Multiple Robust Controllers Design for Linear Uncertain Systems

by

Ehsan Azadi Yazdi

B.Sc. Mechanical Engineering, Shiraz University, 2005

M.Sc. Mechanical Engineering, Sharif University of Technology, 2007

A THESIS SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIREMENTS FOR THE DEGREE OF

Doctor of Philosophy

in

THE FACULTY OF GRADUATE STUDIES
(Mechanical Engineering)

THE UNIVERSITY OF BRITISH COLUMBIA
(Vancouver)

April 2011

Abstract

This thesis presents two design methods for multiple robust controllers (MRC) and a design method for switched robust controllers that can be used for linear systems with parametric uncertainty. These methods are helpful in enhancing the performance of feedback control systems beyond the performance limitations associated with conventional design methods for uncertain linear systems.

MRC have been introduced to overcome the performance limitations imposed by the requirements for robustness. The design of MRC involves dividing the uncertainty set of a linear system into a number of subsets, as well as synthesizing of a robust controller for each subset. We propose two methods for the design of the uncertainty set divisions and controllers. The first method determines each controller and each subset in subsequent steps, and can deal with only time-invariant parametric uncertainties. In the second method, we find both each controller and each subset simultaneously. This method can handle both time-invariant and time-varying uncertainties.

Switched robust controllers, on the other hand, have the potential to advance the existing performance limitations of a single linear time-invariant robust controller by switching among a set of linear time-invariant controllers. We introduce a new concept, *robust finite-time tracking*, that is a property of closed-loop uncertain systems. Robust finite-time tracking focuses on the transient response, as opposed to the steady-state response (as in Lyapunov-based techniques). We formulate an optimization problem for the design of

switched robust finite-time tracking controllers.

The controller design methods that discussed above include nonsmooth non-convex optimization problems. In order to find a local optimum to the problems, nonsmooth optimization techniques are utilized.

We compare the performance of developed methods to existing methods in the literature through several numerical examples, such as inverted pendulum and mass-spring-damper systems. Moreover, to demonstrate the advantages of our method for MRC over traditional robust controllers in a practical example, we design multiple robust track-following controllers for hard disk drive servo-systems.

Preface

In this section, we briefly explain the contents of the papers that are published or submitted for publications from this thesis [14–19]. We also clarify the relative contributions of co-authors in the papers.

- **E. Azadi Yazdi and R. Nagamune, *"Multiple robust H_∞ controller design using the nonsmooth optimization method"*, International Journal of Robust and Nonlinear Control, 2010, Vol. 20, No. 11, pp. 1197-1212.** This paper proposes a systematic technique to design multiple robust H_∞ controllers. The idea of multiple robust controllers (MRC) is originated in a publication by the research supervisor of this thesis, Dr. Ryozo Nagamune [29]. However, we formulated a more practical MRC design problem in this publication. The proposed technique is explained in Chapter 2 of this thesis. To design MRC an optimization problem is formulated by the author of this thesis. A numerical nonsmooth optimization algorithm is proposed to solve this problem. The author of this thesis was the principal researcher of this publication. Dr. Ryozo Nagamune assisted with formulating the problem and writing the paper.

- **E. Azadi Yazdi, M. Sepasi, F. Sassani and R. Nagamune, *"Automated multiple robust track-following control system design in hard disk drives"*, IEEE Transactions on Control**

System Technology, DOI: 10.1109/TCST.2010.2053541.[1] This paper proposes a new design procedure for track-following control systems in hard disk drives. The procedure is automated, in the sense that, for given experimental frequency response data of the suspension arm dynamics and a model structure, it automatically constructs a model set with parametric uncertainties. Subsequently, for the transfer function set it automatically designs a partition of the uncertainties and corresponding MRC. The first step of the procedure, i.e. model set construction, is developed mainly by Mr. M. Sepasi, and the second step, i.e. MRC design, is developed by the author of the thesis. Experiments on actual hard disk drives demonstrate the usefulness and efficiency of the proposed procedure. The experiments are performed by Mr. M. Sepasi and the author of the thesis. The results of this paper is partly presented in Chapter 3 of the thesis. Professors Farrokh Sassani and Ryozo Nagamune provided practical insight to the problem, and contributed significantly to the writing of this paper.

