such that

- the closed-loop system is stable for all admissible parameter trajectories $p(t)$
- the worst-case closed-loop \mathcal{H}_∞ gain from w to z does not exceed some level $\gamma > 0$.

Using the notion of quadratic \mathcal{H}_∞ performance to enforce the root mean square (RMS) gain constraint, this synthesis problem can be reduced to the following LMI problem [9]:

Find two symmetric matrices (R, S) such that

$$\begin{aligned} \left(\begin{array}{c|c} \mathcal{N}_{12} & 0 \\ \hline 0 & I \end{array} \right)^T \left(\begin{array}{cc|c} A_i R + R A_i^T & R C_{1i}^T & B_{1i} \\ C_{1i} R & -\gamma I & D_{11} \\ \hline B_1^T & D_{11i}^T & -\gamma I \end{array} \right) \left(\begin{array}{c|c} \mathcal{N}_{12} & 0 \\ \hline 0 & I \end{array} \right) &< 0, \quad i = 1, \dots, N \\ \left(\begin{array}{c|c} \mathcal{N}_{21} & 0 \\ \hline 0 & I \end{array} \right)^T \left(\begin{array}{cc|c} A_i^T S + S A_i & S B_{1i} & C_{1i}^T \\ B_{1i}^T S & -\gamma I & D_{11i}^T \\ \hline C_{1i} & D_{11i} & -\gamma I \end{array} \right) \left(\begin{array}{c|c} \mathcal{N}_{21} & 0 \\ \hline 0 & I \end{array} \right) &< 0, \quad i = 1, \dots, N \\ \begin{pmatrix} R & I \\ I & S \end{pmatrix} &\geq 0 \end{aligned}$$

where

$$\begin{pmatrix} A_i & B_{1i} \\ C_{1i} & D_{11i} \end{pmatrix} := \begin{pmatrix} A(\Pi_i) & B_1(\Pi_i) \\ C_1(\Pi_i) & D_{11}(\Pi_i) \end{pmatrix} \quad (4.2.3)$$

and \mathcal{N}_{12} and \mathcal{N}_{21} are bases of the null spaces of (B_2^T, D_{12}^T) and (C_2, D_{21}) , respectively.

From there on, the same approach as in Section 3.5.3 can be followed to solve for the controllers at each vertex.

4.3 LFT Approach to Gain Scheduling using LMI

Usually, we can model or approximate the parameter dependence in an LPV system as an LFT. In general, an LFT description is capable of representing any polynomial

or rational matrix function of a scalar variable. LMIs can be used to fully characterize the existence of gain-scheduled controllers when the dependence on time-varying but measured parameters is linear fractional. The problem is then less complex than for the general LPV case and the computational burden required is very similar to that for the standard linear time-invariant case. This is what we are going to look at in more details.

LPV plants with a linear fractional dependence on θ can be represented by the upper LFT interconnection

$$\begin{pmatrix} q \\ y \end{pmatrix} = F_u(P(s), \Theta) \begin{pmatrix} w \\ u \end{pmatrix} \quad (4.3.1)$$

where $P(s)$ is a known LTI plant and Θ is some block diagonal time-varying operator specifying how θ enters the plant dynamics as depicted in Figure 4.2. Specifically

$$\Theta = \text{blockdiag}(\theta_1 I_{r_1}, \dots, \theta_K I_{r_K}) \quad (4.3.2)$$

where $r_i > 1$ whenever the parameter θ_i is repeated. The set of operators with structure (4.3.2) is denoted by

$$\Delta := \{\text{blockdiag}(\theta_1 I_{r_1}, \dots, \theta_K I_{r_K}) : \theta_i(\tau) \in \mathbb{R}\} \quad (4.3.3)$$

The approach is based on the concept of parameter-dependent \mathcal{H}_∞ controllers developed in [1, 2]. The controllers depend on the varying parameters $\theta(t)$ through

$$\dot{x}_K(t) = A_K(\theta(t)) x_K(t) + B_K(\theta(t)) y(t) \quad (4.3.4)$$

$$u(t) = C_K(\theta(t)) x_K(t) + D_K(\theta(t)) y(t) \quad (4.3.5)$$

where A_K, B_K, C_K, D_K are linear fractional functions of θ . This control structure is applicable whenever the value of $\theta(t)$ is measured at each time t . The resulting controller is time-varying and smoothly ‘‘scheduled’’ by the measurements of $\theta(t)$.

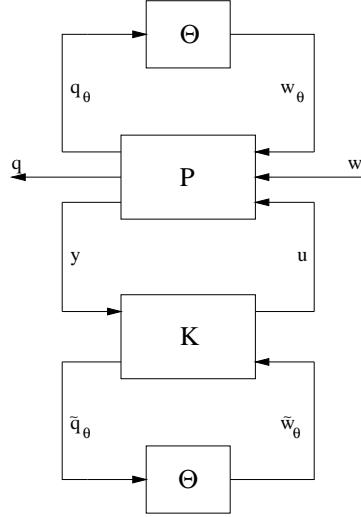


Figure 4.2: LPV control structure

Consistently with (4.3.1), we seek LPV controllers of the form

$$u = F_l(K(s), \Theta)y \quad (4.3.6)$$

Recalling that θ plays the role of scheduling variable, (4.3.6) gives the rule for updating the controller state-space matrices based on the measurements of θ .

The overall LFT interconnection is depicted in Figure 4.2. The closed-loop operator from disturbance w to controlled output q is given by

$$T(P, K, \Theta) = F_l(F_u(P, \Theta), F_l(K, \Theta)) \quad (4.3.7)$$

As previously stated in Section 3.5, given some LTI plant $P(s)$ mapping exogenous inputs w and control inputs u to controlled outputs q and measured outputs y , the usual \mathcal{H}_∞ control problem is concerned with finding an internally stabilizing LTI controller $K(s)$ such that

$$\|F_l(P, K)\|_\infty < \gamma$$

where γ is some prescribed performance level.

The gain-scheduled version of this problem has a similar statement, except that both the plant and the controller are now LPV instead of LTI. Here the objective is to guarantee some closed loop performance $\gamma > 0$ from w to q for all admissible parameter trajectories $\theta_{(\tau)}$. The gain-scheduled problem is now stated :

Consider some minimal realization of an LTI plant $P(s)$

$$P(s) = \begin{pmatrix} D_{\theta\theta} & D_{\theta 1} & D_{\theta 2} \\ D_{1\theta} & D_{11} & D_{12} \\ D_{2\theta} & D_{21} & D_{22} \end{pmatrix} + \begin{pmatrix} C_\theta \\ C_1 \\ C_2 \end{pmatrix} (sI - A)^{-1} (B_\theta \quad B_1 \quad B_2) \quad (4.3.8)$$

The problem dimensions are given by

$$A \in \mathbb{R}^{n \times n}, D_{\theta\theta} \in \mathbb{R}^{r \times r}, D_{11} \in \mathbb{R}^{p_1 \times p_1}, D_{22} \in \mathbb{R}^{p_2 \times m_2}$$

and letting

$$\hat{B}_1 = (B_\theta, B_1), \quad \hat{C}_1 = \begin{pmatrix} C_\theta \\ C_1 \end{pmatrix}, \quad \hat{D}_{11} = \begin{pmatrix} D_{\theta\theta} & D_{\theta 1} \\ D_{1\theta} & D_{11} \end{pmatrix}$$

A gain-scheduled controller can be designed if there exist pairs of symmetric matrices (R, S) in $\mathbb{R}^{n \times n}$ and (L_3, J_3) in $\mathbb{R}^{r \times r}$ such that

$$\mathcal{N}_R^T \left(\begin{array}{ccc} AR + RA^T & R\hat{C}_1^T & \hat{B}_1 \begin{pmatrix} J_3 & 0 \\ 0 & I \end{pmatrix} \\ \hat{C}_1 R & -\gamma \begin{pmatrix} J_3 & 0 \\ 0 & I \end{pmatrix} & \hat{D}_{11} \begin{pmatrix} J_3 & 0 \\ 0 & I \end{pmatrix} \\ \begin{pmatrix} J_3 & 0 \\ 0 & I \end{pmatrix} \hat{B}_1^T & \begin{pmatrix} J_3 & 0 \\ 0 & I \end{pmatrix} \hat{D}_{11}^T & -\gamma \begin{pmatrix} J_3 & 0 \\ 0 & I \end{pmatrix} \end{array} \right) \mathcal{N}_R < 0 \quad (4.3.9)$$

$$\mathcal{N}_S^T \left(\begin{array}{ccc} A^T S + SA & S\hat{B}_1 & \hat{C}_1^T \begin{pmatrix} L_3 & 0 \\ 0 & I \end{pmatrix} \\ \hat{B}_1^T S & -\gamma \begin{pmatrix} L_3 & 0 \\ 0 & I \end{pmatrix} & \hat{D}_{11}^T \begin{pmatrix} L_3 & 0 \\ 0 & I \end{pmatrix} \\ \begin{pmatrix} L_3 & 0 \\ 0 & I \end{pmatrix} \hat{C}_1 & \begin{pmatrix} L_3 & 0 \\ 0 & I \end{pmatrix} \hat{D}_{11} & -\gamma \begin{pmatrix} L_3 & 0 \\ 0 & I \end{pmatrix} \end{array} \right) \mathcal{N}_S < 0 \quad (4.3.10)$$

$$\begin{pmatrix} R & I \\ I & S \end{pmatrix} \geq 0 \quad (4.3.11)$$

$$L_3 \in L_\Delta, J_3 \in L_\Delta, \begin{pmatrix} L_3 & I \\ I & J_3 \end{pmatrix} \geq 0. \quad (4.3.12)$$

where $L_\Delta = \{L > 0 : L\Theta = \Theta L, \forall \Theta \in \Delta\} \subset \mathbb{R}^{r \times r}$

Moreover, there exist γ -suboptimal controllers of order k if and only if (4.3.9)-(4.3.12) hold for some quadruple (R, S, L_3, J_3) where (R, S) further satisfy the rank constraint

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

4.3.1 Computation of the Gain-Scheduled Controller

Provided the LMIs specified in the previous section are satisfied, we can now address the actual computation of the gain-scheduled controller.

Allow the gain-scheduled controller $K(s)$ to be

$$K(s) = \begin{pmatrix} D_{K11} & D_{K1\theta} \\ D_{K\theta 1} & D_{K\theta\theta} \end{pmatrix} + \begin{pmatrix} C_{K1} \\ C_{K\theta} \end{pmatrix} (sI - A_K)^{-1} (B_{K1} \ B_{K\theta}) \quad (4.3.13)$$

$$A_K \in \mathbb{R}^{k \times k}, D_{K11} \in \mathbb{R}^{m_2 \times p_2}, D_{K\theta\theta} \in \mathbb{R}^{r \times r} \quad (4.3.14)$$

and let

$$\Omega := \begin{pmatrix} A_K & B_{K1} & B_{K\theta} \\ C_{K1} & D_{K11} & D_{K1\theta} \\ C_{K\theta} & D_{K\theta 1} & D_{K\theta\theta} \end{pmatrix} \in \mathbb{R}^{(k+m_2+r) \times (k+p_2+r)} \quad (4.3.15)$$

To simplify notation, the following shorthands are used

$$\begin{aligned}
 A_0 &= \begin{pmatrix} A & 0 \\ 0 & 0_{k \times k} \end{pmatrix}, & B_0 &= \begin{pmatrix} 0 & B_\theta & B_1 \\ 0_{k \times r} & 0 & 0 \end{pmatrix}, \\
 \mathcal{B} &= \begin{pmatrix} 0 & B_2 & 0 \\ I_k & 0 & 0_{k \times r} \end{pmatrix}, & C_0 &= \begin{pmatrix} 0 & 0_{r \times k} \\ C_\theta & 0 \\ C_1 & 0 \end{pmatrix}, \\
 \mathcal{D}_{11} &= \begin{pmatrix} 0_{r \times r} & 0 & 0 \\ 0 & D_{\theta\theta} & D_{\theta 1} \\ 0 & D_{1\theta} & D_{11} \end{pmatrix}, & \mathcal{D}_{12} &= \begin{pmatrix} 0_{r \times k} & 0 & I_r \\ 0 & D_{\theta 2} & 0 \\ 0 & D_{12} & 0 \end{pmatrix}, \\
 \mathcal{C} &= \begin{pmatrix} 0 & I_k \\ C_2 & 0 \\ 0 & 0_{r \times k} \end{pmatrix}, & \mathcal{D}_{21} &= \begin{pmatrix} 0_{k \times r} & 0 & 0 \\ 0 & D_{2\theta} & D_{21} \\ I_r & 0 & 0 \end{pmatrix} \tag{4.3.16}
 \end{aligned}$$

Given any solution (R, S, L_3, J_3) of LMI system (4.3.9)-(4.3.12), the state-space data of Ω of some γ -suboptimal $K(s)$ is computed as follows :

- From R, S the bounded real lemma matrix X_{cl} is derived by

1. computing via SVD two full-column rank matrices $M, N \in \mathbb{R}^{n \times k}$ such that

$$MN^T = I - RS$$

This is done by letting $U\Sigma V^T = I - RS$ and then letting $M = U\Sigma$ and $N = V$

2. computing X_{cl} as the unique solution of the linear matrix equation

$$X_{cl} \begin{pmatrix} I & R \\ 0 & M^T \end{pmatrix} = \begin{pmatrix} S & I \\ N^T & 0 \end{pmatrix} \tag{4.3.17}$$

- Two matrices $L_1 \in L_\Delta$ and L_2 commuting with the structure Δ are computed such that

$$L := \begin{pmatrix} L_1 & L_2 \\ L_2^T & L_3 \end{pmatrix} > 0, \quad L^{-1} = \begin{pmatrix} * & * \\ * & J_3 \end{pmatrix} \quad (4.3.18)$$

where $*$ represents “don’t care”.

From Lemma 2.1.2, it can be stated from (4.3.18) that

$$\begin{aligned} J_3 &= (L_3 - L_2^T L_1^{-1} L_2)^{-1} \\ L_3 - J_3^{-1} &= L_2^T L_1^{-1} L_2 \\ \text{Let } L_3 - J_3^{-1} &= U \Sigma U^T \\ \text{and } L_2^T L_1^{-1} L_2 &= U \Sigma U^T \\ \therefore L_1 &= \Sigma^{-1} \quad \text{and} \quad L_2 = U^T \end{aligned} \quad (4.3.19)$$

- Allow

$$\mathcal{L} := \begin{pmatrix} L & 0 \\ 0 & I_{p_1} \end{pmatrix}, \quad \mathcal{J} := \mathcal{L}^{-1} \quad (4.3.20)$$

- The following LMI is then solved for Ω

$$\Psi + \begin{pmatrix} X_{cl} & 0 \\ 0 & I \end{pmatrix} P^T \Omega Q + Q^T \Omega^T P \begin{pmatrix} X_{cl} & 0 \\ 0 & I \end{pmatrix} < 0 \quad (4.3.21)$$

where

$$\Psi = \begin{pmatrix} A_0^T X_{cl} + X_{cl} A_0 & X_{cl} B_0 & C_0^T \\ B_0^T X_{cl} & -\gamma \mathcal{L} & \mathcal{D}_{11}^T \\ C_0 & \mathcal{D}_{11} & -\gamma \mathcal{J} \end{pmatrix} \quad (4.3.22)$$

$$P := (\mathcal{B}^T, 0, \mathcal{D}_{12}^T), \quad Q := (\mathcal{C}, \mathcal{D}_{21}, 0) \quad (4.3.23)$$

A gain-scheduled controller can thus be designed. The controller parameters $(A_K, B_{K1}, \dots, D_{K\theta\theta})$ can be obtained and can be linear fractionally transformed with the scheduling variables giving the gain-scheduled controller.

For the case studies, we look at two main approaches to gain scheduled controller synthesis. The first approach which we shall call the polytopic approach is from [3, 23]. The other approach is the LFT approach which we just looked at. This latter approach [1] is programmed in MATLAB with errors removed as per [2] and is used for the Case Study examples.¹

It is important to remember that this linear matrix inequality based constructions of \mathcal{H}_∞ optimal controllers adapted to compute gain-scheduled controllers for LPV systems can be conservative. The reason for that being that the controller seeks to achieve a desired performance not only for system dynamics with admissible parameter trajectories, but also for system dynamics that can exhibit a much broader range of behaviours, i.e., the parameters are assumed to be complex and to vary arbitrarily quickly [1].

4.4 Stability and Performance of Gain-Scheduled Controllers

Even though gain scheduled designs are based on linearizations, the overall system is still nonlinear [32]. This essentially renders difficult the possibility of nonconservative stability and performance analysis results. Stability and performance properties of a gain scheduled design were not easily guaranteed in the early approaches such

¹For the source code of the program, refer to Appendix A

as multi-model linearization [10, 30, 31]. They were typically inferred from extensive simulations. Such properties are not necessarily assured by the stability and performance properties of the fixed operating condition designs.

The polytopic and LFT approach which we looked at before however ensures stability as the synthesis method takes into account the parameter time variations during the design itself. The newer approaches with rate limitations [9, 35, 39] are beyond the scope of this thesis as they suffer from the curse of dimensionality.

Part II

Case Studies

Chapter 5

Two-Link Robot Manipulator

In this chapter, we will use a sample robot manipulator model in order to illustrate various modelling issues and compare robust controller designs. Specifically, we will look into Mixed-Sensitivity \mathcal{H}_∞ design, μ -synthesis and LPV design using the “polytopic” approach.

5.1 Introduction

Robotics is concerned with the principle, design, manufacture, and application of robots, and is a broad field that involves many areas such as physics, mechanical design, motion analysis and planning, actuators and drivers, control design, sensors, signal and image processing, computer algorithms, and study of behaviour of machines, animals, and even human beings. The focus here is on the area of robot control.

Robot manipulators are familiar examples of position-controllable mechanical systems. However, their nonlinear dynamics present a challenging control problem, since traditional linear control approaches do not easily apply. Our objective is to model the complete nonlinear dynamics of a sample 2-joint manipulator, so that we can

simply control it to move from one point to another in a 2-dimensional space.

Three types of dynamics torques arise from the motion of the manipulator : Inertial, Centripetal, and Coriolis torques. **Inertial torques** are proportional to joint acceleration in accordance with Newton's second law. **Centripetal torques** arise from the centripetal forces which constrain a body to rotate about a point. They are directed towards the center of the uniform circular motion, and are proportional to the square of the velocity. **Coriolis torques** result from vertical forces derived from the interaction of two rotating links. They are proportional to the product of the joint velocities of those links.

For simplicity, we shall consider a planar, two-link, articulated manipulator whose position can be described by a 2-vector $q = (q_1 \ q_2)^T$ of joint angles, and whose actuator inputs consist of a 2-vector $u = (u_1 \ u_2)^T$ of torques applied at the manipulator joints as shown in Figure 5.1. Allowing \dot{q} to denote the joint velocities and \ddot{q} the joint accelerations, the dynamics of this simple manipulator can be written in the general form [25, 41, 55]

$$H(q)\ddot{q} + C(q, \dot{q})\dot{q} + g(q) = u \quad (5.1.1)$$

where $H(q)$ is a 2×2 manipulator inertia tensor matrix (which is symmetric positive definite), $C(q, \dot{q})\dot{q}$ is a 2-vector centripetal and Coriolis torques (with $C(q, \dot{q})$ a 2×2 matrix), and $g(q)$ is the 2-vector of gravitational torques. The feedback control we will be concerned with for such a system is to compute the required actuator inputs to perform desired tasks (*e.g.*, move to a final desired position or follow a desired trajectory), given the measured systems state, namely the vector q of joint angles, and the vector \dot{q} of joint velocities.