- **E. Azadi Yazdi and R. Nagamune, *"A parameter set division and switching gain-scheduling controllers design method for time-varying plants"*, submitted for publication.** This paper presents a systematic technique to design switching MRC for plants with time-varying parameters described in Chapter 4 of the thesis. The integral quadratic constraint (IQC) approach is used to setup an optimization problem for the switching MRC synthesis problem. A nonsmooth optimization technique is developed by the author of the thesis to solve this problem. The author of the thesis was the primary researcher of this paper. Dr. Ryozo Nagamune suggested to use the IQC technique to solve the formulated problem and assisted with writing

[1] A brief version is also published in: E. Azadi Yazdi, M. Sepasi, F. Sassani and R. Nagamune, *"Automated multiple robust track-following control system design in hard disk drives"*, 2010 ASME Dynamic Systems and Control Conference, Boston, MA.

the paper.

- **E. Azadi Yazdi and R. Nagamune, *"Robust finite-time tracking with switched controllers"*, submitted for publication.**[2] In this paper, a problem of designing a switched controller for a plant with parametric uncertainty is considered. The design problem amounts to solving an optimization problem that is explained in Chapter 5 of the thesis. The author of this thesis was the main researcher of this paper. Dr. Ryozo Nagamune supervised the research and assisted with writing the paper.

[2] A brief version is also published in: E. Azadi Yazdi and R. Nagamune, *"Switched tracking controller design for uncertain systems"*, 2010 ASME Dynamic Systems and Control Conference, Boston, MA.

Table of Contents

List of Tables

List of Figures

Notation

Symbol	Description
$\mathbb{Z}[1, M]$	Set of all integers between 1 and M
$\mathbb{R}$	Set of all real numbers
$\mathbb{R}^+$	Set of all positive real numbers
$\mathbb{R}^n$	Set of all n-dimensional real vectors
$\mathbb{R}^{n\times m}$	Set of all $n \times m$-dimensional real matrices
$\mathbb{C}$	Set of all complex numbers
$\mathbb{C}^n$	Set of all n-dimensional complex vectors
$\mathbb{C}^{n\times m}$	Set of all $n \times m$-dimensional complex matrices
$Re(C)$	The real part of a complex number C
$Im(C)$	The imaginary part of a complex number C
diag$\{v\}$	A diagonal matrix with elements of vector v as its elements
A^H	The Hermitian transpose of A
Tr(A)	The trace of a square matrix A
$\lambda_i(A)$	Eigenvalues of a square matrix A
$\lambda_{max}(A)$	The largest eigenvalue of a Hermitian matrix A
$\bar{\sigma}(A)$	The maximum singular value of a Hermitian matrix A
$< X, Y >$	The scalar product defined on the space $\mathbb{M}^{n\times m}$ of $n \times m$ matrices, defined as $< X, Y >= \text{Tr}(X^H Y)$

$\|A\|$ The 2-norm of a constant vector or matrix A is defined by $\|A\| := \sqrt{\text{trace}(A^T A)}$

$\|v\|_2$ The L_2-norm of a time-domain signal $v : \mathbb{R}^+ \to \mathbb{R}^n$, given by $\|v\|_2 := \int_0^\infty \sqrt{v^T(t)v(t)}dt$

$\|z\|_\infty$ The ∞-norm of a constant vector z, given by $\|z\|_\infty := \max_i |z(i)|$

$\|v\|_\infty$ The l_∞-norm of a vector-valued time-domain signal $v(t) = [v_1(t), \cdots, v_n(t)]^T$, defined by $\|v\|_\infty := \max_{t\geq 0} \{\max_{i=1,\ldots,n} |v_i(t)|\}$

$\|v\|_{t_f}$ The finite-horizon l_∞-norm of a vector-valued time-domain signal $v(t) = [v_1(t), \cdots, v_n(t)]^T$, defined by $\|v\|_{t_f} := \max_{0\leq t\leq t_f} \{\max_{i=1,\ldots,n} |v_i(t)|\}$

L_∞^n Set of all n-dimensional vector-valued time-domain signals with bounded l_∞-norm on $\mathbb{R}^+$

$L_{t_f}^n$ Set of all n-dimensional vector-valued time-domain signals with bounded finite-horizon l_∞-norm on $[0, t_f]$

$\|A\|_\infty$ The ∞-norm of a constant matrix A, given by $\|A\|_\infty := \max_i \left\{\sum_j |A(i,j)|\right\}$

$\|F\|_\infty$ The H_∞-norm of a given rational stable transfer function F, defined by $\|F\|_\infty := \sup_{\omega\in\mathbb{R}^+} \bar{\sigma}(F(j\omega))$

$\mathbb{RH}_\infty$ The space of all rational proper stable transfer functions

F_l The lower linear fractional transformations, defined by $F_l\left(\begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}, N\right) := M_{11} + M_{12}N\left(I - M_{22}N\right)^{-1} M_{21}$

F_u The upper linear fractional transformations, defined by $F_u\left(\begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}, N\right) := M_{22} + M_{21}N\left(I - M_{11}N\right)^{-1} M_{12}$

$\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right]$ Transfer function representation of a state space realization (A, B, C, D) given by $C(sI-A)^{-1}B+D$

Glossary

BMI bilinear matrix inequality

CLF common lyapunov function

FDI frequency domain inequality

FRF frequency response function

HDD hard disk drive

IFT iterative feedback tuning

IQC integral quadratic constraint

LDV laser doppler vibrometer

LMI linear matrix inequality

LTI linear time-invariant

LTV linear time-varying

MA microactuator

MRC multiple robust controllers

PES position error signal

R/W head read/write head

RFTT robust finite-time tracking

RMS root mean square

RFTTC robust finite-time tracking controllers

SDP semi-definite program

VCM voice coil motor

Acknowledgments [3]

I would like to express my deepest appreciation to my research supervisor, Dr. Ryozo Nagamune, for his encouragement, inspiration and patience. He has showed me an incredible view of science as well as life. I am really thankful for his unconditional support and genuine care through my studies.

I would like to convey my gratitude to the members of my research committee Drs. Farrokh Sassani, Yusuf Altintas, and Meeko Oishi for being generous with their time and providing invaluable comments.

I am indebted to my dear colleagues, Mohammad, Marius, and Masih, in Control Engineering Laboratory. They have been friendly, warm, and willing to help.

I will never forget good memories that I have with my fiends in Vancouver. I have had some of my deepest friendships with Ali, Amir, Ashkan, Behnam, Elyas, Farzad, Hamed, Hedyeh, Mahkameh, Noushin, Pirooz, Sara, Shahla, Sima, and Raha. I was lucky to have wonderful roommates Rasool and Sina.

Lastly, I wish to express my genuine gratitude to my parents whose love, encouragement and moral support have always been a motivation for me.

[3]This research was financially supported by the Natural Sciences and Engineering Research Council (NSERC) of Canada, Faculty of Graduate Studies (FoGS) of the University of British Columbia, Ministry of Advanced Education of the Province of British Columbia, Canada Foundation for Innovation (CFI), and the Institute for Computing, Information and Cognitive Systems (ICICS).

Dedication

بنگر به جهان، چه طرف بربستم؟ هیچ

وز حاصل عمر چیست در دستم؟ هیچ

شمع طربم ولی چو بنشستم، هیچ

من جام جمم ولی چو بشکستم، هیچ

حکیم عمر خیام

Chapter 1

Introduction

1.1 Motivation

Recent trends in the high-technology industry, such as in machine tools, automotive technology, and information technology, show the need for improvements in the performance of feedback control systems [47, 84, 93]. Also, in some applications, such as ball-screw drives and hard disk drive (HDD) servo systems, the controllers need to maintain robustness of feedback systems against unavoidable uncertainty in the systems. Robust precision controllers are one solution to achieve high performance and to maintain robustness simultaneously [20, 76, 105, 112].

Nevertheless, several practical and theoretical performance limitations constrain the attainable performance of robust control systems [92]. Examples that cause such limitations are excessive noise, inadequate actuation forces, large uncertainty in the dynamic model of the plant, right half-plane poles/zeros and time delay, and use of a linear time-invariant (LTI) controller for LTI plants. Many attempts have been made to remove these performance limitations in robust precision control [29, 34, 68].