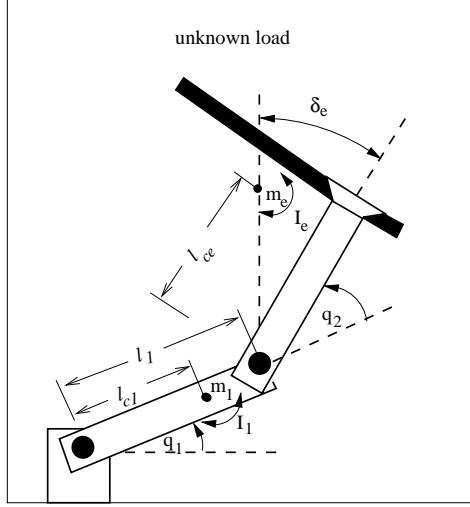


Figure 5.1: Two-link manipulator

5.2 Dynamics of the Model

Let us assume that the manipulator is in the horizontal plane ($g(q) \equiv 0$), then from (5.1.1) the dynamics of the two-link manipulator can be explicitly written as

$$\begin{pmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{pmatrix} \begin{pmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{pmatrix} + \begin{pmatrix} -h\dot{q}_2 & -h(\dot{q}_1 + \dot{q}_2) \\ h\dot{q}_1 & 0 \end{pmatrix} \begin{pmatrix} \dot{q}_1 \\ \dot{q}_2 \end{pmatrix} = \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \quad (5.2.1)$$

where

$$h_{11} = a_1 + 2a_3 \cos q_2 + 2a_4 \sin q_2 \quad (5.2.2)$$

$$h_{12} = h_{21} = a_2 + a_3 \cos q_2 + a_4 \sin q_2 \quad (5.2.3)$$

$$h_{22} = a_2 \quad (5.2.4)$$

$$h = a_3 \sin q_2 - a_4 \cos q_2 \quad (5.2.5)$$

with

$$a_1 = I_1 + m_1 l_{c1}^2 + I_e + m_e l_{ce}^2 + m_e l_1^2 \quad (5.2.6)$$

$$a_2 = I_e + m_e l_{ce}^2 \quad (5.2.7)$$

$$a_3 = m_e l_1 l_{ce} \cos \delta_e \quad (5.2.8)$$

$$a_4 = m_e l_1 l_{ce} \sin \delta_e \quad (5.2.9)$$

and

I_1 : inertia of arm 1

I_e : inertia due to arm 2 and unknown load

m_1 : mass of arm 1

m_e : mass of arm 2 and unknown load

l_1 : distance between joint 1 and joint 2

l_{c1} : distance of joint 1 from centre of mass of arm 1

l_{ce} : distance of joint 2 from centre of mass of arm 2

and unknown load

δ_e : angle between arm 2 and centre of mass of unknown
load

Let

$$x = [x_1 \ x_2 \ x_3 \ x_4]^T = [q_1 \ q_2 \ \dot{q}_1 \ \dot{q}_2]^T \quad (5.2.10)$$

then the dynamics of the two-link manipulator can be expressed as

$$h_{11}(x)\dot{x}_3 + h_{12}(x)\dot{x}_4 - h(x)x_4x_3 - h(x)(x_3 + x_4)x_4 = u_1 \quad (5.2.11)$$

$$h_{21}(x)\dot{x}_3 + h_{22}(x)\dot{x}_4 + h(x)x_3^2 = u_2 \quad (5.2.12)$$

$$\dot{x}_1 = x_3 \quad (5.2.13)$$

$$\dot{x}_2 = x_4 \quad (5.2.14)$$

These equations (5.2.11)–(5.2.14) can be rewritten in matrix form as follows:

$$\begin{array}{c|cc} 1 & 0 & 0 \\ 0 & 1 & 0 \\ \hline 0 & 0 & h_{11}(x) & h_{12}(x) \\ 0 & 0 & h_{21}(x) & h_{22}(x) \end{array} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{array}{c|cc} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \hline 0 & 0 & 2h(x)x_4 & h(x)x_4 \\ 0 & 0 & -h(x)x_3 & 0 \end{array} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 1 & 0 \\ 0 & 1 \end{bmatrix} \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad (5.2.15) \end{array}$$

or

$$\begin{bmatrix} I & O \\ O & H \end{bmatrix} \dot{x} = \begin{bmatrix} O & I \\ O & -J \end{bmatrix} x + \begin{bmatrix} O \\ I \end{bmatrix} u \quad (5.2.16)$$

where

$$H = \begin{bmatrix} h_{11}(x) & h_{12}(x) \\ h_{21}(x) & h_{22}(x) \end{bmatrix}; J = \begin{bmatrix} -2h(x)x_4 & -h(x)x_4 \\ h(x)x_3 & 0 \end{bmatrix}; I = I_{2 \times 2}; O = 0_{2 \times 2}$$

Therefore,

$$\dot{x} = \begin{bmatrix} I & O \\ O & M \end{bmatrix} \begin{bmatrix} O & I \\ O & -J \end{bmatrix} x + \begin{bmatrix} I & O \\ O & M \end{bmatrix} \begin{bmatrix} O \\ I \end{bmatrix} u \quad (5.2.17)$$

and

$$M = \begin{bmatrix} h_{11}(x) & h_{12}(x) \\ h_{21}(x) & h_{22}(x) \end{bmatrix}^{-1}$$

Parameter	Value	Units
m_1	1	kg
m_e	2	kg
I_1	0.12	kgm^2
I_e	0.25	kgm^2
l_1	1	m
l_{c1}	0.5	m
l_{ce}	0.6	m
δ_e	30	deg

Table 5.1: Parametric values used for simulation

Hence

$$\dot{x} = \begin{bmatrix} O & I \\ O & -MJ \end{bmatrix} x + \begin{bmatrix} O \\ M \end{bmatrix} u \quad (5.2.18)$$

or

$$\dot{x} = A(x) x + B(x) u \quad (5.2.19)$$

where

$$A(x) = \begin{bmatrix} O & I \\ O & -MJ \end{bmatrix} \quad \text{and} \quad B(x) = \begin{bmatrix} O \\ M \end{bmatrix}$$

and

$$MJ = \frac{1}{\Delta} \begin{bmatrix} -2h(x)h_{22}(x)x_4 - h(x)h_{12}(x)x_3 & h(x)h_{22}(x)x_4 \\ 2h(x)h_{21}(x)x_4 + h(x)h_{11}(x)x_3 & h(x)h_{21}(x)x_4 \end{bmatrix}$$

$$\Delta = h_{11}(x)h_{22}(x) - h_{12}^2(x)$$

The parameters used for the simulations are illustrated in Table 5.1. Note that a solution to (5.2.19) exists if and only if H is invertible, i.e. H is non-singular.¹

¹For an analysis of the invertibility of H , refer to Appendix B

5.3 Linearized Model of the Two-Link Manipulator

Linearization is the process of finding a linear model that approximates a nonlinear one. As Lyapunov proved over a hundred years ago [24], if a small signal linear model is valid near an equilibrium and is stable, then there is a region (which may be small, of course) containing the equilibrium within which the nonlinear system is stable. So one can safely make a linear model and design a linear control for it such that, at least in the neighbourhood of the equilibrium, ones design will be stable.

Lyapunov's linearization method is concerned with the local stability of a nonlinear system and is a formalization of the intuition that a nonlinear system should behave similarly to its linearized approximation for small range motions. Hence, the two-link manipulator is linearized about a certain point (q_1, q_2) to study its linearized behaviour. As we shall see, the dynamics of the linearized model would actually depend only on q_2 . Mathematically, this means that for a given system

$$\dot{x} = f(x, u) \quad (5.3.1)$$

where $f(q_1, q_2) = 0$ and $f(x, u)$ is continuously differentiable, linearization of (5.3.1) about the given point results in the linear system

$$\dot{x} = A x + B u \quad (5.3.2)$$

where

$$A = \left. \frac{\partial f}{\partial x}(x, u) \right|_{q=[q_1 \ q_2]^T, u=0}; \quad B = \left. \frac{\partial f}{\partial u}(x, u) \right|_{q=[q_1 \ q_2]^T, u=0}$$

For the 2-link manipulator, we ended up with a simple model of the form

$$\begin{aligned} A &= \left[\begin{array}{c|cc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \hline 0 & 0 & h_{11} & h_{12} \\ 0 & 0 & h_{21} & h_{22} \end{array} \right]^{-1} \left[\begin{array}{c|cc} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \hline 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{array} \right] \\ &= \begin{bmatrix} I & O \\ O & H \end{bmatrix}^{-1} \begin{bmatrix} O & I \\ O & O \end{bmatrix} = \begin{bmatrix} O & I \\ O & O \end{bmatrix} \end{aligned} \quad (5.3.3)$$

and

$$\begin{aligned} B &= \left[\begin{array}{c|cc} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ \hline 0 & 0 & h_{11} & h_{12} \\ 0 & 0 & h_{21} & h_{22} \end{array} \right]^{-1} \left[\begin{array}{c} 0 & 0 \\ 0 & 0 \\ \hline 1 & 0 \\ 0 & 1 \end{array} \right] \\ &= \begin{bmatrix} I & O \\ O & H \end{bmatrix}^{-1} \begin{bmatrix} O \\ I \end{bmatrix} = \begin{bmatrix} O \\ H^{-1} \end{bmatrix} \end{aligned} \quad (5.3.4)$$

Note that h_{11} , h_{12} , h_{21} and h_{22} are now constants depending on the point of linearization.

We are now going to design various controllers for this simple 2-link manipulator model using the techniques studied previously in Part I of this thesis.

5.4 Position Control

In this section, we will look into the design and simulation of controllers to control the 2-link manipulator based on:

- \mathcal{H}_∞ controller,
- μ or Robust controller,
- LPV controller based on the “polytopic” approach.

Recall that the task is simply to move the manipulator to a given final position, as specified by a constant vector r of desired joint angles.

5.4.1 Mixed-sensitivity \mathcal{H}_∞ control

Mixed-sensitivity is the name given to transfer function shaping problems in which the sensitivity function $S = (I + GK)^{-1}$ is shaped along with one or more other closed-loop transfer functions such as KS or the complementary sensitivity function $T = I - S$. The approach is a direct and effective way of achieving multi-variable loop shaping. In the mixed-sensitivity problem formulation, nominal disturbance attenuation specifications, stability margin specifications as well as other specifications can be combined into a single infinity norm specification of the form :

$$\left\| \begin{bmatrix} W_1S \\ W_2KS \\ W_3T \end{bmatrix} \right\|_\infty$$

This is the mixed-sensitivity cost function, so called because it penalizes the sensitivity function, S , complementary sensitivity, T , and KS .

Figure 5.2 shows the perturbed system interconnection for the 2-link manipulator. The Δ block is the uncertain element which parameterizes all of the assumed model uncertainty in the problem.

The open-loop interconnection can then be represented as shown in Figure 5.3. The system labelled P is the open-loop interconnection and contains all of the known elements including the nominal plant model and performance and uncertainty weighting functions. The controller to be synthesized is K . Three sets of inputs enter P : perturbation inputs u_1 , disturbances u_2 , and controls u_3 . Three sets of outputs are generated: perturbation outputs y_1 , errors y_2 and measurements y_3 .

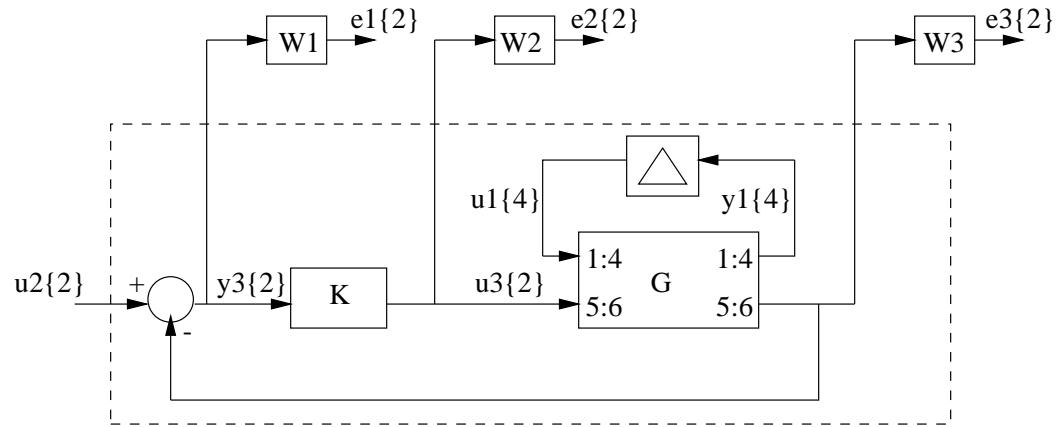


Figure 5.2: Perturbed System with associated weighting functions

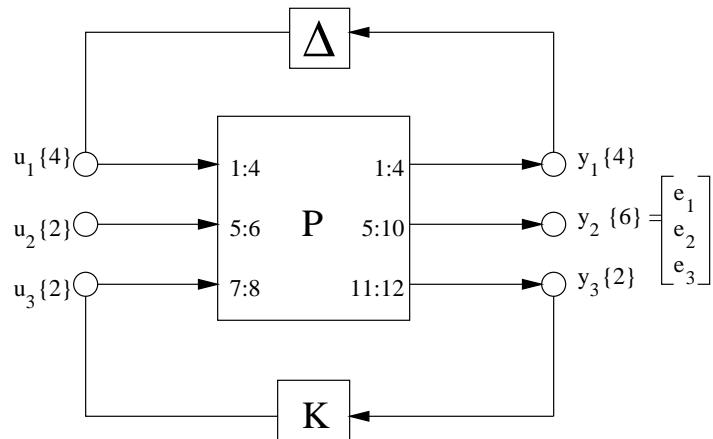


Figure 5.3: LFT Description of the System

The linearized two-link manipulator can be represented in descriptor form as

$$E \dot{x} = Ax + Bu \quad (5.4.1)$$

$$y = Cx + Du \quad (5.4.2)$$

where

$$\begin{aligned} E &= \begin{bmatrix} I & O \\ O & H \end{bmatrix} & A &= \begin{bmatrix} O & I \\ O & O \end{bmatrix} & B &= \begin{bmatrix} O \\ I \end{bmatrix} \\ C &= \begin{bmatrix} I & O \end{bmatrix} & D &= \begin{bmatrix} O \end{bmatrix} \end{aligned} \quad (5.4.3)$$

and

$$H = \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \quad I = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \quad O = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix} \quad (5.4.4)$$

The descriptor form is chosen to represent the model because of the possibility of allowing us to represent the parameters as linear dependent on the system and thus reduce the number of parameters required to represent the whole system which in turn helps with controller synthesis as explained in [25]. It is clear that the system definition depends primarily on E which in turn depends on H . H is a 2×2 matrix whose variables depend only on q_2 . Hence the parameter variations in the matrix E is due to variations in values of q_2 which is the angular position of arm 2 of the robot manipulator.

$$E = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & h_{11} & h_{12} \\ 0 & 0 & h_{21} & h_{22} \end{bmatrix}$$

and

$$h_{11} = a_1 + 2a_3 \cos q_2 + 2a_4 \sin q_2$$

$$h_{12} = h_{21} = a_2 + a_3 \cos q_2 + a_4 \sin q_2$$

$$h_{22} = a_2$$

where a_1, a_2, a_3 and a_4 are constants. Note that h_{22} is also a constant.

Taking out the sources of uncertainty leads us to a simplified model suitable for controller synthesis and analysis as shown below [25]

$$\begin{aligned} E &= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & a_1 & a_2 \\ 0 & 0 & a_2 & a_2 \end{bmatrix} \\ &+ \cos q_2 \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 2a_3 & a_3 \\ 0 & 0 & a_3 & 0 \end{bmatrix} + \sin q_2 \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 2a_4 & a_4 \\ 0 & 0 & a_4 & 0 \end{bmatrix} \end{aligned}$$

The weights are chosen as

$$\begin{aligned} W_1^{-1} &= \begin{bmatrix} \frac{1.25(s+0.007)^2}{(s+0.7826)^2} & 0 \\ 0 & \frac{1.25(s+0.007)^2}{(s+0.7826)^2} \end{bmatrix}; \quad W_3^{-1} = \begin{bmatrix} \frac{25}{(s+0.5)^2} & 0 \\ 0 & \frac{25}{(s+0.5)^2} \end{bmatrix} \\ W_2^{-1} &= \begin{bmatrix} \frac{0.0001(s+1e7)}{(s+10)} & 0 \\ 0 & \frac{0.0001(s+1e7)}{(s+10)} \end{bmatrix} \end{aligned}$$

The design is performed at the point $q_2 = 180$ deg.

The \mathcal{H}_∞ controller designed has 10 states and a $\gamma = 0.9877$ is achieved meaning that all the specifications are met. To verify this, the sensitivity S and the complementary sensitivity T are plotted and compared against weighting functions W_1^{-1} and

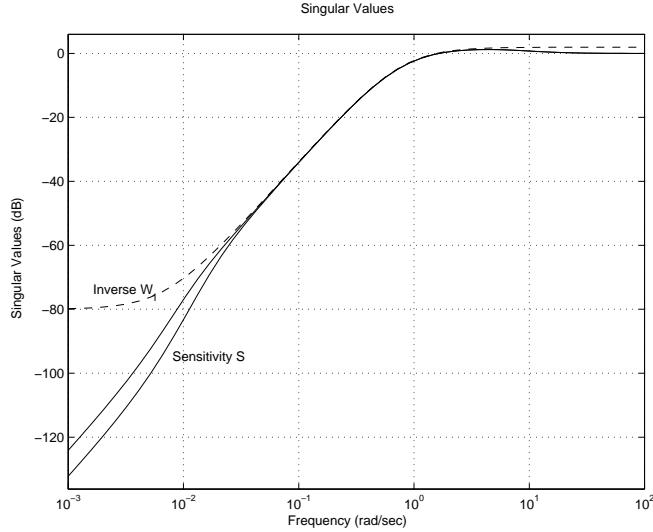


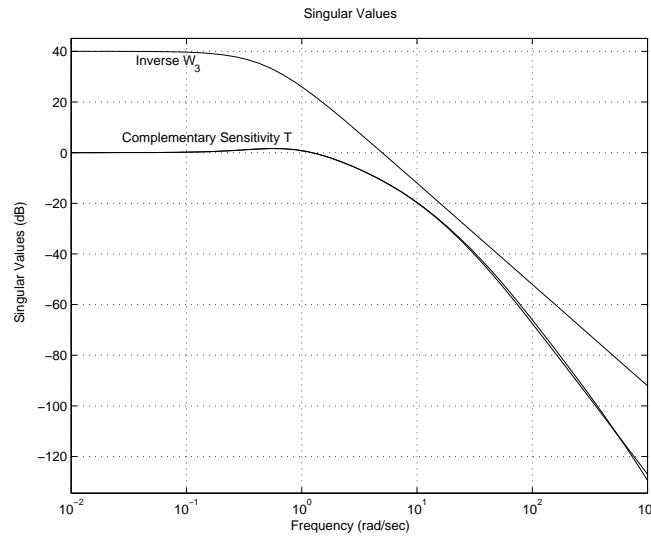
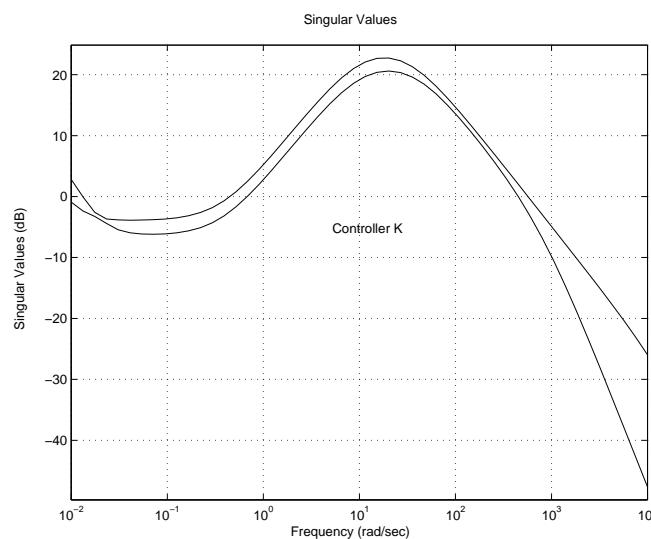
Figure 5.4: Sensitivity S and W_1^{-1}

W_3^{-1} respectively. The plots are shown in Figure 5.4 and Figure 5.5 respectively.

Clearly, both the specifications are met and therefore the controller is a suitable one for the robot model “at the point of design” or close to its vicinity.

Another issue of concern in a good design is the bandwidth of the controller K . Practically, very high bandwidth of the order of 10^5 Hz are very difficult to implement. The role of weighting function W_2 was to control the bandwidth. It was chosen to have a cross-over frequency of 10^3 Hz thus limiting the controller bandwidth to below that value. The singular value plot of the controller in Figure 5.6 shows that this specification is met as well.

The closed-loop system is now analyzed. Figure 5.7 shows the response and the control signal of the linearized closed-loop system to a step input of magnitude (0.01,0.01). As can be seen from the response, the settling time is about 7 seconds which is quite reasonable. Should a better performance be desired, it could be achieved by choosing a controller with a larger bandwidth, or in other words, by

Figure 5.5: Complementary sensitivity T and W_3^{-1} Figure 5.6: Singular value plot of the controller K

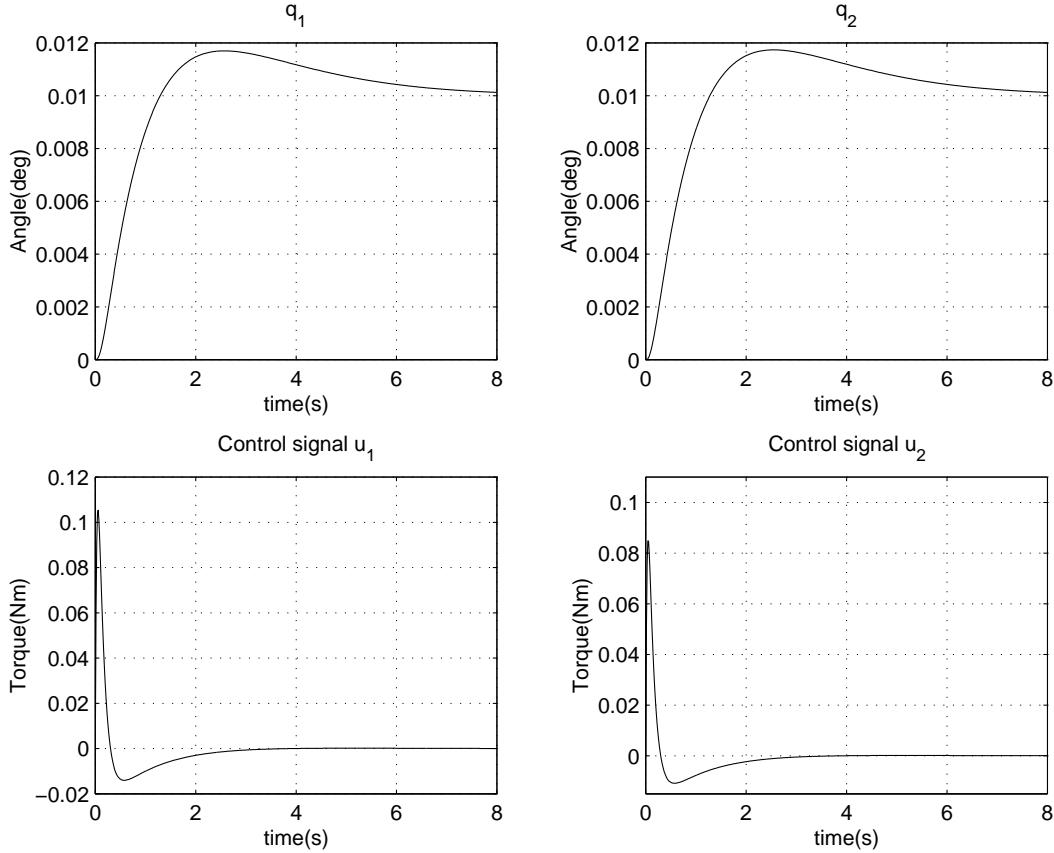


Figure 5.7: Step response and control signal of Linear system to a small signal at the point of design $q_2 = 180$ deg with \mathcal{H}_∞ Controller

relaxing weighting function W_2 . However, one has to bear in mind that the controller gain, $\bar{\sigma}(K)$, should be kept not too large in the frequency range where the loop gain is small in order not to saturate the actuators. That was the main aim in designing for a small bandwidth controller. From the plot, it can be seen that the control signal is also reasonable in the sense that it is not too large and decreases to zero very quickly.

To check for the versatility of the \mathcal{H}_∞ controller, we would also be interested in looking at the performance of the controller at different values of q_2 , which in the design was taken as some parametric uncertainty. As such, it is worthwhile looking

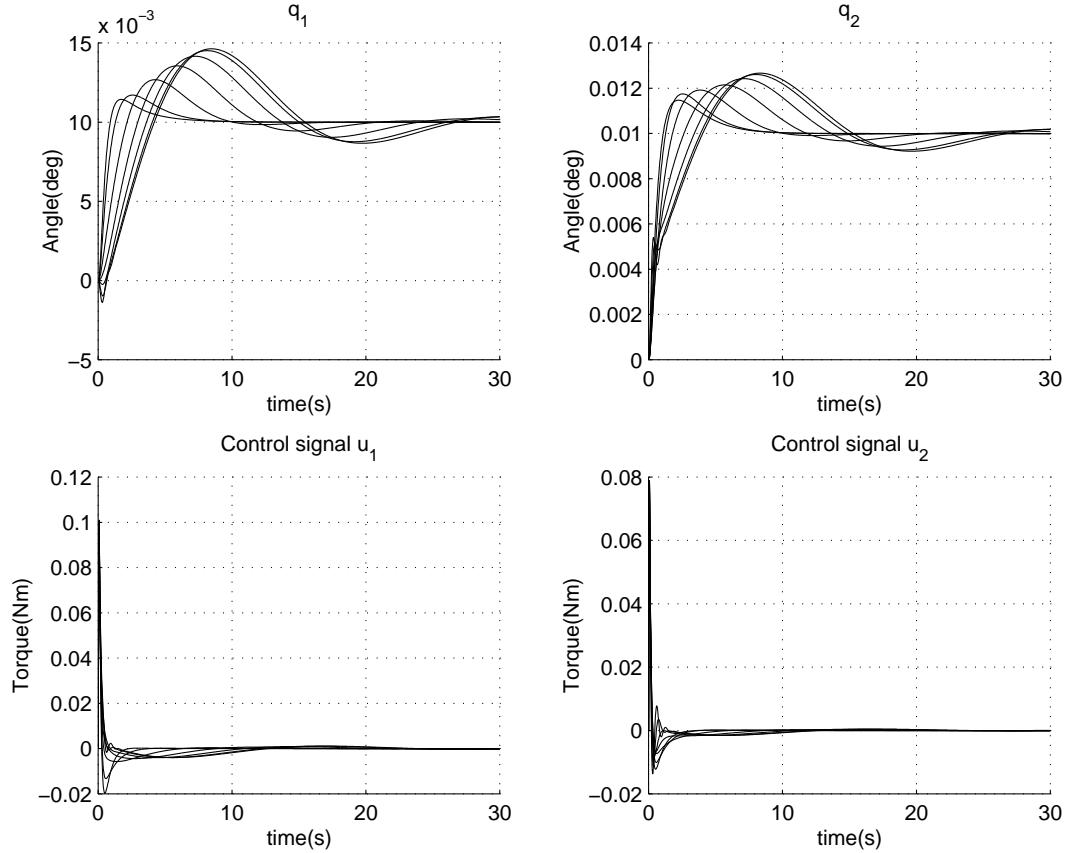


Figure 5.8: Small-signal step response and control signal for different q_2 using the same controller

at the different time-responses for different q_2 using the same controller designed at $q_2 = 180$ deg. The plot of the step responses and of the control signals are shown in Figure 5.8.

The time-response of the \mathcal{H}_∞ controller at points other than the \mathcal{H}_∞ point of design clearly shows that the controller performs very well and has a fair degree of versatility to it, i.e., it performs well for a large range of variation of the plant. The plots of the sensitivity S and complementary sensitivity T for different q_2 are also shown in Figure 5.9 and Figure 5.10 respectively.

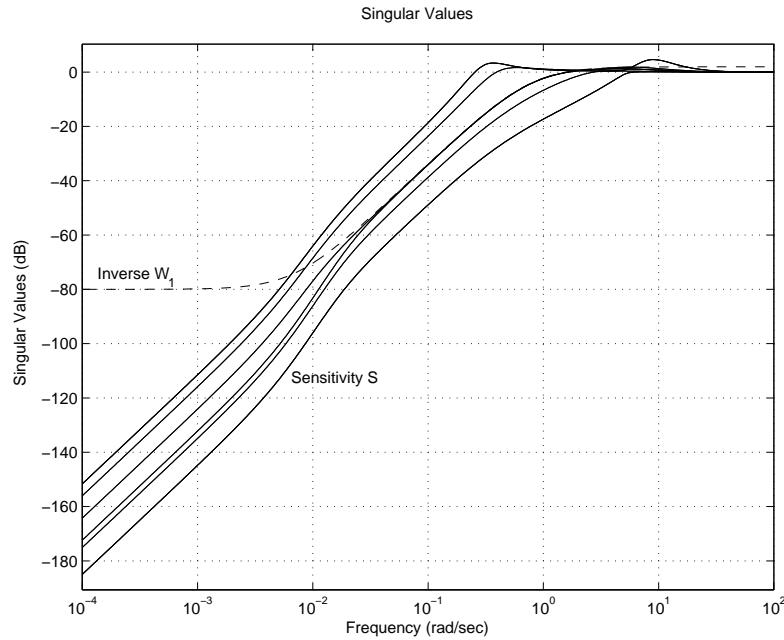


Figure 5.9: Sensitivity function S for different q_2 using the same controller

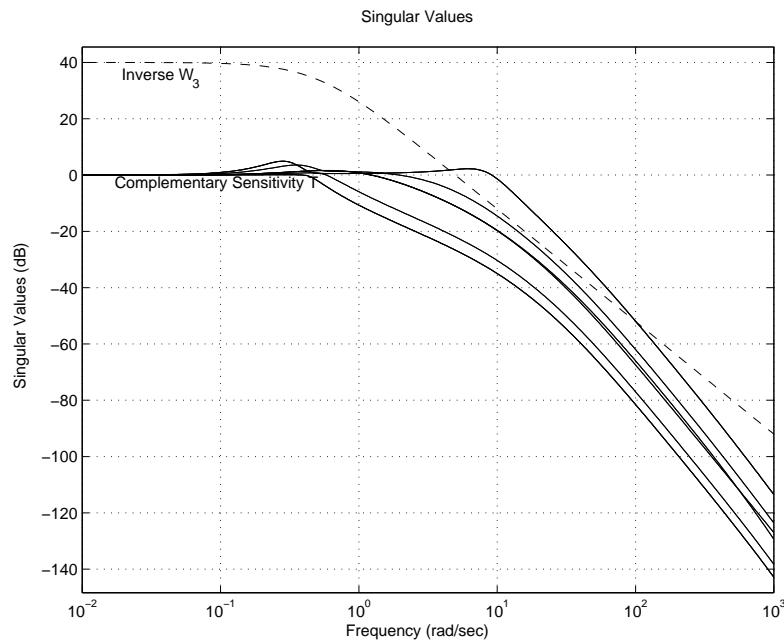


Figure 5.10: Complementary Sensitivity T for different q_2 using the same controller

However, performing well at various values of fixed q_2 and maintaining that performance when q_2 changes during the simulation itself are two different issues. This is what we are going to look at with the nonlinear simulation of the \mathcal{H}_∞ controller in the next section.

Simulation of Nonlinear System with H_∞ Controller

Finally, the nonlinear system with the H_∞ controller in place is simulated as shown in Figure 5.11. For consistency with previous results obtained, the manipulator initially at rest at $(q_1 = 0, q_2 = 0)$, is commanded to $(q_{d1} = 30 \text{ deg}, q_{d2} = 60 \text{ deg})$. Clearly, the controller performance is not all that good which is as expected since no bound of uncertainty on the parametric variation was assumed during the synthesis process. The controller was designed at a fixed value of q_2 .

5.4.2 μ -Synthesis using D - K Iteration

In this section, we will synthesize a controller using μ -Theory as discussed in Section 3.6.

The whole problem is specified as shown in Figure 5.12. The controller is designed on the linearized system, with two uncertainty boxes in place. The first one due to (δ_1, δ_2) represents the structured parametric uncertainty in the plant. The second uncertainty box Δ is put in place to represent high-frequency unmodelled dynamics always present in real-life applications.

For our 2-link manipulator model, we used the weighting functions (5.4.5) to

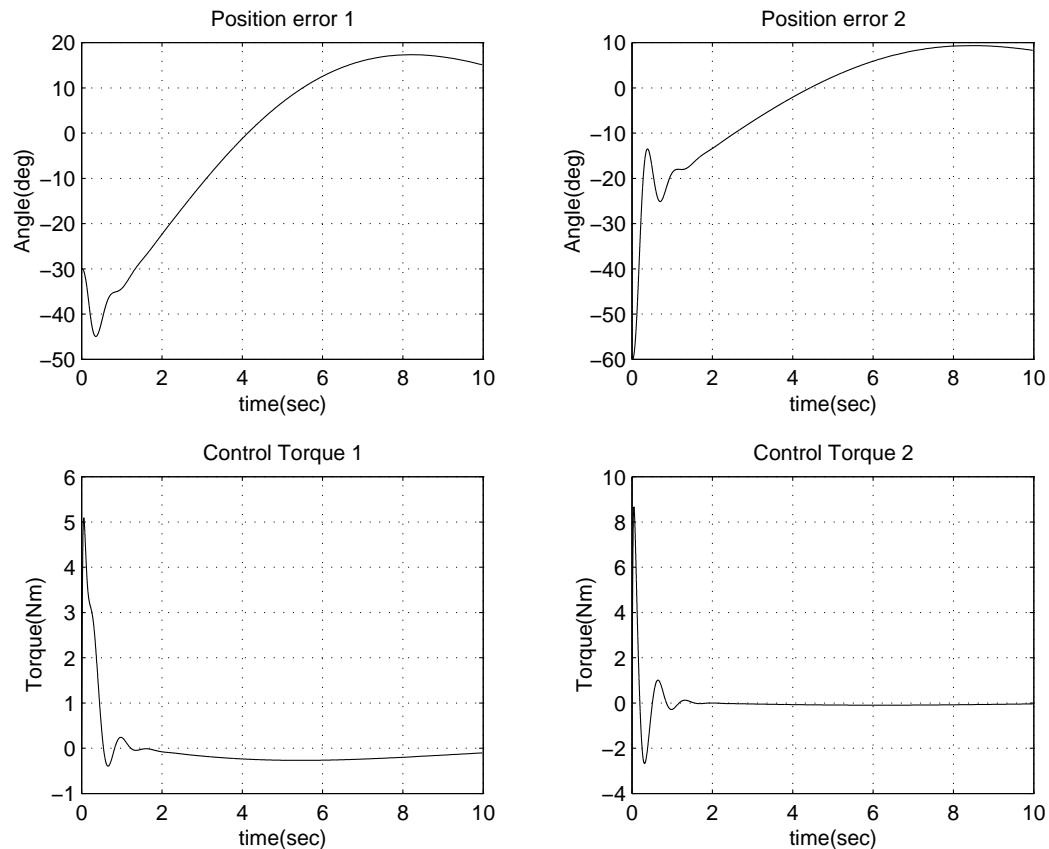


Figure 5.11: Nonlinear simulation with \mathcal{H}_∞ Controller

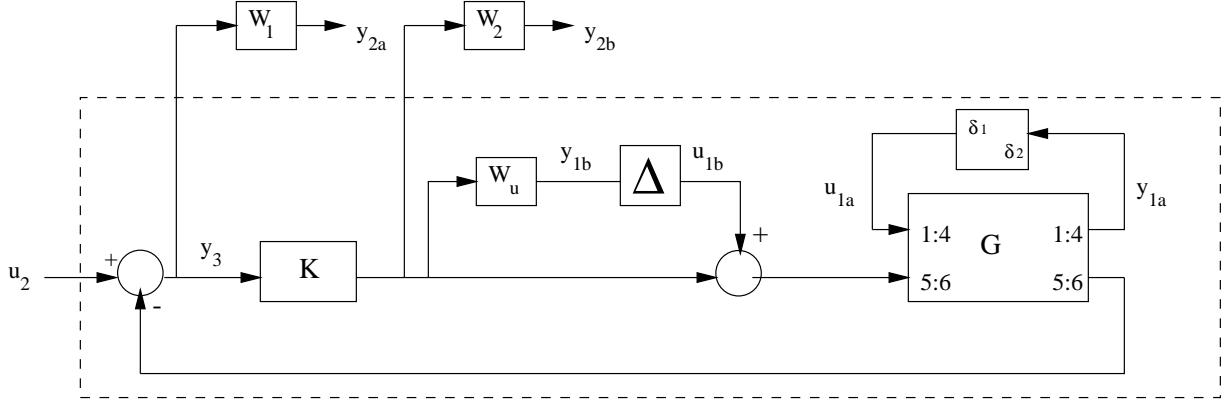


Figure 5.12: Perturbed System with Multiplicative Uncertainty used for μ -Synthesis

obtain reasonable robust stability and performance.

$$W_1^{-1} = \begin{bmatrix} \frac{4(s+0.003)^2}{(s+0.6)^2} & 0 \\ 0 & \frac{4(s+0.003)^2}{(s+0.6)^2} \end{bmatrix}; \quad W_2^{-1} = \begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix};$$

$$W_u^{-1} = \begin{bmatrix} \frac{0.01(s+200)^2}{(s+2)^2} & 0 \\ 0 & \frac{0.01(s+200)^2}{(s+2)^2} \end{bmatrix} \quad (5.4.5)$$

Using the μ -Toolbox [6], we came up with a 36th order controller which gave the minimum peak value of $\mu = 1.28$ at a frequency of $\omega = 4.94$ rad/sec.

μ -Analysis

This peak value can be confirmed by looking at Figure 5.13 which is a robust stability μ plot giving the lower and upper bounds on μ as a function of frequency. This value of μ is nowhere near the desired value of 1 which would ensure robust stability. If μ at a given frequency is different from 1, then the interpretation is that at this frequency we can tolerate $1/\mu$ times more uncertainty and still be stable with a margin of $1/\mu$. Clearly then, stability is not guaranteed for all perturbations and $\max_{\omega} \bar{\sigma}[\Delta(j\omega)] < \frac{1}{1.28} \approx 0.78$, meaning that the controller can only tolerate 78% of the plant uncertainty while maintaining stability.

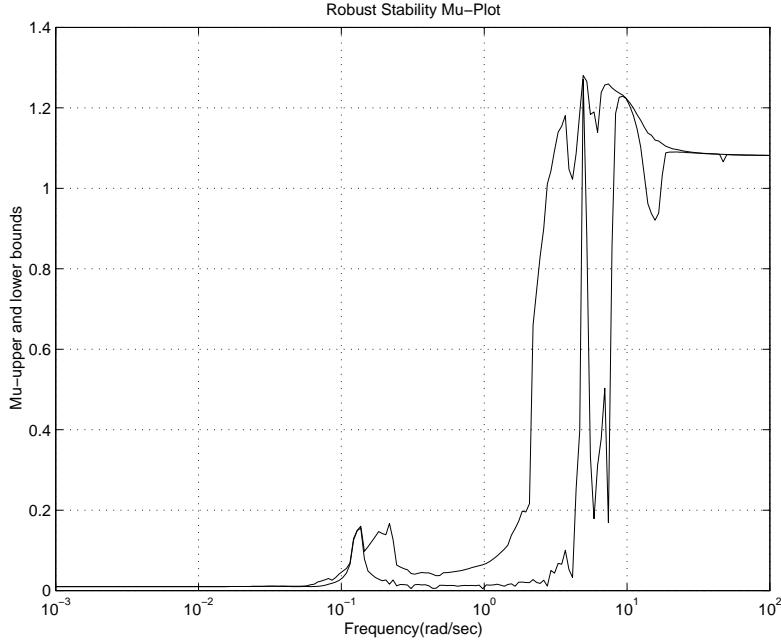
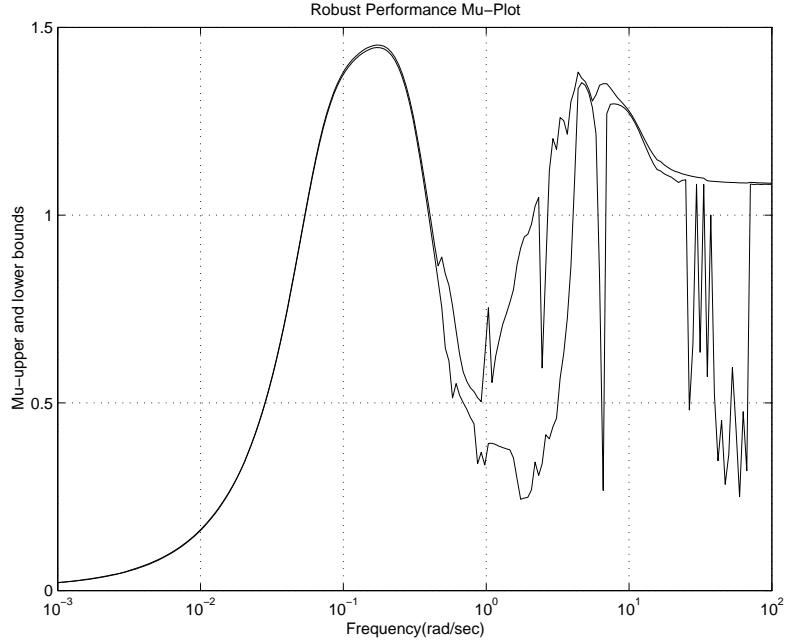


Figure 5.13: Robust Stability μ plot

Using the same analogy, we can analyse the performance of the μ -controller synthesized. The variation of the μ -upper and lower bound of performance with frequency can be seen in Figure 5.14. It can be said then that the performance objective has not been met at all frequencies. The peak value of $\mu = 1.45$ at the frequency $\omega = 0.17$ rad/sec means that the controller can ensure robust performance of the plant for only $(1/1.45 =)69\%$ of uncertainty present in the system.

In addition to determining if a system has robust performance to uncertainty, we also get the worst-case perturbation of a given size. Using perturbations of a particular structure Δ , and restricting to those of size $\leq \alpha$, we look at the worst performance possible as measured in $\|\cdot\|_\infty$ norm and also look at the perturbation that causes the largest degradation of performance. Given $\alpha > 0$, the worst-case

Figure 5.14: Robust Performance μ plot

performance associated with structured perturbations of size α is defined as

$$\mathcal{W}(M, \Delta, \alpha) = \max_{\substack{\Delta \in \Delta \\ \max_{\omega} \bar{\tau}[\Delta(j\omega)] \leq \alpha}} \|F_U(M, \Delta)\|_{\infty}$$

Plotting $\mathcal{W}(M, \alpha)$ versus α yields the worst-case performance tradeoff curve, which shows the tradeoff between size of uncertainty and worst-case performance. However, since our system can only handle 78% of the uncertainty present, it makes no sense to try to simulate the performance at a value of $\alpha > 0.78$. The performance degradation curves are as shown in Figure 5.15.

Finally, we compute the worst-case perturbation of size 0.78 and simulate the worst-case performance time response using this worst-case perturbation. Figure 5.16 shows the step response and the control signal obtained with a perturbation whose infinity norm lies between 0.774 and 0.775. The $\|F_U(M, \Delta_{bad})\|_{\infty} \approx 4.79$ and occurs at 4.88 rad/sec.

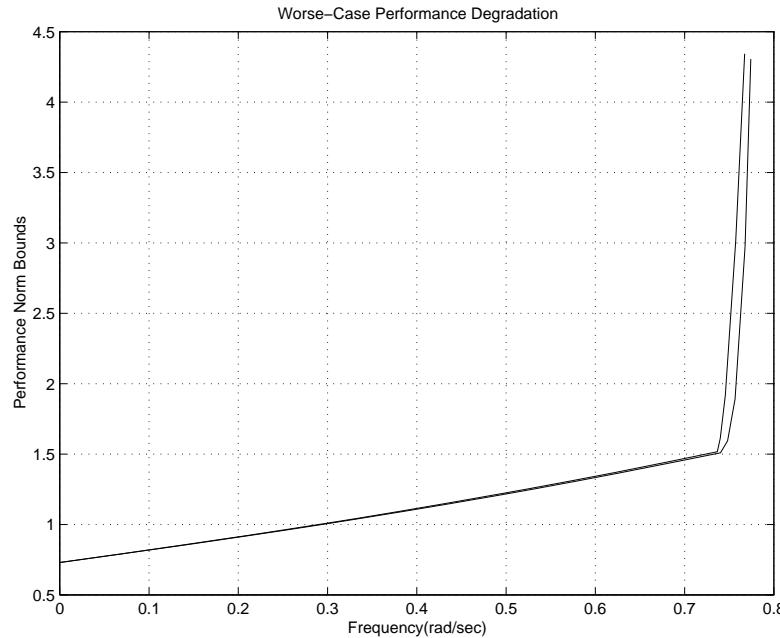


Figure 5.15: Worst-Case Performance Degradation

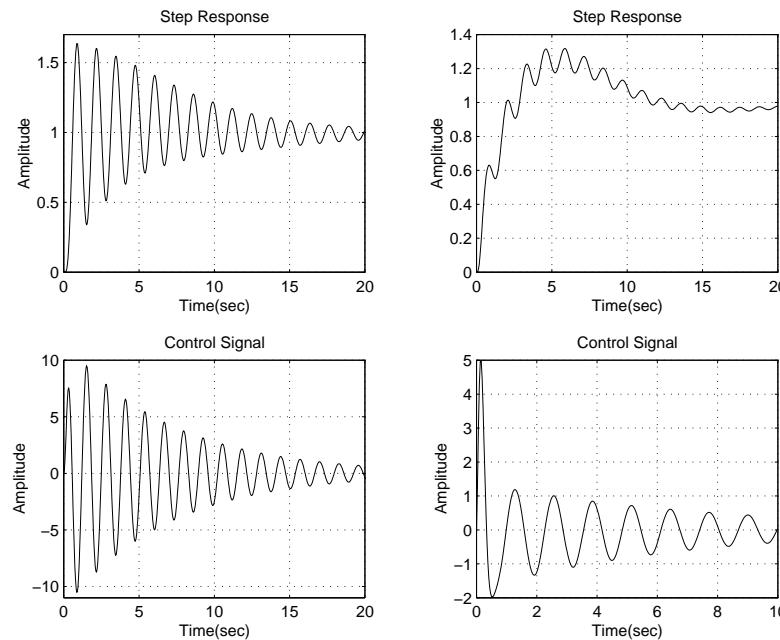


Figure 5.16: Step response and control signal for worst-case perturbation

All these analysis that were performed by the μ -Toolbox [6] assumed that the parametric uncertainties δ_1 and δ_2 to be not repeated as the software is incapable of dealing with repeated parametric blocks. This has introduced a fair bit of conservatism into the analysis. Furthermore, we have the 2 parametric uncertainty blocks with elements $\delta_1 = \cos q_2$ and $\delta_2 = \sin q_2$, implying that we should not consider cases which in actual fact are not possible. For example, in analyzing the controller performance, we cannot consider cases where $\delta_1 = \delta_2 = 1$ which is not possible since there is no value of q_2 at which $\cos q_2$ and $\sin q_2$ are both 1. A better analysis of the controller is therefore warranted in the absence of suitable analysis tools. What we do is, with the μ -controller in place, we take different values of q_2 and calculate the worse-case perturbation at each of these values of q_2 using only the full complex uncertainty block. What we have done here is get rid of two 2×2 uncertainty block representing the parametric uncertainty. We then plot $\|F_U(F_L(P, K), \Delta)\|_\infty$ against q_2 .

Figure 5.17 shows the new worst-case performance bound obtained and it can be seen from the plot that the worst-case occurs when $q_2 = 30$ deg and the performance bound peaks at 1.51 at that point.