This thesis aims to developing robust control methods that can achieve performances beyond the two well-known limitations associated with conven-

tional methods; i.e., limitations due to uncertainty in the dynamic model of uncertain plants, and limitations due to the use of an LTI controller for LTI uncertain plants.

1.2 Methodology and problem definition

To circumvent the limitations caused by the uncertainty in the plants and the use of an LTI controller for LTI uncertain plants, multiple robust controllers (MRC) [29] and switched robust controllers [79] have been developed, respectively. This section briefly describes these control methods and provides controller design problem formulations. Here, to prevent confusing notations and excessive introductory material, the controller design problems are stated in an informal form. The problems are revisited and reformulated precisely in later chapters of the thesis.

1.2.1 Multiple robust controllers

In robust control, while improving the performance of the system, robustness must be maintained against conceivable uncertainty in system dynamics. However, uncertainty and performance are well known to be conflicting properties [113]. To overcome the conflict, the MRC method was proposed in [29, 30]. In the MRC method, the model set is divided into a number of subsets, and for plants in each subset, a controller is specialized to achieve a robust performance. This is in contrast to conventional robust control methods, where only one controller is used to meet the required robust performance for all plants. Clearly, MRC method improves the closed-loop performance, since each controller only has to be robust for a subset of the model set, rather than for the entire set.

Let us verify the improvement in the closed-loop performance due to use of the MRC method through a simple example. Consider a second order

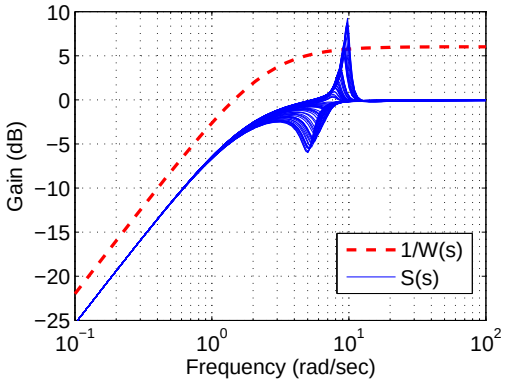


Figure 1.1: Bode diagram of the closed-loop sensitivity for the single controller.

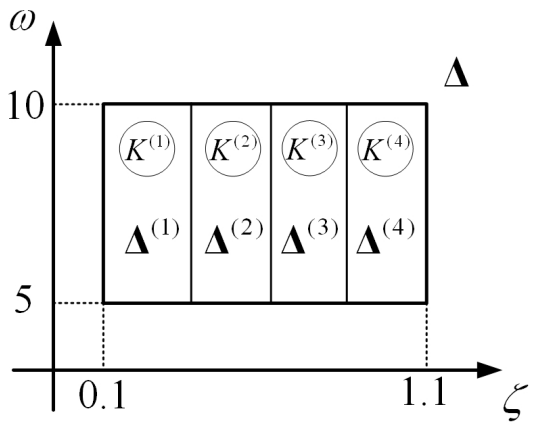
uncertain plant defined by

$$P(s) = \frac{\omega^2}{s^2 + 2\zeta\omega s + \omega^2}, \tag{1.1}$$

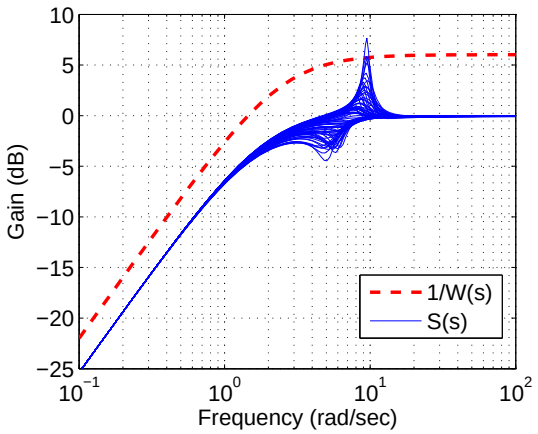
where, $\omega \in [5, 10]$, and $\zeta \in [0.1, 1.1]$ are uncertain time-invariant parametric uncertainties. Suppose, we want to design a controller $K(s)$ such that the sensitivity transfer function ($S(s) := (1 + P(s)K(s))^{-1}$) for all plants is less than the weighting function $W(s)$ given by