Similarly, a new analysis is performed to check for robust stability by allowing q_2 to take different values and then doing a μ -analysis at each of those points taking only the 2×2 complex block as the uncertainty present. This will enable us to compare this result with the conservative analysis done earlier using all three 2×2 uncertainty blocks. Figure 5.18 shows the worst-case robust stability bound for this controller at different values of q_2 . The peak value is 0.30 and occurs when $q_2 = 30$ deg. Clearly then, with this controller, and using this method of analysis, we can conclude

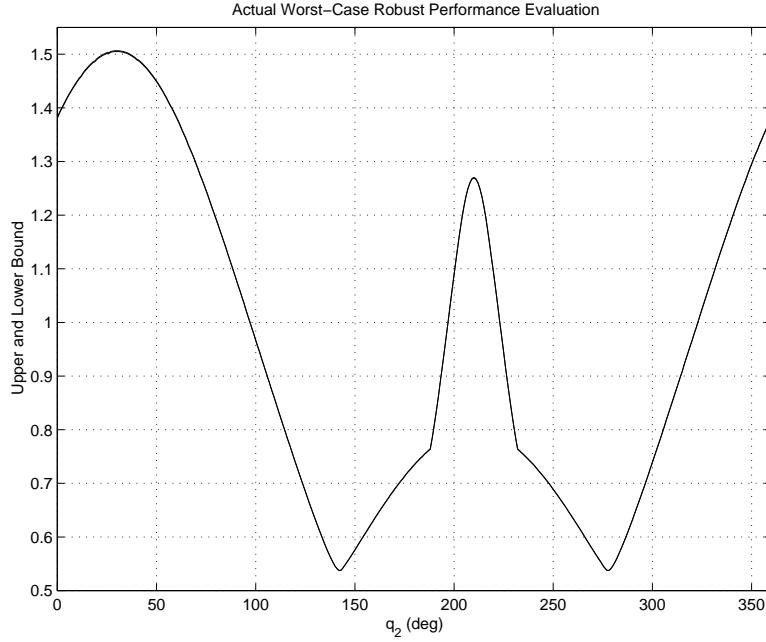


Figure 5.17: Actual Worst-Case Performance Bound at Different q_2

that we have achieved robust stability which was not the case using the analytical methods used previously. Moreover, our controller can tolerate $(1/0.3 \approx) 330\%$ more uncertainty while being stable.

It can also be shown that the actual specifications are just slightly not satisfied for some values of q_2 on the linearized plant. Figure 5.19 is a plot showing the weighting function W_1^{-1} and sensitivity S at different values of q_2 . Finally, to sum it all up, we look at the nonlinear behaviour of the plant with the μ -controller in place. This is shown in Figure 5.20 where the manipulator is commanded to move from (30,60) to (45,30). It can be deduced that the controller performs very well and even though we have not achieved robust performance for all q_2 , we can say that the controller designed is a very satisfactory one, providing fairly good performance all throughout and not exceeding the margin of robust performance by much.

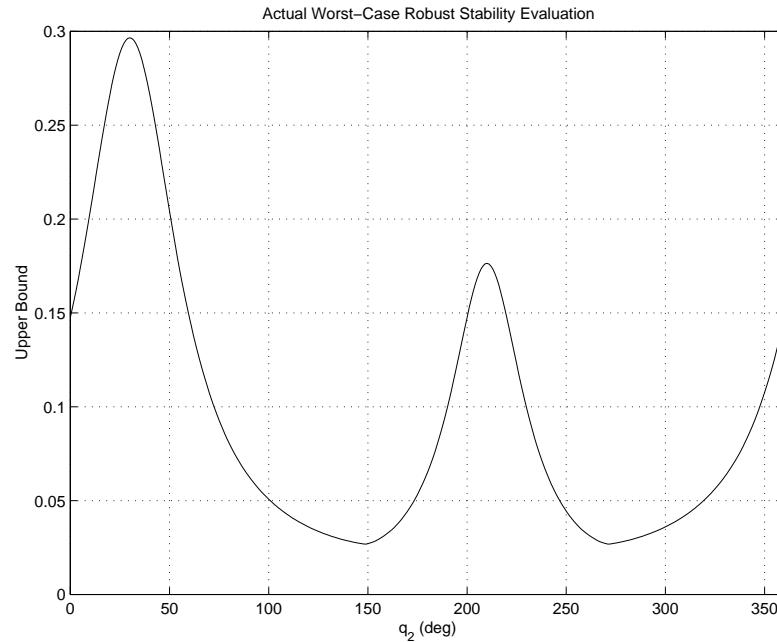
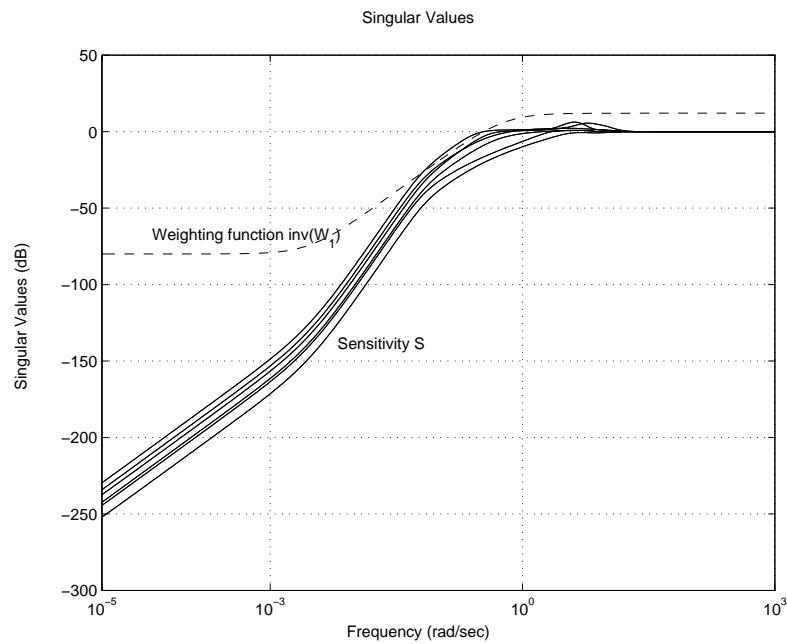


Figure 5.18: Actual Worst-Case Stability Bound

Figure 5.19: Sensitivity S and Weighting function W_1^{-1}

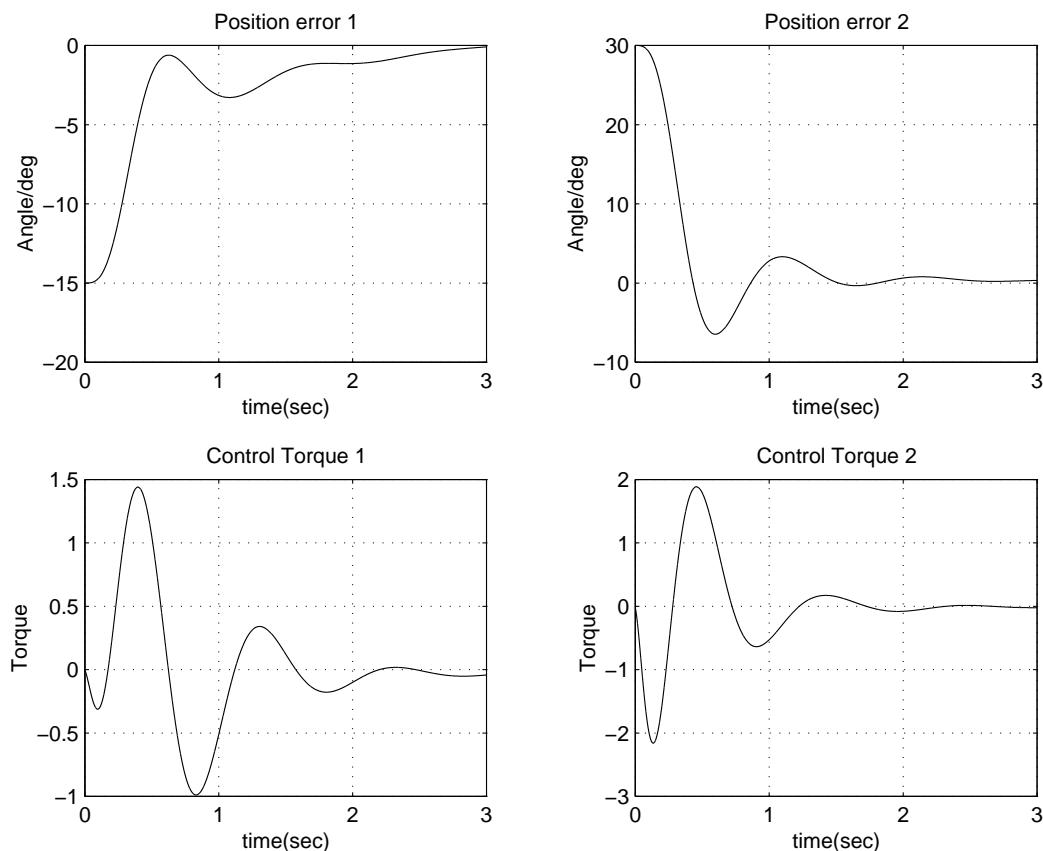


Figure 5.20: Nonlinear simulation of μ -controller when commanded to move from (30,60) to (45,30)

5.4.3 Gain-Scheduled Controller Design

For the nonlinear manipulator, we reformulate our model using the following substitutions [25],

$$\begin{aligned} h_{11} &= a_1 + 2a_3 \cos x_2 + 2a_4 \sin x_2 \\ &= a_1 + 2m_1(x_2), \end{aligned} \quad (5.4.6)$$

$$\text{where } m_1(x_2) = a_3 \cos x_2 + a_4 \sin x_2.$$

$$\begin{aligned} h_{12} &= a_2 + a_3 \cos x_2 + a_4 \sin x_2 \\ &= a_2 + m_1(x_2). \end{aligned} \quad (5.4.7)$$

$$h_{21} = h_{12}.$$

$$h_{22} = a_2. \quad (5.4.8)$$

$$\text{and } h = a_3 \sin x_2 - a_4 \cos x_2$$

$$= m_2(x_2), \quad (5.4.9)$$

$$\text{where } m_2(x_2) = a_3 \sin x_2 - a_4 \cos x_2. \quad (5.4.10)$$

then from

$$\begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix} \begin{bmatrix} \ddot{x}_1 \\ \ddot{x}_2 \end{bmatrix} + \begin{bmatrix} -h\dot{x}_2 & -h(\dot{x}_1 + \dot{x}_2) \\ h\dot{x}_1 & 0 \end{bmatrix} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} u_1 \\ u_2 \end{bmatrix} \quad (5.4.11)$$

and since

$$x = \begin{bmatrix} x_1 & x_2 & x_3 & x_4 \end{bmatrix}^T = \begin{bmatrix} q_1 & q_2 & \dot{q}_1 & \dot{q}_2 \end{bmatrix}^T \quad (5.4.12)$$

the dynamics of the two-link manipulator can be rewritten as

$$[a_1 + 2m_1] \dot{x}_3 + [a_2 + m_1] \dot{x}_4 - m_2 x_3 x_4 - m_2(x_3 + x_4)x_4 = u_1 \quad (5.4.13)$$

$$[a_2 + m_1] \dot{x}_3 + a_2 \dot{x}_4 + m_2 x_3^2 = u_2 \quad (5.4.14)$$

$$\dot{x}_1 = x_3 \quad (5.4.15)$$

$$\dot{x}_2 = x_4 \quad (5.4.16)$$

The model can then be expressed as

$$E(p)\dot{x} = A(p)x + Bu \quad (5.4.17)$$

where

$$E(p) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & a_1 + 2m_1 & a_2 + m_1 \\ 0 & 0 & a_2 + m_1 & a_2 \end{bmatrix}; \quad A(p) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & m_2(2x_3 + x_4) \\ 0 & 0 & -m_2 x_3 & 0 \end{bmatrix}$$

Let

$$p_1 = m_1(x_2); \quad p_2 = m_2(x_2)x_3; \quad p_3 = m_2(x_2)x_4$$

then

$$E(p) = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & a_1 & a_2 \\ 0 & 0 & a_2 & a_2 \end{bmatrix} + p_1 \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 2 & 1 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

$$A(p) = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \end{bmatrix} + p_2 \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 2 \\ 0 & 0 & -1 & 0 \end{bmatrix} + p_3 \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$

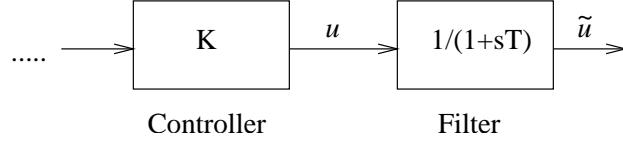


Figure 5.21: Pre-filtering of control signal

Recall that the system matrix for a polytopic system has the form:

$$S(p) := \left(\begin{array}{c|cc} A(p) & B_1(p) & B_2 \\ C_1(p) & D_{11}(p) & D_{12} \\ \hline C_2 & D_{21} & D_{22} \end{array} \right)$$

In order to design a gain-scheduled controller on this system, it is a prerequisite that B_2, C_2, D_{12} and D_{21} not be affine parameter varying. To ensure that to be the case, the control signal is pre-filtered through a low-pass filter with sufficiently large bandwidth before being fed to the plant as shown in Figure 5.21

$$\frac{\tilde{u}}{u} = \frac{1}{1 + sT} \quad (5.4.18)$$

Letting $z = \tilde{u}$, we end up having

$$\begin{aligned} E\dot{x} &= Ax + Bz \\ \dot{z} &= -\frac{1}{T}z + \frac{1}{T}u \end{aligned} \quad (5.4.19)$$

The final model looks like

$$\begin{bmatrix} E(p) & 0 \\ 0 & I \end{bmatrix} \begin{bmatrix} \dot{x} \\ \dot{z} \end{bmatrix} = \begin{bmatrix} A(p) & B \\ 0 & -\frac{1}{T} \end{bmatrix} \begin{bmatrix} x \\ z \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{T} \end{bmatrix} u \quad (5.4.20)$$

Note that the selection of T is very much a matter of personal choice depending on the type of input signal the system is expected to handle.

For the robot model, a suitable value of $T = 0.001$ is used giving a bandwidth of about 1000 Hz. The control structure is set as shown in Figure 5.22. The same

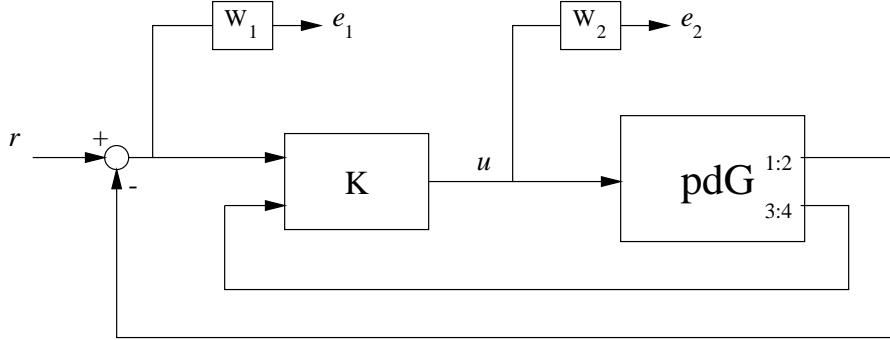


Figure 5.22: Control structure for gain-scheduled controller design

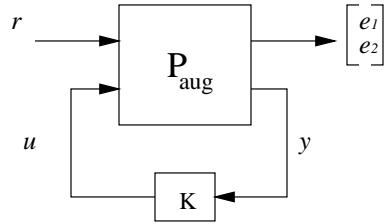


Figure 5.23: Augmented plant interconnection

weighting functions as used with the μ -controller design are used for gain-scheduled controller design.

$$W_1^{-1} = \begin{bmatrix} \frac{1.25(s+0.007)^2}{(s+0.7826)^2} & 0 \\ 0 & \frac{1.25(s+0.007)^2}{(s+0.7826)^2} \end{bmatrix}; W_2^{-1} = \begin{bmatrix} \frac{0.0001(s+1e7)}{(s+10)} & 0 \\ 0 & \frac{0.0001(s+1e7)}{(s+10)} \end{bmatrix}$$

The augmented plant interconnection is shown in Figure 5.23. Also, the range of variation of the parameters p_1, p_2 and p_3 are taken as shown in Table 5.2.

No bound is placed on the rate of change of these parameters during the synthesis.

Parameter	Bound
$ p_1 $	1.2
$ p_2 $	6
$ p_3 $	6

Table 5.2: Bounds on parameters for synthesis

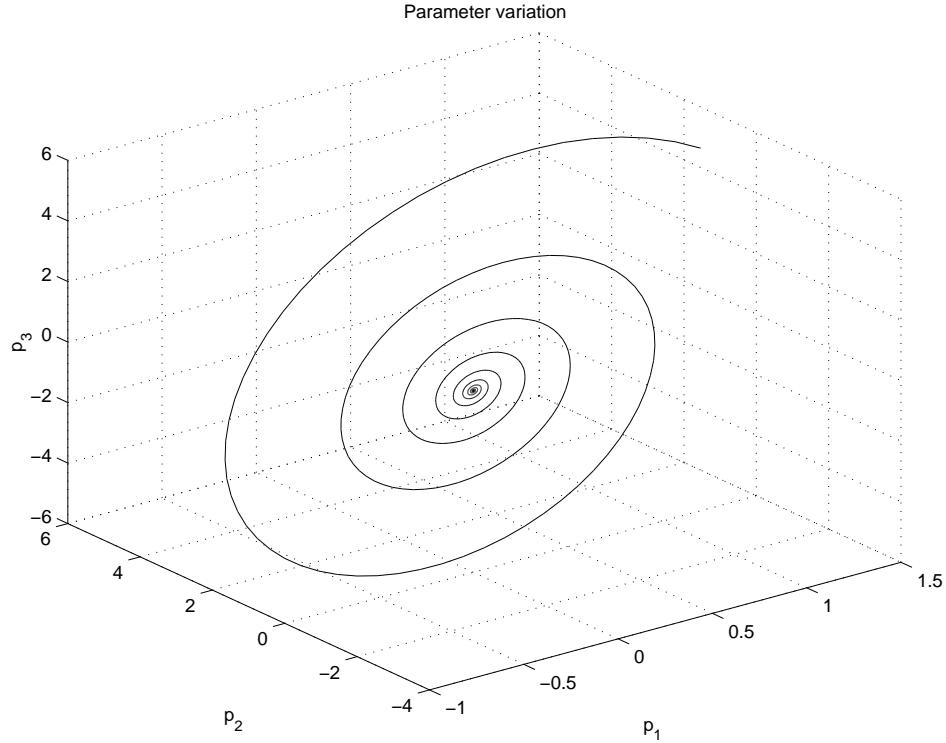


Figure 5.24: Variation of parameters used for simulation

The parameters are allowed to vary along the following spiral trajectories as shown in Figure 5.24.

$$p_1 = 1.2e^{-10t} \cos(100t);$$

$$p_2 = 6e^{-10t} \sin(100t);$$

$$p_3 = 6e^{-10t} \cos(100t).$$

The controller is then designed using the ‘`hinfgsx`’ command which is a slightly modified version of ‘`hinfgs`’². The command gave rise to a polytopic controller with 8 vertex systems, each system having 12 states, 4 inputs and 2 outputs. A γ value of 1.87 is achieved with the present weighting functions implying that all the performance

²The modification was necessary to avoid numerical computational problems, `sx1` was changed to $1e^{-9}$ on line 85

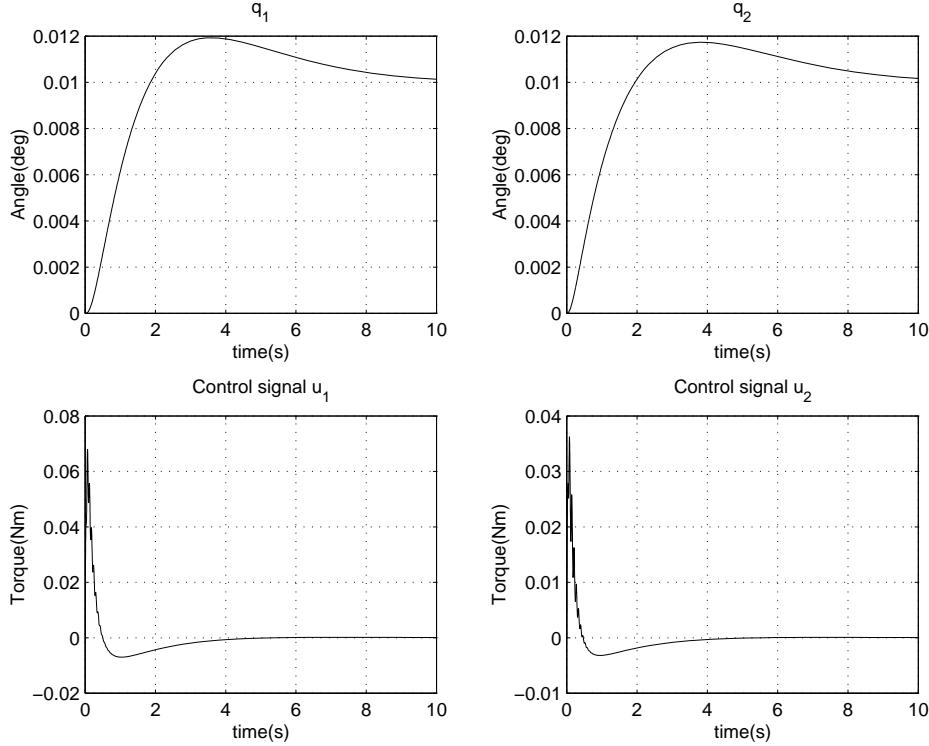


Figure 5.25: Step Response of the system with the gain-scheduled controller to a small signal of $(0.01, 0.01)$

specifications are not met. The parameter dependent system is then simulated using the “`pdsimul`” command. The step response when subjected to a small signal of $(0.01, 0.01)$ is shown in Figure 5.25.

The stability of the parametric-dependent closed loop system is also investigated using the “`quadstab`” command. This command seeks a single Lyapunov matrix $Q > 0$ such that

$$A(p) Q E(p)^T + E(p) Q A(p)^T < 0 \quad (5.4.21)$$

over the range of values of the parameter vector p . The Lyapunov function

$$V(x) = x^T P x \text{ with } P = Q^{-1} \quad (5.4.22)$$

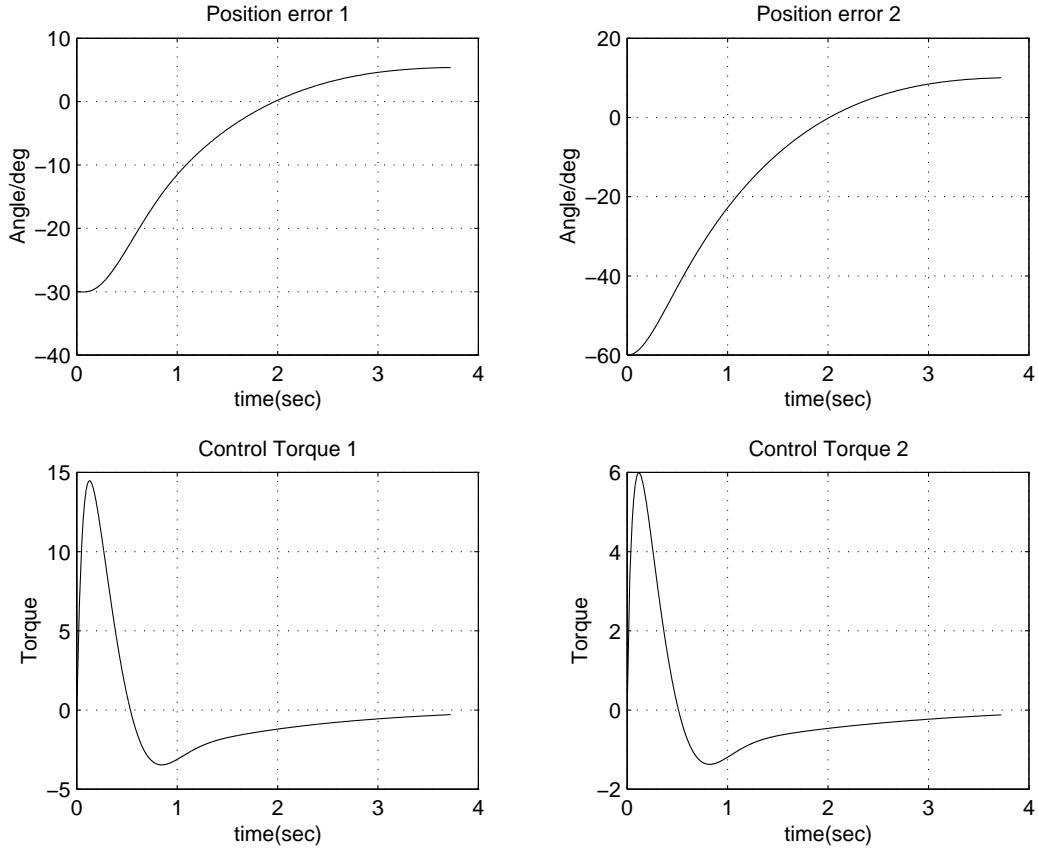


Figure 5.26: Nonlinear simulation with gain-scheduled controller

then establishes stability over the entire parameter range and for arbitrarily fast parameter variations. For our case, such a Q is found implying that the closed loop system is stable for all parametric variations.

More importantly, the nonlinear simulation of the manipulator with the LPV controller is shown in Figure 5.26. The robot arm is commanded to move from $(0,0)$ to $(30,60)$.

The nonlinear simulation of the 2-link manipulator is proof of its superior performance capability. The response is very smooth and the control signal is by no means excessive. Unfortunately it has not been simulated for long enough due to

processor capability limitation, the step size required for this simulation to converge being extremely small. It took 3 days to complete on a Pentium III 450 MHz.

5.4.4 Comparison of Discussed techniques

To compare the different techniques studied, a comparison of their nonlinear simulation is made for a particular operating condition. The manipulator arm is moved from (0,0) to (30,60) by the use of three different controllers.

- H_∞ Controller
- μ Controller
- Linear Parameter Varying(LPV) Controller

Figure 5.27 shows the nonlinear simulation of the system for the 3 different controllers.

The μ controller has 12 states (much less than that of the μ controller(36 states) and is globally stable for the whole operating range provided the parametric bounds assumed are not violated. The H_∞ controller, even though, being more easily implementable with 10 states provides poor performance at best. The only drawback one can think of the LPV controller is that it needs to be updated each time the robot arm moves as it is dependent on the position and velocity of the arm. The faster this information is sent to a processor and the faster the controller is updated, the better the performance. One can safely conclude that the LPV controller is much better than the μ and H_∞ controllers in terms of performance and guaranteed stability.

It can be argued that a simple Proportional-Derivative (P.D) controller could have done the job while at the same time guarantee the global stability of the system. Why the trouble then of going through all this ? First of all, with a P.D controller, the

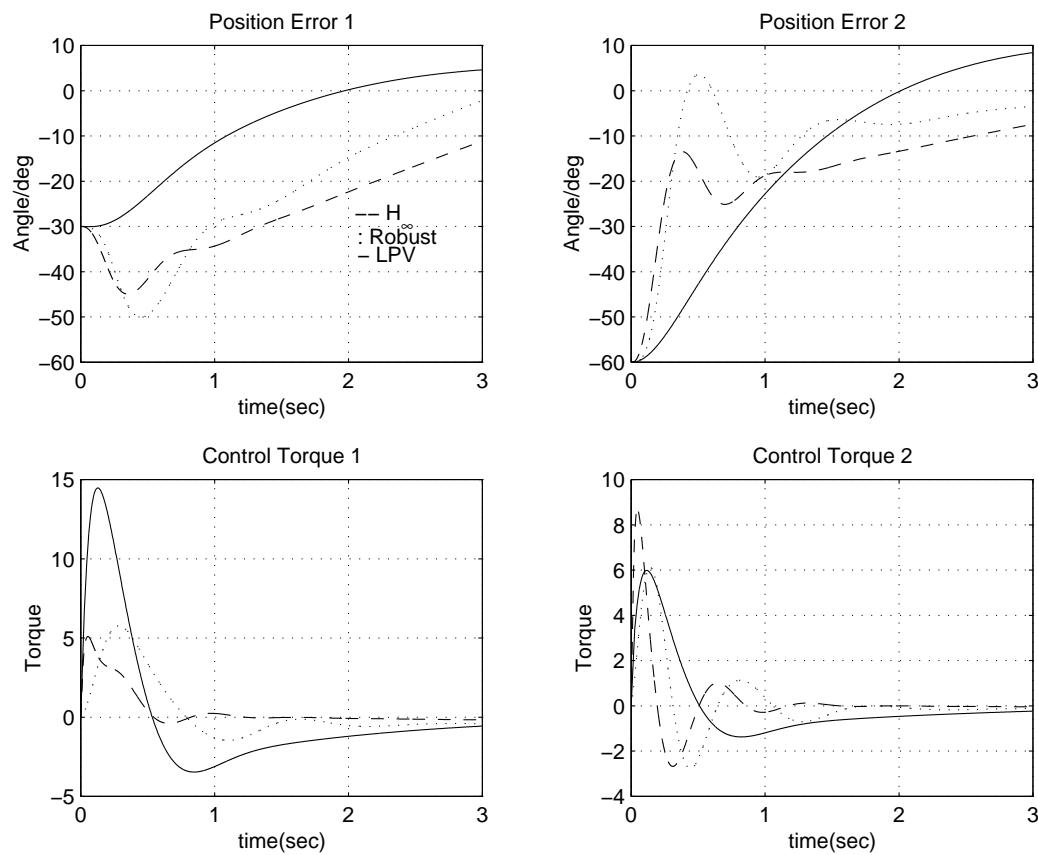


Figure 5.27: Comparison of different controllers : \mathcal{H}_∞ (dashed), Robust (dotted), LPV (solid)

performance and robustness specifications cannot be clearly specified. On top of that, we have very little influence over the control signal which more often than not tends to be excessive causing actuator saturation. This has been studied and found to be the case but has not been included in this thesis. Having said that, our main objective in introducing this 2-link robot manipulator model was to investigate various robust and gain-scheduled control theory. In other case studies discussed later, things are quite different and simple state feedback approach will not be an option to controlling the nonlinear systems. Those are cases where one can really appreciate the advantages of robust and gain-scheduled control.

5.5 μ -Controller Reduction

In this section, we reduce the μ -controller designed in Section 5.4.2 and simulate the system with the new controller and compare that with the full-order controller. Design methods such as μ -synthesis produce controllers of order much higher than that of the plant, because of the inclusion of the scaling matrices. These control laws may be too complex with regards to practical implementation and simpler designs are then sought. For this purpose, one can either reduce the order of the plant prior to controller design, or reduce the controller in the final stage, or both. Here we will consider reducing the controller in the final stage and we will be using what is known as the Balanced Model Reduction via truncated and Schur Methods [61].

The stability of these reduced controllers are also investigated and it is also found that with the new controllers, the system is stable for all values of q_2 as was the case with the full order controller.

The 36th order controller obtained is simulated for (q_1, q_2) moving from (30,60) to

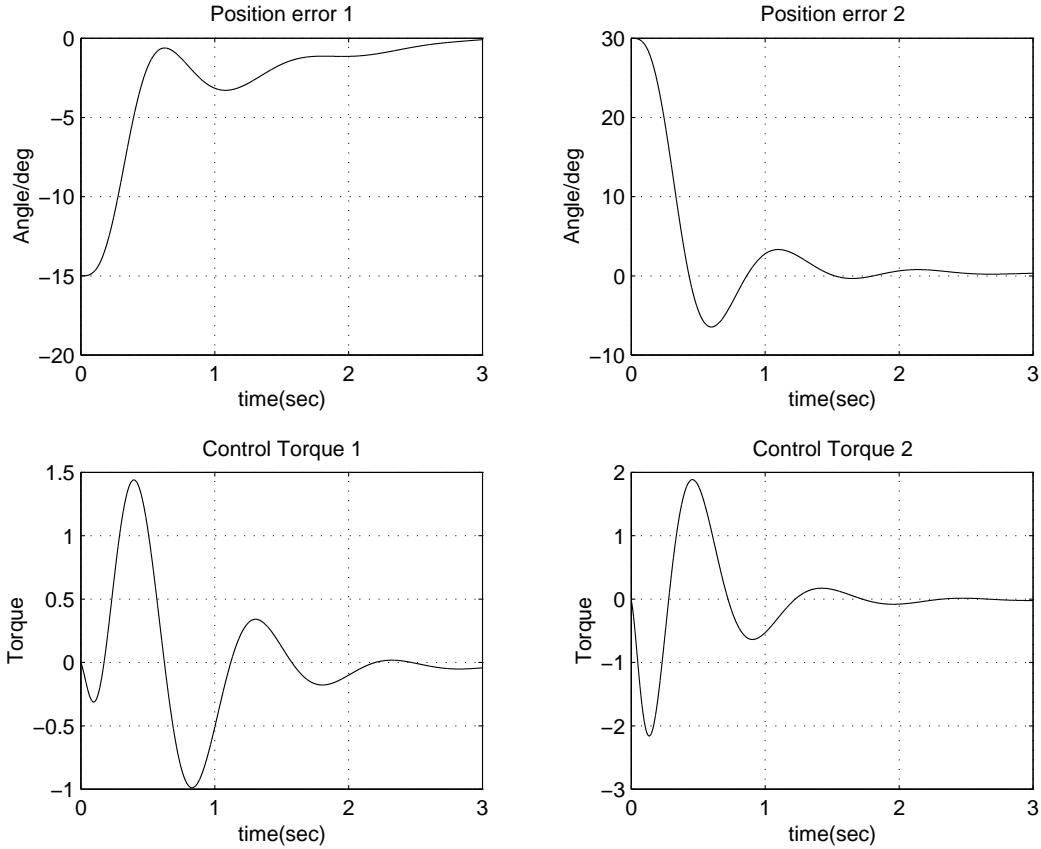


Figure 5.28: Response of nonlinear system with 36th order controller

(45,30). Then the 36th order controller is reduced, first to a 15th order controller and then to a 10th order controller and the response simulated for each one of them.

The performance seems to be maintained with the 15th order controller and after further analysis, it can be used as an alternative to the full-order controller if implementation of high-order systems is an issue of concern. However the performance degrades when we reduce the full-order controller further to a 10th order controller and we start having constant steady-state error.

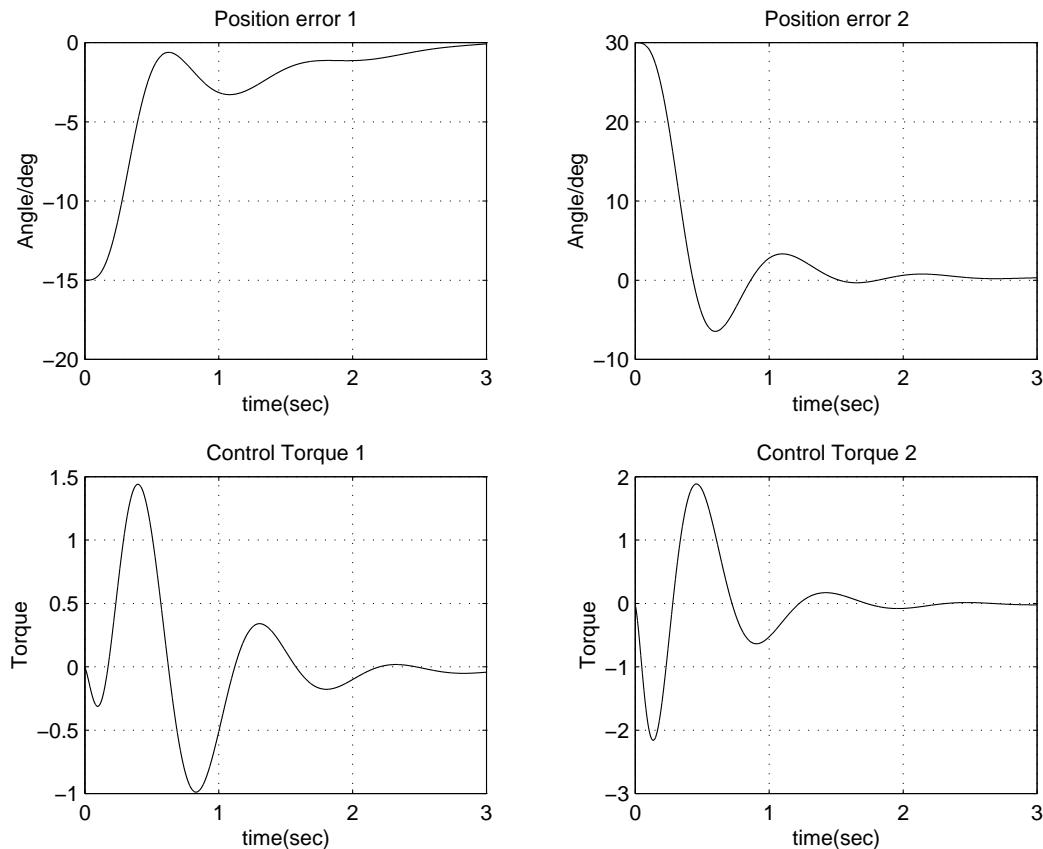
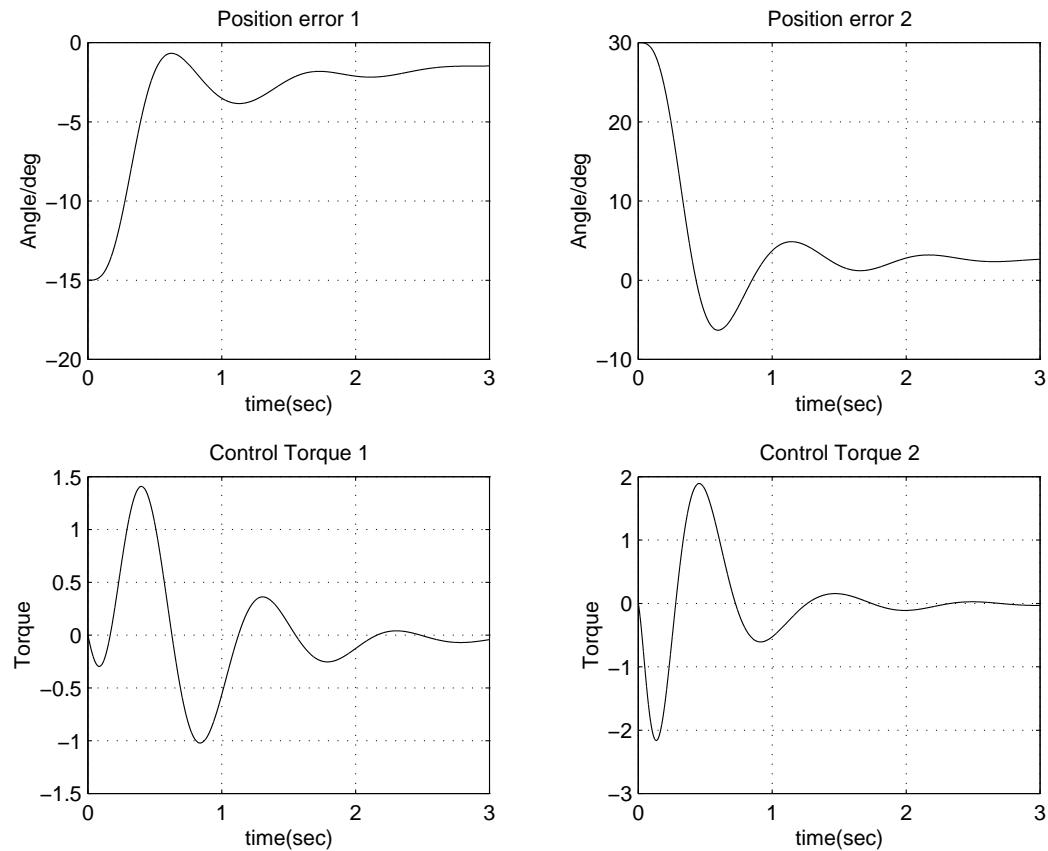


Figure 5.29: Response of nonlinear system with 15thorder controller

Figure 5.30: Response of nonlinear system with 10thorder controller

Chapter 6

RTAC Benchmark Problem for Nonlinear Control Design

6.1 Introduction

This problem provides the benchmark for examining nonlinear control design techniques within the framework of a non-linear fourth-order dynamical system. It involves the nonlinear interaction of a translational oscillator and an eccentric rotational proof mass [12].

6.2 Problem Statement

The system consists of a cart of mass M connected to a fixed wall by a linear spring of stiffness k as shown in Figure 6.1. The cart is constrained to have one-dimensional travel. A proof-mass actuator attached to the cart has mass m and moment of inertia I about its centre of mass, which is located a distance e from the point about which the proof mass rotates. The motion occurs in a horizontal plane, so that no gravitational forces need to be considered. N denotes the control torque applied to the proof mass, and F is the disturbance force on the cart.

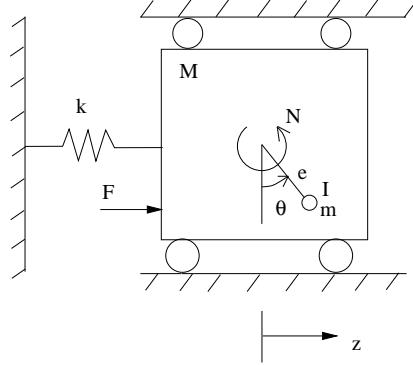


Figure 6.1: Rotational Actuator to control a translational oscillator

Let z and \dot{z} denote the translational position and velocity of the cart, and let θ and $\dot{\theta}$ denote the angular position and velocity of the rotational proof mass, where $\theta = 0$ is perpendicular to the motion of the cart, and $\theta = 90$ deg is aligned with the positive z direction. The equations of motion are given in [12] by

$$(M + m)\ddot{z} + kz = -me(\ddot{\theta} \cos \theta - \dot{\theta}^2 \sin \theta) + F \quad (6.2.1)$$

$$(I + me^2)\ddot{\theta} = -me\ddot{z} \cos \theta + N \quad (6.2.2)$$

Our novel approach to this problem is to use gain-scheduling as follows.

Letting $x = [x_1, x_2, x_3, x_4]^T = [\theta, z, \dot{\theta}, \dot{z}]^T$, $u = [u_1, u_2] = [F, N]$ and $p = [p_1, p_2] = [\cos x_1, x_3 \sin x_1]$, the equations of motion can be reformulated as

$$E(p_1)\dot{x} = A(p_2)x + Bu$$

or

$$\begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & (I + me^2) & mep_1 \\ 0 & 0 & mep_1 & (M + m) \end{pmatrix} \begin{pmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \\ 0 & -k & mep_2 & 0 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ 0 & 0 \\ 0 & 1 \\ 1 & 0 \end{pmatrix} \begin{pmatrix} u_1 \\ u_2 \end{pmatrix} \quad (6.2.3)$$

Description	Parameter	Value	Units
Cart mass	M	1.3608	kg
Arm mass	m	0.096	kg
Arm eccentricity	e	0.0592	m
Arm inertia	I	$2.175e^{-4}$	kgm^2
Spring stiffness	k	186.3	N/m
Coupling parameter	ε	0.200	—

Table 6.1: Parameter Values

A laboratory-scale version of the nonlinear benchmark problem has the parameters for a nominal configuration as shown in Table 6.1.

The σ -plot of the mapping $d \rightarrow [\theta \ z]^T$ when $(\theta = \frac{\pi}{4}, \dot{\theta} = 0)$ is shown in Figure 6.2.