$$W(s) = \frac{0.5s + 1.257}{s + 0.001257}. \tag{1.2}$$

In traditional robust control the goal has to be met by only one controller. The Bode diagram of the closed-loop sensitivity for a number of randomly sampled plants equipped with a single controller is shown in Figure 1.1. Ideally, all sensitivity Bode diagrams should lie under the Bode diagram of the weighting function shown by the dashed line. However, in this example, a few closed-loop sensitivity transfer functions violate the constraint imposed by



(a) The uncertainty set divisions.

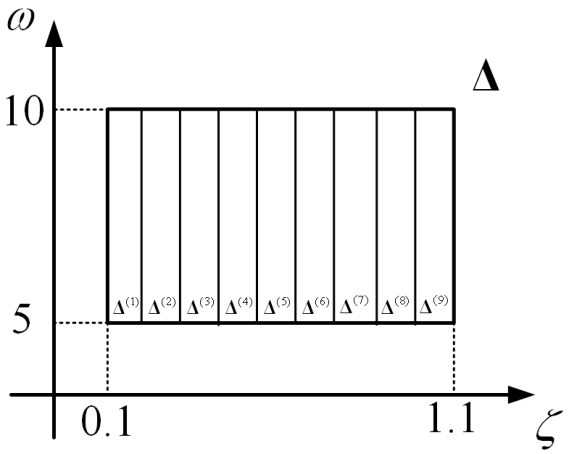


(b) Bode diagram of the closed-loop sensitivity.

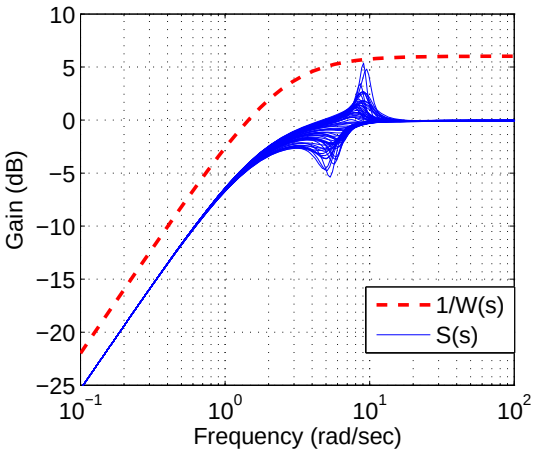
Figure 1.2: The MRC with 4 uncertainty set divisions.

the weighting function. Therefore, a single robust controller cannot achieve the desired performance.

Let us construct the MRC for the plant (1.1). We divide the two-dimensional uncertainty set $\mathbf{\Delta}$, with coordinates (ζ, ω), into four evenly distributed subsets $\{\mathbf{\Delta}^{(i)}\}_{i=1}^{4}$, and assign a controller $K^{(i)}$ to each subset $\mathbf{\Delta}^{(i)}$ as shown in Figure 1.2(a). The closed-loop sensitivity Bode diagrams are



(a) The uncertainty set divisions.



(b) Bode diagram of the closed-loop sensitivity.

Figure 1.3: The MRC with 9 uncertainty set divisions.

shown in Figure 1.2(b). Similar to the single controller case, a number of the sensitivity plots violate the constraint imposed by the weighting function. However, the amount of violation is noticeably smaller than the single controller (Figure 1.1).

Now, the question is: What is the sufficient number of uncertainty set

divisions to achieve the desired performance objective? In the above simple example, we can robustly satisfy the desired performance by dividing the uncertainty set into 9 subsets, as shown in Figure 1.3. For a general plant the answer to above question is found by solving the following *MRC design problem*:

Problem 1.2.1. *Given a linear uncertain plant with an uncertainty set* $\mathbf{\Delta}$ *and a performance requirement, design a set of uncertainty subset and controller pairs* $\{(\mathbf{\Delta}^{(i)}, K^{(i)})\}$*, such that*

(i) the uncertainty set $\mathbf{\Delta}$ *is covered by the union of subsets* $\{\mathbf{\Delta}^{(i)}\}$*, i.e.* $\mathbf{\Delta} \subset \bigcup_i \mathbf{\Delta}^{(i)}$*,*

(ii) the closed-loop uncertain system is stable[1] *for all values of uncertainty in the set* $\mathbf{\Delta}$*,*

(iii) robust performance is met for all plants equipped with the designed controllers.