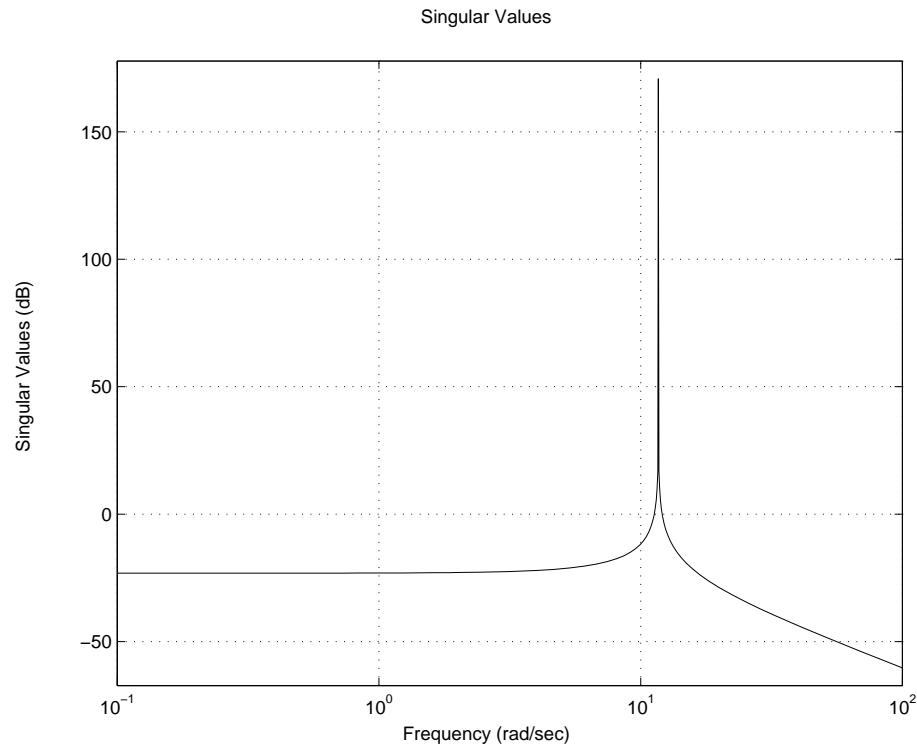
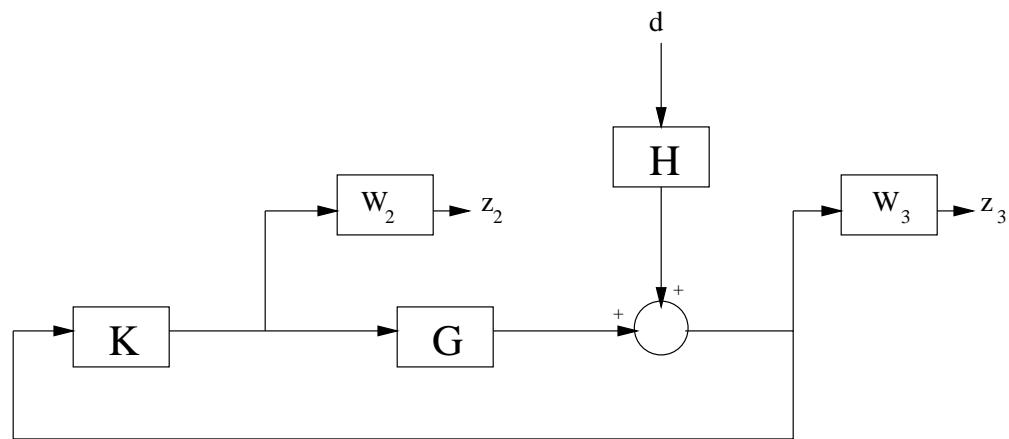
6.3 \mathcal{H}_∞ Controller Design on the RTAC Model

In this section we will start by designing an \mathcal{H}_∞ Controller for the RTAC model at a fixed angle θ . The plant 6.2.3 is linearized at $(\theta = \frac{\pi}{4}, \dot{\theta} = 0)$. The plant is set up as shown in Figure 6.3 and augmented using suitable weighting functions. The error signals are selected to be $z_2 = W_2 u$ and $z_3 = [w_3 \theta \ w_3 z]^T$. Both θ and z are used for feedback and the weighting functions used are

$$W_2^{-1} = 0.2; \quad W_3^{-1} = \begin{bmatrix} 10 \frac{(s+0.1)}{(s+10)} & 0 \\ 0 & 10 \frac{(s+0.1)}{(s+10)} \end{bmatrix}$$

The LFT description of the augmented plant is shown in Figure 6.4. As a first try with $\theta = \frac{\pi}{4}$, an H_∞ controller is obtained with a value of $\gamma = 0.735$

The sensitivity of the closed loop-system, i.e., the map $d \rightarrow [\theta \ z]^T, S = (I - GK)^{-1} H$ is then plotted together with the weighting function $W_3^{-1}(s)$ to ensure that the specifications are met. Clearly, this is the case from Figure 6.5.

Figure 6.2: σ -plot of the mapping $d \rightarrow [\theta \ z]^T$ Figure 6.3: System Interconnection for \mathcal{H}_∞ Design

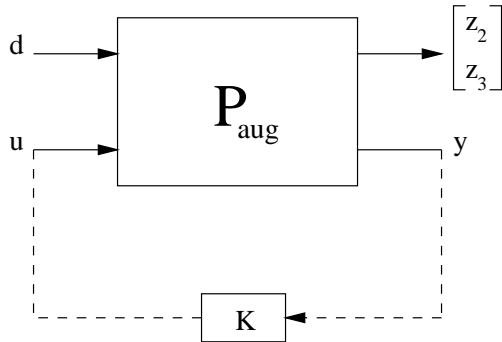
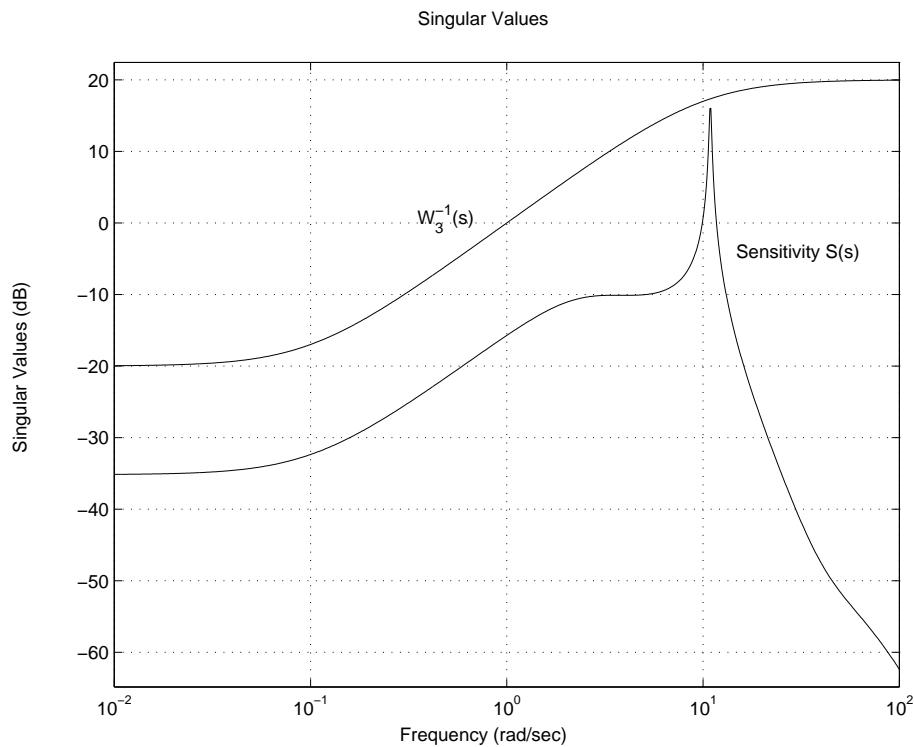


Figure 6.4: LFT Description for Augmented Plant

Figure 6.5: Sensitivity S and W_3^{-1}

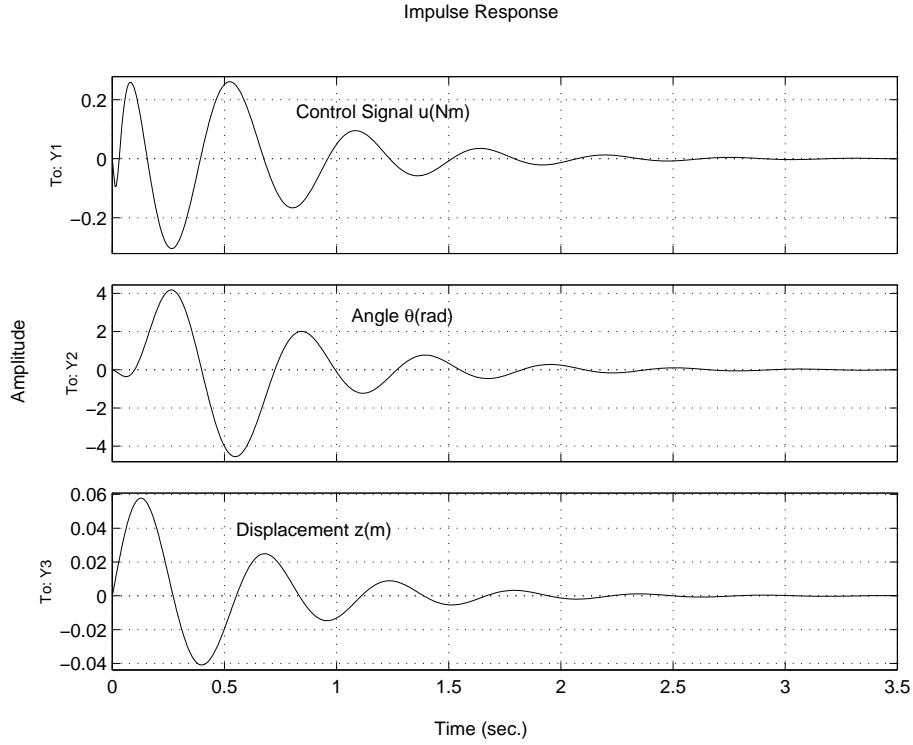


Figure 6.6: Impulse Response of the Closed-Loop System

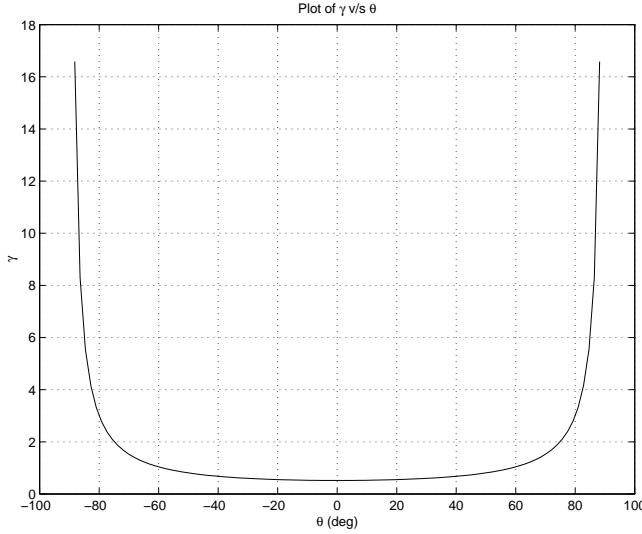
And the impulse response of the closed loop system is shown in Figure 6.6. The closed-loop system has an excellent settling time of about 2 seconds with reasonable control signal.

The H_∞ design is then repeated at numerous values of θ and the values of γ obtained are then plotted against θ as shown in Figure 6.7.

6.4 Gain-Scheduled Control

The aim here is to design a controller that satisfies certain criteria:

- The closed loop system is stable(locally and globally).
- The closed loop system exhibits good settling behaviour for a class of initial conditions.

Figure 6.7: Plot of γ v/s θ

- The closed loop system exhibits good disturbance rejection compared to the uncontrolled oscillator for a class of disturbance signals.
- The control effort should be reasonable.

6.4.1 LFT approach to Controller Synthesis

Using the LFT approach as explained previously in Chapter 4 based on [1, 2], the same problem is set up as shown in Figure 6.8 for synthesis and the block diagram of the augmented plant shown in Figure 6.9.

Weighting functions W_3 and W_n are used for the synthesis. W_n is added to make the number of exogenous input m_1 equal to the number of error signals z_1 as that is a prerequisite for the synthesis method to work with minimum manipulations of the similarity scaling matrices. The weighting functions used were

$$W_3^{-1} = \frac{8(s + 0.1)}{s + 10}; \quad W_n^{-1} = 1000$$

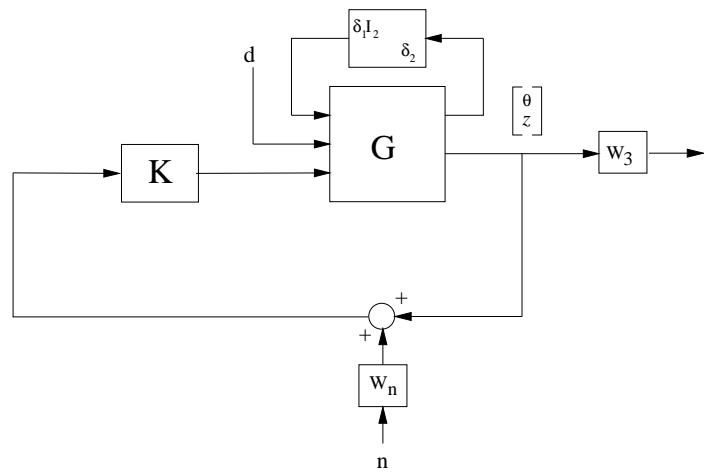


Figure 6.8: Gain-Scheduled Control of RTAC

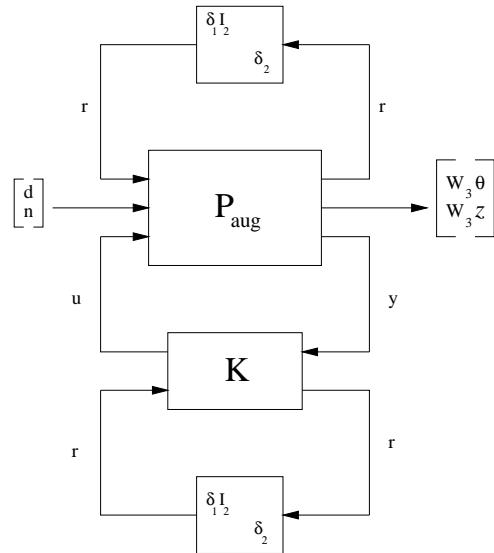


Figure 6.9: LFT Block diagram of the Augmented Plant

Parameter	Bound
p_1	$0.1 \leftrightarrow 1$
$ p_2 $	20

Table 6.2: Parameter bounds for controller design

Table 6.2 shows the parameter range used for the synthesis of the controller. The parameter variation of p_1 is restricted to be able to achieve a controller. This is due to the fact that when $\theta = 90$ deg and $\dot{\theta} = 0$, the system is uncontrollable. This means that it is assumed that $(-84.3 < \theta < 84.3)$ and it is ensured those limited range of parameter have to be ensured during simulation to ensure that the system remains stable. Appropriate weighting functions have been chosen to take care of this reduced range of parameter variation for p_1 . With these limitations in place, we designed a controller with 6 states and achieved a γ of 2.9.

First of all, we look at the new mapping $d \rightarrow [\theta \ z]^T$ in Figure 6.10. It is observed that the huge peak at $\omega \approx 12$ rad/sec has been pushed down drastically, helping the system to be less sensitive to disturbance when the controller is implemented. The weighting function W_3^{-1} can also be seen and looking at it, one might think that the controller could have been made to work more by selecting other weights. We have to bear in mind that these singular values are at fixed values of θ and $\dot{\theta}$ and even though, the specifications seem to be met here, there is no guarantee that they are met with the actual system at other operating conditions. The γ value of 2.9 achieved is proof of that.

The next stage is then controller validation. This is done through nonlinear simulations for various input disturbances. First of all, we look at the effect of a sinusoidal disturbance $d = \sin 2\pi t$. The system response is shown in Figure 6.11.

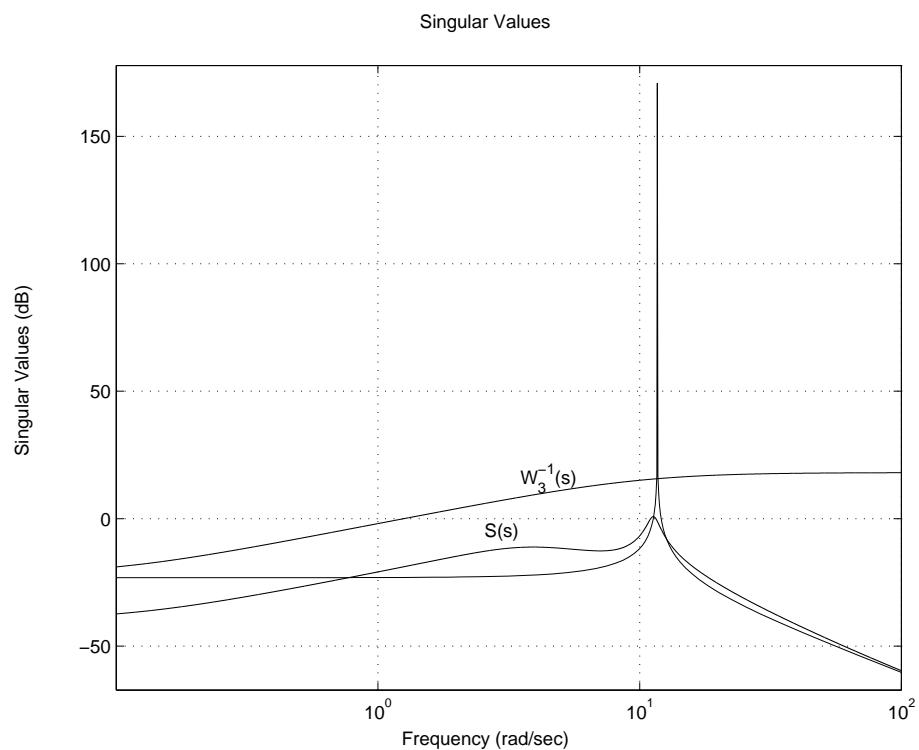


Figure 6.10: Open-loop mapping, closed-loop mapping and W_3^{-1}

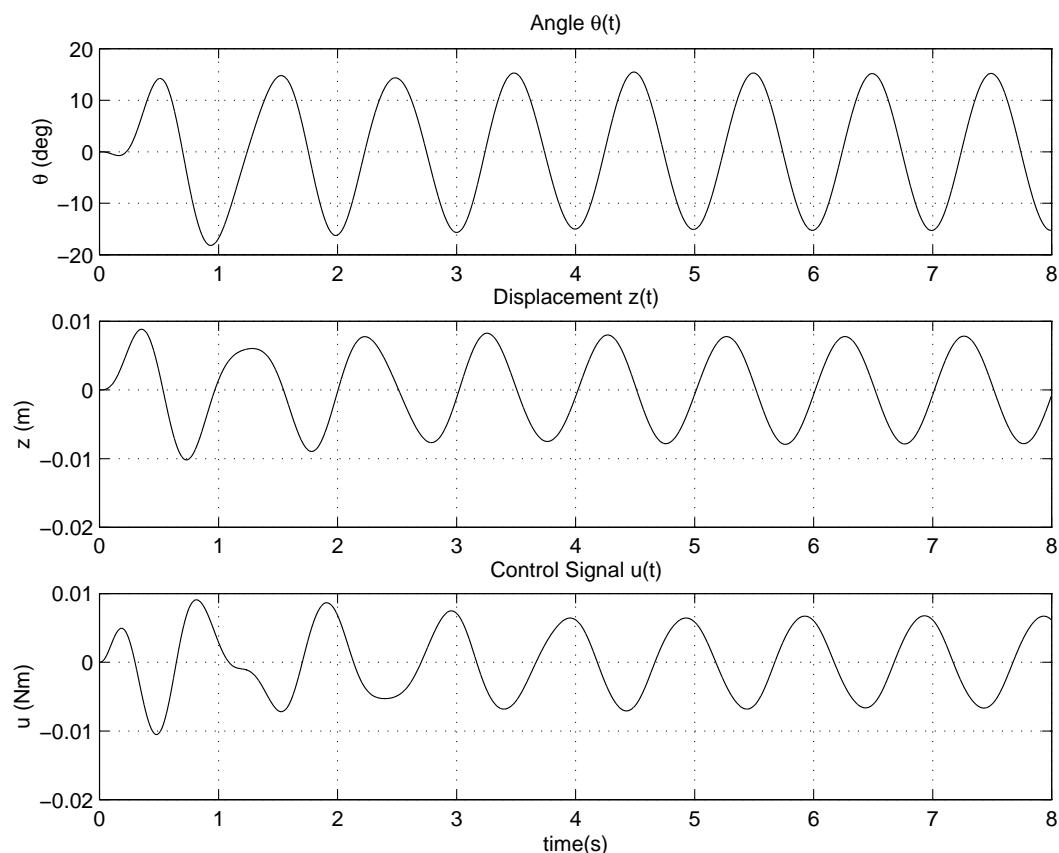


Figure 6.11: Nonlinear simulation to a disturbance $d = \sin 2\pi t$

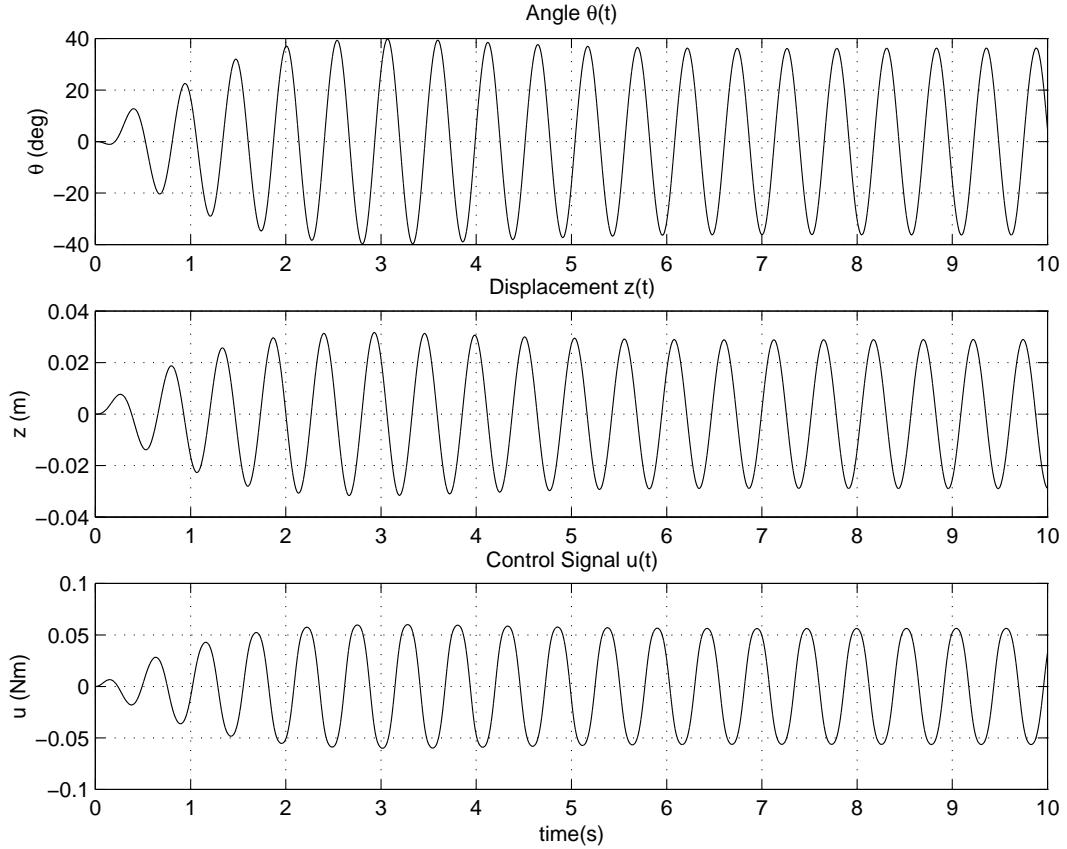


Figure 6.12: Nonlinear simulation to a disturbance $d = \sin 12t$

First of all, we can see that θ is very well-bounded between 15 deg and -15 deg and at no time did we approach a region which was outside the parametric range assumed when designing the controller. Similarly z is very small moving to and fro with an amplitude of 1 cm.