Remark. The MRC design is computationally more expensive and consequently slower than a traditional robust controller design for two reasons. First, multiple controllers need to be synthesized, and design of each controller involves a non-convex optimization problem[2]. Second, because of the extra amount of computations required for dividing the uncertainty set to achieve the performance requirement. Hence, high computational efficiency is essential for MRC design methods.

1.2.2 Switched robust controllers

The design and analysis of switched controllers have been studied for many years, because of their numerous applications in mechanical systems, power

[1] Various types of stability, such as Lyapunov stability or internal stability, can be used here. Later chapters clarify the types of stability that are sought in this thesis.

[2] The numerical solution of a non-convex optimization problem does not necessarily converge to a global optimum.

systems, network control, aircraft and traffic control and many other areas; see [52, 73] and references therein. The primary objectives in using switched controllers are to overcome limitations due to use of an LTI controller for an LTI plant [25, 41, 64, 79], to improve the robustness against uncertainty [34, 86, 97], and to deal with logic-based systems [50]. Here, we use switched robust controllers to make closed-loop systems free from limitations caused by the use of an LTI controller for an LTI plant.

To overcome the limitation, a set of LTI controllers is designed to achieve various, possibly conflicting, performance requirements. A supervisory system decides which controller in the controller set should be active at each instant of time. A typical block diagram for such controllers is shown in Figure 1.4, where $\{K_i\}_{i=1}^{M}$ is the set of LTI controllers and the switching signal $\sigma(t)$ determines the active controller.

A wide variety of design methods have been developed for switched controllers to achieve various robustness requirements [75, 101]. This thesis introduces the new concept of a robustness requirement, called *robust finite-time tracking (RFTT)*. Roughly speaking, the RFTT ensures that the distance between trajectories of an uncertain system and a desired trajectory

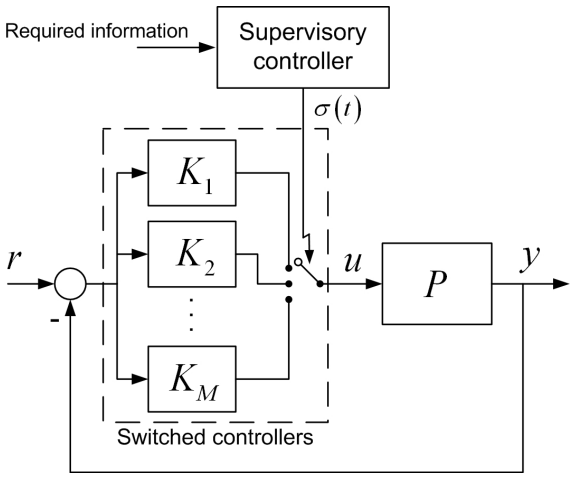
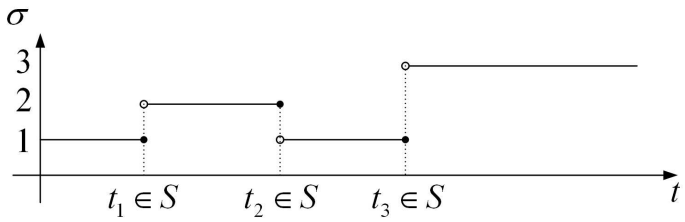
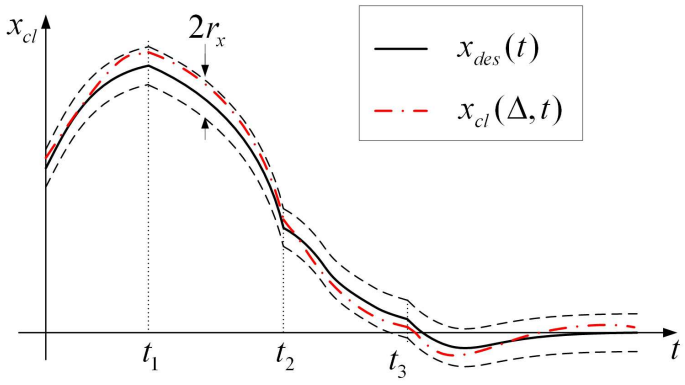


Figure 1.4: Block diagram of switched controllers.