Next, we are going to disturb the system with a sinusoidal input but this time at the resonant frequency of the system which can be gathered to be approximately 12 rad/sec from Figure 6.2. We therefore use $d = \sin 12t$ to look at the response when the system is most sensitive to disturbance. The nonlinear response is shown in Figure 6.12. As expected, the response has deteriorated with θ varying between 35

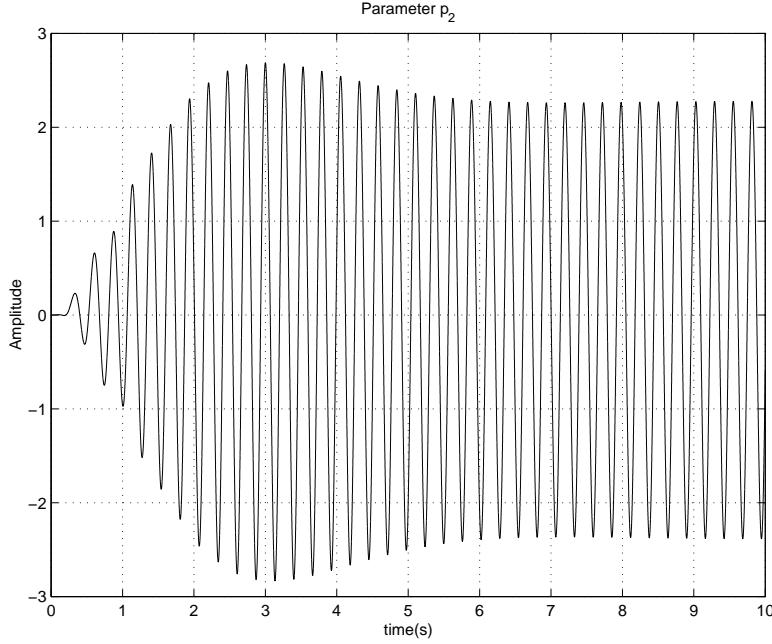


Figure 6.13: Variation of parameter p_2 to a disturbance $d = \sin 12t$

deg and -35 deg. z is now oscillating with a maximum amplitude of about 3 cm and the control energy needed to control the system now has multiplied more than 6 times from what was required in the previous simulation. However given all that, we have to realize that we are actually considering the worst case sinusoidal response that the system can offer and that still none of the parameters have been violated. For proof, since θ is still well within the range where the design was performed, $p_1 = \cos \theta$ is clearly within 0.1 and 1. For parameter $p_2 = \dot{\theta} \sin \theta$, we look at its variation in Figure 6.13 and it is well within the parameter variation allowed for p_2 .

And finally, we look at the response to an impulse disturbance in Figure 6.14. The system comes to rest in about 4.5 seconds which is an excellent settling time [12, 20, 57]. The angle θ peaks at about 55 degrees which is again within bounds and the system undergoes a maximum displacement of about 3.8 cm. We look at the variation

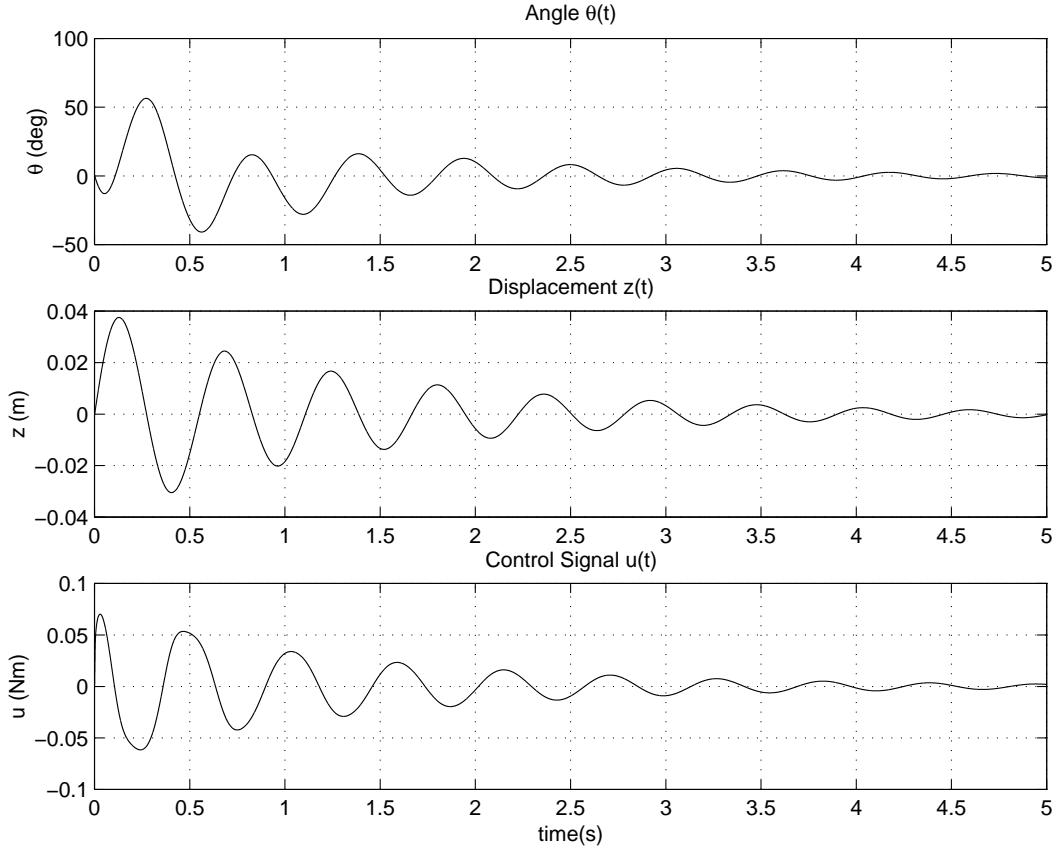


Figure 6.14: Nonlinear simulation to an impulse disturbance

parameter p_2 in Figure 6.15 and we confirm that the parameter bounds have not been violated.

The RTAC has been described and modelled appropriately for nonlinear design. With the parameter p_1 restricted, suitable weights were chosen to ensure that the parameter bounds assumed are not violated while at the same time guaranteeing high standards of performance. We designed a gain-scheduled controller using the LFT approach on the system and assessed its performance through nonlinear simulation using different disturbance signals. Conclusive results were obtained using this approach as can be seen from the simulation results.

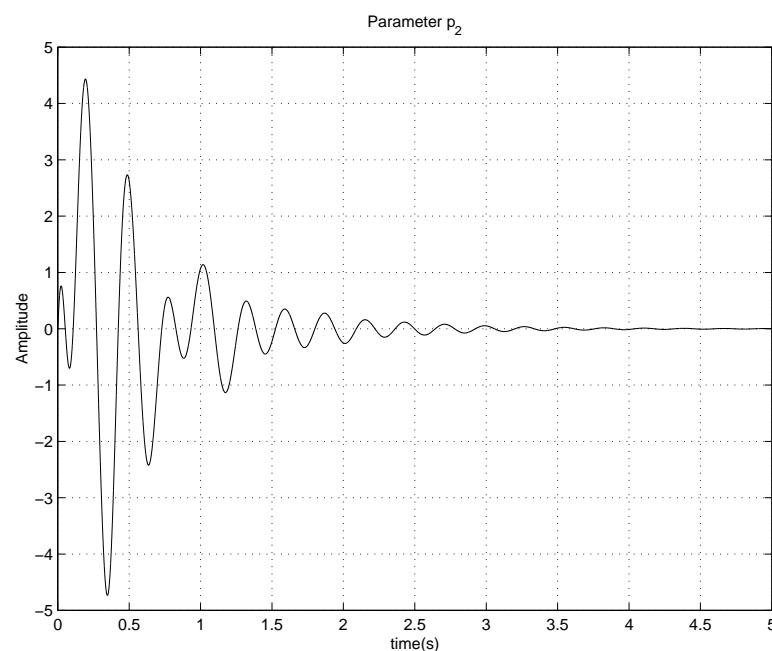


Figure 6.15: Variation of parameter p_2 to an impulse disturbance

Chapter 7

Missile Model

7.1 Introduction

In this chapter, we will apply the LFT design method of [1, 2] as discussed in Chapter 4 to a nonlinear missile model. For reference, this model has been looked at in [23] where the “polytopic” approach was used to synthesize a gain scheduled controller. The LFT approach will be a first on this model.

7.2 Dynamics of the Model

The missile dynamics are strongly dependent on the angle of attack α , the air speed V and the altitude H . These three variables undergo large variations during operation. They completely define the flight conditions(operating point) of the missile and they are assumed to be measured in real time. The LPV model is developed for this problem on the basis of linearization of the missile equations around its flight conditions. To simplify the design procedure while retaining the main difficulties of the problem, it is assumed that the pitch, yaw and roll axes are decoupled. A simple

model of the linearized dynamics of the missile is the parameter-dependent model

$$\begin{aligned}\begin{pmatrix} \dot{\alpha} \\ \dot{q} \end{pmatrix} &= \begin{pmatrix} -Z_\alpha & 1 \\ -M_\alpha & 0 \end{pmatrix} \begin{pmatrix} \alpha \\ q \end{pmatrix} + \begin{pmatrix} 0 \\ 1 \end{pmatrix} \delta_m \\ \begin{pmatrix} a_{zv} \\ q \end{pmatrix} &= \begin{pmatrix} -1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} \alpha \\ q \end{pmatrix}\end{aligned}$$

where a_{zv} is the normalized vertical acceleration, q is the pitch rate, δ_m the fin deflection, and Z_α, M_α are aerodynamical coefficients depending on α, V , and H . These two coefficients are functions of the flight conditions and are therefore available in real time. Though the parameter dependence of the original plant has been simplified, the control of the missile dynamics remains a hard task. Indeed, the parameters Z_α and M_α abruptly change as functions of the flight conditions, and range over a large parameter domain where the stability properties of the missile vary greatly. Analyzed as an LTI plant, the characteristic polynomial of the plant is $s^2 + Z_\alpha s + M_\alpha$. It follows that the plant is LTI-unstable whenever M_α is negative. The parameter Z_α has a less dramatic effect, and influences the damping.

As V, H and α vary in

$$V \in [0.5, 4] \text{ Mach}, \quad H \in [0, 18000] \text{ m}, \quad \alpha \in [0, 40] \text{ deg}$$

during operation, the coefficients Z_α and M_α range in

$$Z_\alpha \in [0.5, 4], \quad M_\alpha \in [0, 106].$$

7.3 Gain-Scheduled Control

7.3.1 Problem Specification and Synthesis

Our goal is to control the vertical acceleration a_{zv} over this operating range. The stringent performance specifications (settling time < 0.5 s) and the bandwidth limitation

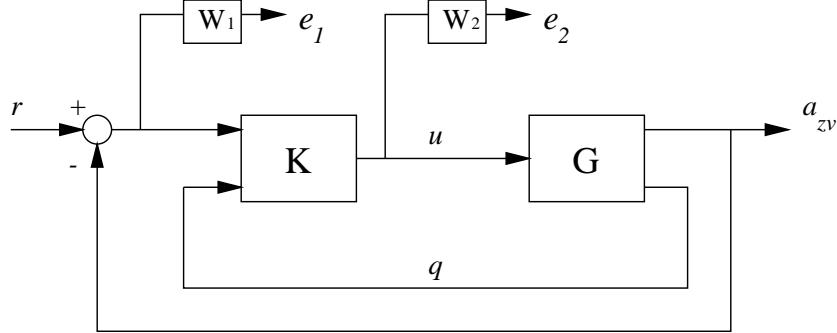


Figure 7.1: Missile Control Autopilot

imposed by unmodeled high frequency dynamics make gain scheduling desirable. The control structure is as shown in Figure 7.1. To enforce the performance and robustness requirements, we used the loop-shaping criterion

$$\left\| \begin{bmatrix} W_1 S \\ W_2 K S \end{bmatrix} \right\|_\infty < 1. \quad (7.3.1)$$

The shaping filters chosen are as follows

$$W_1(s) = \frac{2.01}{s + 0.201} \quad (7.3.2)$$

$$W_2(s) = \frac{9.678s^3 + 0.029s^2}{s^3 + 1.206e4s^2 + 1.136e7s + 1.066e10} \quad (7.3.3)$$

The LMIs (4.3.9–4.3.12) are solved for R , S , L_3 , and J_3 and a 6th order controller is computed yielding a performance level of $\gamma = 0.33$.

7.3.2 Assessment of LPV Controller

The resulting LPV controller is now tested through time-domain simulations. The step response of the gain-scheduled system is simulated along the two parameter trajectories shown in Figure 7.2. Trajectory 1 is a smooth spiral trajectory

$$Z_\alpha(t) = 2.25 + 1.70 e^{-4t} \cos(20t)$$

$$M_\alpha(t) = 50 + 49 e^{-4t} \sin(20t)$$

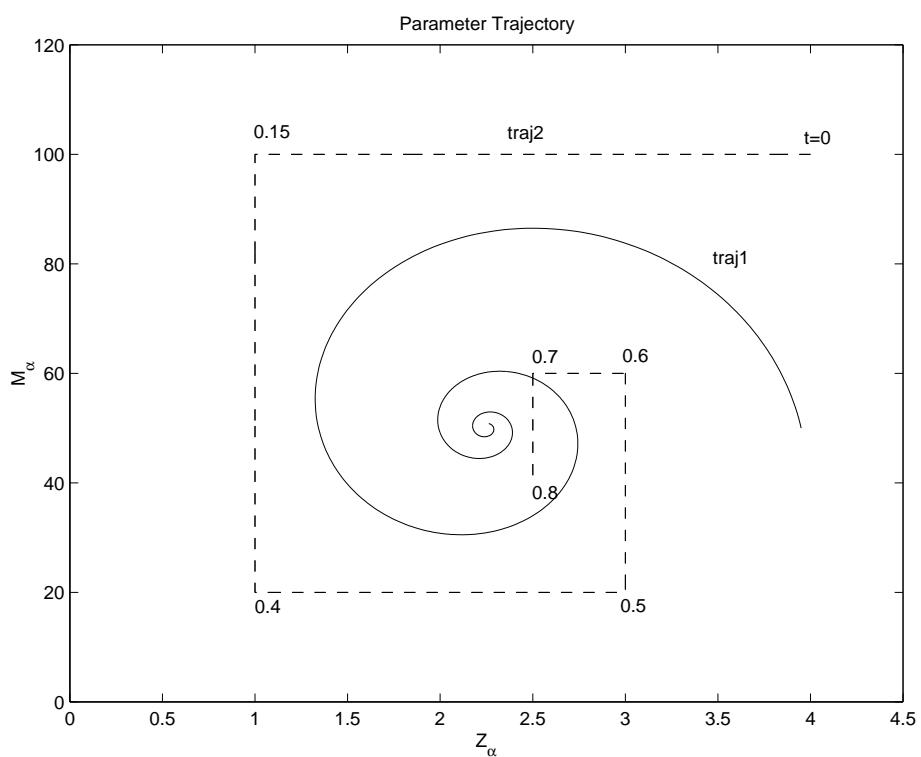


Figure 7.2: Parameter trajectories 1 (solid), 2 (dashed)

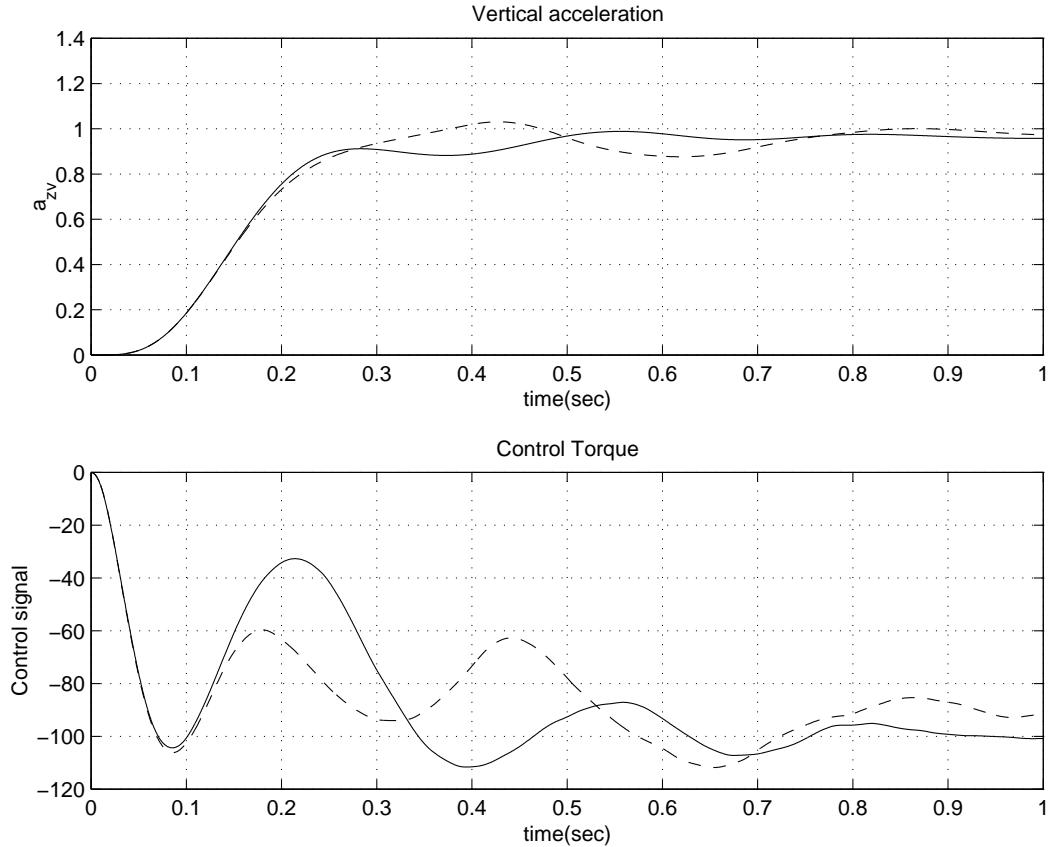


Figure 7.3: Nonlinear simulation of missile over trajectories 1 (solid), 2 (dashed)

In contrast, trajectory 2 is non-smooth and is intended to test the reaction of the LPV control law in the face of abrupt parameter changes.

The corresponding nonlinear simulations are presented in Figure 7.3. It can be seen that the gain-scheduled controller performs well. The settling time specification is clearly met for trajectory 1. A slight degradation occurs for trajectory 2. This degradation is due to the corner points of the parameter trajectory, where there is an infinite derivative of the parameter. However, from a practical point of view, the response is still satisfactory, since infinite derivatives are not realistic. Summing up then, the controller behaved very well as predicted by the theory and managed to meet

tight specifications with little apparent conservatism. The potential and performance of the approach has been demonstrated on a realistic missile pilot model. Adequate filter selections allow for tight robustness and performance and more importantly, such specifications have been maintained, even for rapidly changing parameters.

Chapter 8

Conclusions and Further Work

8.1 Conclusions

This thesis has done a survey of present control techniques and applied them to three case studies. The control techniques have been covered in considerable detail, especially with the gain-scheduled approaches where explicit equations were given towards the numerical computation of controllers using LMIs. The two approaches to gain-scheduling that we have studied, LFT and polytopic, even though closely related are based on quite different approaches and each has its own set of pros and cons. For instance, the LFT approach is able to cope with a lot more uncertainties, e.g., if the number of uncertainties entering the plant is n , the LFT approach deals with $2n$ uncertainties. This is unfortunately not the case with the polytopic approach where the problem blows up and the number of solutions and consequently LMIs to be solved for becomes 2^n . That is quite a drawback of the polytopic approach. The LFT approach on the other hand tends to be conservative. The uncertainties are treated as complex and no bound is present on them. While the polytopic approach allows for rate of parameter variation and cater for real parametric uncertainty, the LFT approach is limited in that way.

The approach that one decides to go for will vary from problem to problem but at first sight, the LFT approach looks more appealing. The main reason for that being that it involves less computational burden which can ultimately decide the feasibility of the problem. For problems that can be modelled to a high degree of accuracy, with known range of parameter variation and with known rate of range of parameter variation, i.e., with low number of uncertainties, the polytopic approach might be better suited. But for complex problems with huge number of uncertainties and where parameter varies arbitrarily fast, we would recommend the LFT approach.

Let us now look at a summary of the results obtained in each case-study. The first model, a two-link manipulator model has been modelled and set up appropriately depending on the control method to be applied. In the first instance, we looked at applying the Mixed-sensitivity approach, followed by the μ -synthesis method. The mixed-sensitivity approach being restricted to unstructured uncertainty, we could not guarantee stability for the whole range of parameter variation and performance was poor at best. With the μ controller, the controller obtained was of a very high order (36 states). The μ -synthesis approach is able to cope with structured and unstructured uncertainties as long as they are complex. That introduced a fair degree of conservatism in the design since the two-link manipulator had repeated real uncertainties. This conservatism of course resulting in average performance. In each of these two cases, the model had been linearized first and then the appropriate techniques applied. We then moved on and looked at an approach that took care of the complete nonlinear two-link manipulator model. The nonlinearities in the model were removed through appropriate substitutions and we ended up with what we call a “quasi-LPV” model. A gain-scheduled controller was then designed on this controller

using the polytopic approach. An LPV controller with 12 states and a $\gamma = 1.87$ was achieved. The system was found to be stable with the controller and performance was superior to that achieved with the previous controllers.

The second model (RTAC) was a more challenging model which has been studied extensively [12, 20, 57]. However our approach was a first on the model and we successfully designed a gain-scheduled controller using the LFT approach. Nonlinear simulations performed on the model showed that the results were superior to those obtained previously. It has to be pointed out though that the controller was designed for a limited range of parameter variation of the system. Having said that, the system never went outside those parameter bounds when simulation was done.

Finally, we looked at a missile model [3] and again a gain-scheduled controller was successfully designed using the LFT approach. Extensive simulation was done on the model and it was found to be extremely robust even when subjected to abrupt parameter changes. In [3], the same model is simulated with a gain-scheduled controller designed using the polytopic approach and our simulation compares favourably to theirs especially when the parameter changes are quick and abrupt.

8.2 Future Work

Throughout this research, numerous difficulties have been encountered and these can be presented as avenues for future work:

- While doing μ -synthesis on the two-link manipulator, it was realised that the μ -Toolbox [6] could not cope well with purely real or mixed uncertainty as far as synthesis is concerned. The final results of course tended to be conservative as the parameters were assumed to be complex.

- After the polytopic controller was designed on the two-link manipulator model, nonlinear simulation was the next step. But it was found that nonlinear simulation took an extremely long time. This resulted in the inability to perform “extensive” simulation with the LPV controller on the robot model. The iteration step size had to be reduced considerably compared to the step size used for μ -analysis to ensure that the states actually converge. The reason for this resistance to convergence is that the controller parameters vary at the same time as the states. This is fortunately not the case with the gain-scheduled LFT controller.
- The LFT approach being able to only deal with complex uncertainties introduces a fair bit of conservatism into the controller design. Future work could look into incorporating real parametric uncertainties to it.
- The polytopic approach even though less conservative, allowing parametric uncertainty and catering for rates of parameter variation, tends to blow up in size as the number of parametric uncertainties increase. The problem becomes extremely complex and more often than not tends to be infeasible.

Part III

Appendices

Appendix A

MATLAB Source File

```
function [K] = gslmi(Paug,r,m1,m2,p1,p2)
% Usage [K] = gslmi(Paug,r,m1,m2,p1,p2);
% Find an lft gain-scheduled controller for an augmented plant Paug using LMI.
% Algorithm based on paper "Convex Characterization of Gain-Scheduled
% H-inf Controllers by Pierre Apkarian and Pascal Gahinet.
% Output
%   K : the gain-sched controller
% Inputs
%   P : augmented plant
%   r : no. of uncertainties
%   m1 : no. of disturbances
%   m2 : no. of control signals
%   p1 : no. of error signals
%   p2 : no. of measurement signals

[ap,bp,cp,dp] = ltiss(Paug);
A = ap; [n,n] = size(A);
Btheta = bp(:,1:r); B1 = bp(:,r+1:r+m1); B2 = bp(:,r+m1+1:r+m1+m2);
Ctheta = cp(1:r,:); C1 = cp(r+1:r+p1,:); C2 = cp(r+p1+1:r+p1+p2,:);
Dtheta = dp(1:r,1:r); Dtheta1 = dp(1:r,r+1:r+m1);
Dtheta2 = dp(1:r,r+m1+1:r+m1+m2); D1theta = dp(r+1:r+p1,1:r);
D11 = dp(r+1:r+p1,r+1:r+m1); D12 = dp(r+1:r+p1,r+m1+1:r+m1+m2);
D2theta = dp(r+p1+1:r+p1+p2,1:r); D21 = dp(r+p1+1:r+p1+p2,r+1:r+m1);
D22 = dp(r+p1+1:r+p1+p2,r+m1+1:r+m1+m2);

% Null Space Nr and Ns
X = [B2' Dtheta2' D12' zeros(m2,r+m1)]; NR = null(X);
Y = [C2 D2theta D21 zeros(p2,r+p1)]; NS = null(Y);
```

```

% Define 1st set of LMIs setlmis([])
gamma = 2.9;
fprintf('gamma is set to %3.2f',gamma);
% R and S need only be symmetric
R = lmivar(1,[n 1]); S = lmivar(1,[n 1]);
% L3 & J3 are symmetric and diagonal matrices
xx=[1 0]; i=1;
while i<r
xx=[xx;1 0]; i=i+1;
end
L3 = lmivar(1,xx); J3 = lmivar(1,xx);

% 1st LMI
lmiterm([1 0 0 0],NR); % outer factor NR
lmiterm([1 1 1 R],A,1,'s'); % A*R + R*A'
lmiterm([1 2 1 R],Ctheta,1); % Ctheta*R
lmiterm([1 2 2 J3],-gamma,1); % -gamma*J3
lmiterm([1 3 1 R],C1,1); % C1*R
lmiterm([1 3 2 0],0); % 0
lmiterm([1 3 3 0],-gamma); % -gamma*I
lmiterm([1 4 1 J3],1,Btheta'); % J3*Btheta'
lmiterm([1 4 2 J3],1,Dthetatheta'); % J3*Dthetatheta'
lmiterm([1 4 3 J3],1,D1theta'); % J3*D1theta'
lmiterm([1 4 4 J3],-gamma,1); % -gamma*J3
lmiterm([1 5 1 0],B1'); % B1'
lmiterm([1 5 2 0],Dtheta1'); % Dtheta1'
lmiterm([1 5 3 0],D11'); % D11'
lmiterm([1 5 4 0],0); % 0
lmiterm([1 5 5 0],-gamma); % -gamma*I

% 2nd LMI
lmiterm([2 0 0 0],NS); % outer factor NS
lmiterm([2 1 1 S],A',1,'s'); % A'*S + S*A
lmiterm([2 2 1 S],Btheta',1); % Btheta'*S
lmiterm([2 2 2 L3],-gamma,1); % -gamma*L3
lmiterm([2 3 1 S],B1',1); % B1'*S
lmiterm([2 3 2 0],0); % 0
lmiterm([2 3 3 0],-gamma); % -gamma*I
lmiterm([2 4 1 L3],1,Ctheta); % L3*Ctheta
lmiterm([2 4 2 L3],1,Dthetatheta); % L3*Dthetatheta
lmiterm([2 4 3 L3],1,Dtheta1); % L3*Dtheta1
lmiterm([2 4 4 L3],-gamma,1); % -gamma*L3
lmiterm([2 5 1 0],C1); % C1

```

```

lmiterm([2 5 2 0],D1theta); %D1theta
lmiterm([2 5 3 0],D11); %D11
lmiterm([2 5 4 0],0); %0
lmiterm([2 5 5 0],-gamma); %-gamma*I

% 3rd LMI
lmiterm([-3 1 1 R],1,1); % R
lmiterm([-3 2 1 0],1); % I
lmiterm([-3 2 2 S],1,1); % S

% 4th LMI
lmiterm([-4 1 1 L3],1,1); %L3
lmiterm([-4 2 1 0],1); %I
lmiterm([-4 2 2 J3],1,1); %J3

LMISYS = getlmis;

[tmin, xfeas] =feasp(LMISYS,[0 100 1e4 0 0],-5);
R = dec2mat(LMISYS, xfeas, R);
S = dec2mat(LMISYS, xfeas, S);
L3 = dec2mat(LMISYS,xfeas, L3);
J3 = dec2mat(LMISYS,xfeas, J3);

% MN' = I - R*S
[u,s,v] = svd(eye(n) - R*S);
M = u*s; [n,k] = size(M); N = v;

% Finding Xcl
Xcl = [S eye(n,k);N' zeros(k)]*inv([eye(n,k) R;zeros(k) M']);

% Finding L
mm = L3 - inv(J3);
[u,s,v] = svd(mm);
% u = v for symm and pos.def matrix
L1 = inv(s); L2 = u'; L = [L1 L2;L2' L3];
LL = [L zeros(2*r,p1);zeros(p1,2*r) eye(p1)]; JJ = inv(LL);

% Shorthands
Ao = [A zeros(n,k);zeros(k,n) zeros(k)];
Bo = [zeros(n,r) Btheta B1;zeros(k,m1+2*r)];
BB = [zeros(n,k) B2 zeros(n,r);eye(k) zeros(k,m2+r)];
Co = [zeros(r,n+k);Ctheta zeros(r,k);C1 zeros(p1,k)];
DD11 =[zeros(r,2*r+m1);zeros(r) Dthetatheta Dtheta1;zeros(p1,r) D1theta D11];

```

```

DD12 = [zeros(r,k+m2) eye(r);zeros(r,k) Dtheta2 zeros(r);zeros(p1,k) D12 zeros(p1,r)];
CC = [zeros(k,n) eye(k);C2 zeros(p2,k);zeros(r,n+k)];
DD21 =[zeros(k,2*r+m1);zeros(p2,r) D2theta D21;eye(r) zeros(r,r+m1)];
%Phi
Phi = [Ao'*Xcl+Xcl*Ao Xcl*Bo Co';
Bo'*Xcl -gamma*LL DD11';
Co DD11 -gamma*JJ];

%PP
PP = [BB' zeros(r+m2+k,m1+2*r) DD12'];

%Q
Q = [CC DD21 zeros(r+p2+k,p1+2*r)];

%XX and YY
XX = [Xcl zeros(n+k,p1+m1+4*r);zeros(p1+m1+4*r,n+k) eye(p1+m1+4*r)]*PP';

% LMI for controller theta
K = basiclmi(Phi,XX',Q);
Ak = K(1:n,1:n); Bk = K(1:n,n+1:n+p2+r);
Ck = K(n+1:n+m2+r,1:n);Dk = K(n+1:n+m2+r,n+1:n+p2+r);
K = ltisys(Ak,Bk,Ck,Dk);

```

Appendix B

Non-Singularity of the Manipulator

$$\begin{aligned}
M &= H^{-1} \\
&= \begin{bmatrix} h_{11} & h_{12} \\ h_{21} & h_{22} \end{bmatrix}^{-1} \\
&= \frac{1}{\Delta} \begin{bmatrix} h_{22} & -h_{12} \\ -h_{21} & h_{11} \end{bmatrix} \\
h_{11} &= a_1 + 2a_3 \cos q_2 + 2a_4 \sin q_2 \\
h_{12} &= h_{21} = a_2 + a_3 \cos q_2 + a_4 \sin q_2 \\
h_{22} &= a_2 \\
\Delta &= h_{11} h_{22} - h_{12} h_{21} \\
&= (a_1 + 2a_3 \cos q_2 + 2a_4 \sin q_2)a_2 \\
&\quad + (a_2 + a_3 \cos q_2 + a_4 \sin q_2)(a_2 + a_3 \cos q_2 + a_4 \sin q_2) \\
&= a_1 a_2 - a_2^2 - a_3^2 \cos^2 q_2 - 2a_3 a_4 \sin q_2 \cos q_2 - a_4^2 \sin^2 q_2 \\
&= a_1 a_2 - a_2^2 - (a_3 \cos q_2 + a_4 \sin q_2)^2
\end{aligned} \tag{B.0.1}$$

From equation (B.0.1), it is clear that the determinant is a sinusoidally varying function and our aim is to make sure that the determinant never becomes zero, ensuring that the matrix H remain non-singular as q_2 varies from 0 to 2π .

For our model, the variation of the determinant with q_2 is shown in Fig B.1. The determinant remains always positive and the graph does not cross the x -axis for any value of q_2 . This is ensured with proper selection of a_1, a_2, a_3 and a_4 which in turn depends on the masses, lengths and inertias of the manipulator arms.

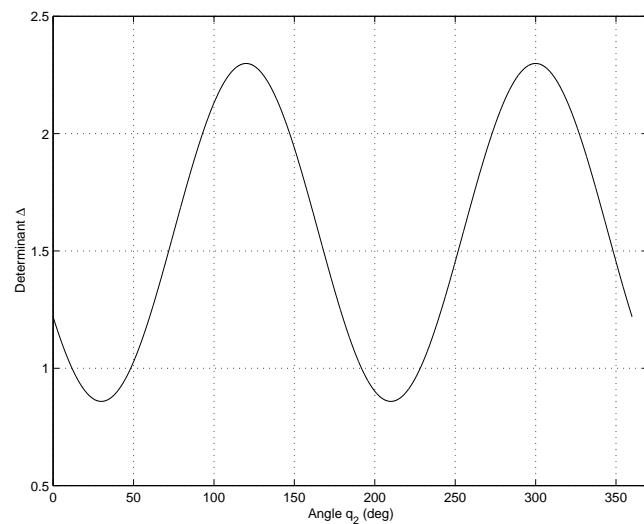


Figure B.1: Variation of Determinant with angle q_2

Bibliography

- [1] P. Apkarian and P. Gahinet, *A convex characterization of gain-scheduled \mathcal{H}_∞ controllers*, IEEE Trans. Autom. Control **40** (1995), no. 5, 853–863.
- [2] ———, *Erratum to “A convex characterization of gain-scheduled \mathcal{H}_∞ controllers”*, IEEE Trans. Autom. Control **40** (1995), no. 9, 1681.
- [3] P. Apkarian, P. Gahinet, and G. Becker, *Self-scheduled \mathcal{H}_∞ control of linear parameter-varying systems : A design example*, Automatica **31** (1995), no. 9, 1251–1261.
- [4] M. Athans, *Editorial on the LQG problem*, IEEE Trans. Autom. Control **16** (1971), no. 6, 528.
- [5] M. Athans, D. Castanon, K.P. Dunn, and C.S. Greene, *The stochastic control of the F-8C aircraft using a multiple model adaptive control method*, IEEE Trans. Autom. Control **22** (1977), no. 5, 768–791.
- [6] G. Balas, J.C. Doyle, K. Glover, A. Packard, and R. Smith, *μ -analysis and synthesis toolbox*, The Math Works, 1995.
- [7] G.J. Balas, I. Fialho, A. Packard, J. Renfrow, and C. Mullaney, *On the design of LPV Controllers for the F-14 aircraft lateral-directional axis during powered approach*, Proc. American Control Conf., June 1997, pp. 123–127.
- [8] J.A. Ball, J.W. Helton, and M. Verma, *A factorization principle for stabilization of linear control systems*, Int J. Robust and Nonlinear Control **1** (1999), 229–294.

- [9] G. Becker and A. Packard, *Robust performance of linear parametrically varying systems using parametrically-dependent linear feedback*, Systems and Control Letters (1994), no. 23, 205–215.
- [10] S. Bennani, D.M.C. Willemsen, and C.W. Scherer, *Robust control of linear parametrically varying systems with bounded rates*, Journal of Guidance, Control and Dynamics **21** (1998), no. 6, 916–922.
- [11] S. Boyd, L. El-Ghaoui, E. Feron, and V. Balakrishnan, *Linear matrix inequalities in system and control theory*, SIAM, 1994.
- [12] R.T. Bupp, D.S. Bernstein, and V.T. Coppola, *A benchmark problem for nonlinear control design*, Int J. Robust and Nonlinear Control (1998).
- [13] J.B. Burl, *Linear optimal control*, Addison-Wesley Publishing Company, Inc, 1999.
- [14] G. Chen and T. Sugie, *An upper bound of μ based on the parameter dependent multipliers*, Proc. American Control Conf., June 1997, pp. 2604–2608.
- [15] J.C. Doyle, *Guaranteed margins for LQG regulators*, IEEE Trans. Autom. Control **23** (1978), no. 4, 756–757.
- [16] J.C. Doyle, K. Glover, and P.P. Khargonekar, *State-space solutions to standard \mathcal{H}_2 and \mathcal{H}_∞ control problems*, IEEE Trans. Autom. Control **34** (1989), 831–847.
- [17] J.C. Doyle and G. Stein, *Multivariable feedback design*, IEEE Trans. Autom. Control **26** (1981), 4–16.
- [18] J.C. Doyle, A. Packard, and K. Zhou, *Review of LFTs, LMIs, and μ* , Proc. IEEE Conf. on Decision and Control, December 1991, pp. 1227–1232.

- [19] J.C. Doyle, J. Wall, and G. Stein, *Performance and robustness analysis for structured uncertainty*, Proc. IEEE Conf. on Decision and Control, December 1982, p. 629.
- [20] S. Dussy and L. El-Ghaoui, *Measurement-scheduled control for the RTAC problem : An LMI approach*, Int J. Robust and Nonlinear Control **8** (1998), 377–400.
- [21] P. Gahinet, *Explicit controller formulas for LMI-based \mathcal{H}_∞ synthesis*, Automatica **32** (1996), no. 7, 1007–1014.
- [22] P. Gahinet and P. Apkarian, *A linear matrix inequality approach to \mathcal{H}_∞ control*, Int J. Robust and Nonlinear Control **4** (1994), 421–448.
- [23] P. Gahinet, A. Nemirovski, A.J. Laub, and M. Chilali, *LMI control toolbox*, The Mathworks, Inc., 1995.
- [24] W. Hahn, *Stability of motion*, Springer-Verlag Berlin, July 1967.
- [25] S. Hassen, F. Crusca, and H. Abachi, *Modelling systems for control studies - An overview*, ISCA 15th International Conference on Computers and Their Applications, March 2000.
- [26] R.A. Horn and C.R. Johnson, *Matrix analysis*, Cambridge University Press, 1985.
- [27] A. Isidori and A. Astolfi, *Disturbance attenuation and \mathcal{H}_∞ -control via measurement feedback in nonlinear systems*, IEEE Trans. Autom. Control **37** (1992), no. 9, 1283–1293.
- [28] E.A. Jonckheere and N. Ke, *Complex-analytic theory of the μ -function*, Proc. American Control Conf., June 1997, pp. 366–371.
- [29] H. Kajiwara, P. Apkarian, and P. Gahinet, *Wide-range stabilization of an arm-driven inverted pendulum using linear parameter-varying techniques*, American Institute of Aeronautics and Astronautics (1998), 1853–1865.

- [30] _____, *LPV techniques for control of an inverted pendulum*, IEEE Control Systems (1999).
- [31] I. Kaminer, A.M. Pascoal, P.P. Khargonekar, and E.E. Coleman, *A velocity algorithm for the implementation of gain-scheduled controllers*, Automatica **31** (1995), no. 8, 1185–1191.
- [32] H.K. Khalil, *Nonlinear systems*, Prentice Hall, 1996.
- [33] P.P. Khargonekar, I.R. Petersen, and K. Zhou, *Robust stabilization of uncertain linear systems: Quadratic stabilization and \mathcal{H}_∞ control theory*, IEEE Trans. Autom. Control **35** (1990), 356–361.
- [34] D.A. Lawrence and W.J. Rugh, *Gain scheduling dynamic linear controllers for a nonlinear plant*, Automatica **31** (1995), no. 3, 381–390.
- [35] L.H. Lee, *Reduced-order solutions to \mathcal{H}_∞ and LPV control problems involving partial-state feedback*, Proc. American Control Conf., June 1997, pp. 1762–1765.
- [36] L.H. Lee and M. Spillman, *A parameter-dependent performance criterion for linear parameter-varying systems*, Proc. IEEE Conf. on Decision and Control, December 1997, pp. 984–989.
- [37] W. Lu and J.C. Doyle, *Robustness analysis and synthesis for nonlinear uncertain systems*, IEEE Trans. Autom. Control **42** (1997), no. 12, 1654–1662.
- [38] J.M. Maciejowski, *Multivariable feedback design*, Addison-Wesley Publishing Company, Inc, 1989.
- [39] M.W. McConley, M.A. Dahleh, and E. Feron, *Polytopic control Lyapunov functions for robust stabilization of a class of nonlinear systems*, Proc. American Control Conf., June 1997, pp. 416–419.

- [40] T. McKelvey and A. Helmersson, *μ -Synthesis with matrix valued scalings - Algorithms and examples*, Proc. American Control Conf., June 1997, pp. 361–365.
- [41] P.J. McKerrow, *Introduction to robotics*, Addison-Wesley Publishing Company, Inc, 1991.
- [42] H.H. Niemann, J. Stoustrup, S. Toffner-Clausen, and P. Andersen, *μ -Synthesis for the coupled mass benchmark problem*, Proc. American Control Conf., June 1997, pp. 2611–2615.
- [43] A. Packard, *Gain scheduling via linear fractional transformations*, Systems and Control Letters (1994), no. 22, 79–92.
- [44] A. Packard and J.C. Doyle, *The complex structured singular value*, Automatica **29** (1993), no. 1, 71–109.
- [45] A. Packard, K. Zhou, P. Pandey, and G. Becker, *A collection of robust control problems leading to LMIs*, Proc. IEEE Conf. on Decision and Control, December 1991, pp. 1245–1250.
- [46] I.R. Petersen, V.A. Ugrinovskii, and A.V. Savkin, *Robust control design using \mathcal{H}_∞ methods*, Springer-Verlag, London, 2000.
- [47] H.H. Rosenbrock and P.D. Morran, *Good, bad or optimal?*, IEEE Trans. Autom. Control **16** (1971), no. 6, 522–529.
- [48] W.J. Rugh and J.S. Shamma, *Research on gain scheduling*, Automatica **36** (2000), 1401–1425.
- [49] M.G. Safonov and M. Athans, *Gain and phase margin for multiloop LQG regulators*, Proc. IEEE Conf. on Decision and Control, December 1976, pp. 361–368.
- [50] ———, *A multiloop generalization of the circle criterion for stability margin analysis*, IEEE Trans. Autom. Control **26** (1981), no. 2, 415–422.

- [51] M.G. Safonov and M.K.H. Fan, *Editorial*, Int J. Robust and Nonlinear Control **7** (1997), 97–103.
- [52] M.G. Safonov, A.J. Laub, and G.L. Hartmann, *Feedback properties of multi-variable systems: the role and use of the return difference matrix*, IEEE Trans. Autom. Control **26** (1981), no. 1, 47–65.
- [53] M.G. Safonov, D.J.N. Limebeer, and R.Y. Chiang, *Simplifying the \mathcal{H}^∞ theory via loop-shifting, matrix-pencil and descriptor concepts*, Int. J. Control **50** (1989), no. 6, 2467–2488.
- [54] S. Skogestad and I. Postlethwaite, *Multivariable feedback control : Analysis and design*, John Wiley & Sons, 1996.
- [55] J.E. Slotine and W. Li, *Applied nonlinear control*, Prentice Hall, 1991.
- [56] M. Spillman, P. Blue, S. Banda, and L. Lee, *A robust gain-scheduling example using linear parameter-varying feedback*, IFAC 13th Triennial World Congress, 1996, pp. 221–226.
- [57] P. Tsiotras, M. Corless, and M.A. Rotea, *An \mathcal{L}_2 disturbance attenuation solution to the nonlinear benchmark problem*, Int J. Robust and Nonlinear Control **8** (1998), 311–330.
- [58] P.K. Wong, *On the interaction structure of linear multi-input feedback control systems*, Master's thesis, Electrical Engineering, MIT, 1975, Supervised by M. Athans.
- [59] L. Xie and C.E. de Souza, *Robust \mathcal{H}_∞ control for linear systems with norm-bounded time-varying uncertainty*, IEEE Trans. Autom. Control **37** (1992), 1188–1191.
- [60] K. Zhou and J.C. Doyle, *Essentials of robust control*, Prentice-Hall, 1998.

- [61] K. Zhou, J.C. Doyle, and K. Glover, *Robust and optimal control*, Prentice Hall, 1996.
- [62] K. Zhou and P. Khargonekar, *An algebraic Riccati equation approach to \mathcal{H}_∞ optimization*, Systems and Control Letters **11** (1988), 85–91.
- [63] K. Zhou, P.P. Khargonekar, J. Stoustrup, and H.H. Niemann, *Robust performance of systems with structured uncertainties in state space*, Automatica **31** (1995), no. 2, 249–255.