(a) Switching signal $\sigma(t)$.



(b) Closed-loop state x_{cl}.

Figure 1.5: A switched system with RFTT property.

is less than a user-specified bound for all allowable uncertainty values in a finite time horizon. Figure 1.5 depicts an illustration of a switched system with the RFTT property. The controller set consists of three members. For the given switching signal $\sigma(t)$ in Figure 1.5(a), the perturbed trajectory $x_{cl}(\Delta, t)$ stays in a ball of radius r_x around the desired trajectory $x_{des}(t)$. The RFTT property is useful in practical applications with strict requirements on the tracking tolerance. Examples of those applications are the track-following control in hard disk drives and tracking control in machine tools, where a strict tracking tolerance secures the read/write process and machining tolerance, respectively. The *switched controllers design problem with robust finite-time tracking requirement* is formulated as follows:

Problem 1.2.2. *Given a linear uncertain plant, a set of performance requirements, a desired trajectory, and an allowable tracking tolerance, design*

a set of controllers $\{K^{(i)}\}$*, such that*

(i) each performance requirement is achievable by at least one controller,

(ii) the switched closed-loop system is stable,

(iii) the switched closed-loop system has the RFTT property with respect to the desired trajectory and the allowable tracking tolerance.

Remark. The design of the supervisory controller is not within the scope of this thesis. Consequently, the switching signal $\sigma(t)$, that is generated by the supervisory controller, is assumed to be unknown. Hence, the condition (ii) and (iii) of the Problem 1.2.2 should hold for any arbitrary switching signal $\sigma(t)$.

1.3 Objectives of the thesis

Both the multiple robust controllers and the switched robust controllers design problems have been previously addressed in the literature. Nevertheless, the solutions proposed previously often suffer from a lack of generality and/or excessive conservatism, as discussed in Section 1.4. Therefore, these problems are still considered to be unsolved. The objectives of this thesis are to

O1. develop a MRC design technique that

(O1-i) designs the MRC in a computationally efficient manner,

(O1-ii) is applicable to plants with various types of uncertainty such as LTI uncertainty and linear time-varying (LTV) uncertainty,

O2. provide a switched controllers design method that ensures the RFTT for any arbitrary switching signal.

In this thesis, general plant dynamics and control configurations are considered. The proposed controller design methods will be compared to existing

methods from the literature. The controller design methods are also numerically and/or experimentally validated, using typical benchmark problems from the control literature, such as the inverted pendulum or mass-spring-damper systems, as well as a practical control problem, namely, the hard disk drive servo.

1.4 Literature review

This section reviews some of the work done by other researchers that is relevant to the problems formulated in Section 1.2. The aim of this section is to clarify the importance of the objectives of the thesis. Also, due to the significant role of nonsmooth analysis in this thesis, Section 1.4.3 provides a brief literature survey on nonsmooth optimization and its applications in control theory.

1.4.1 Multiple robust control

In this thesis, the MRC design problem is considered for two types of linear uncertain plants; specifically, plants with time-invariant parametric uncertainty, and plants with time-varying parametric uncertainty. Next, we will review previous work on MRC design methods for uncertain LTI and LTV plants.

Multiple robust control for LTI plants

The MRC method is firstly introduced in [29, 30] for LTI plants. The MRC method of [29, 30] partitions the uncertainty set into a pre-specified number of disjoint subsets, and designs a robust controller for each subset. The robust performance can be improved with the MRC method, because each controller has to be robust for only one uncertainty subset, rather than for the entire uncertainty set. Pre-specifying the number of partitions, the MRC design in [30] uses an iterative method to find the partition that reduces the

worst performance over all subsets. During the iterations, the boundaries of subsets are modified, and the robust controllers of the modified subsets are redesigned accordingly.

The MRC method in [30] uses a bilinear matrix inequality (BMI) approach to design the robust controllers. This method is computationally slow as a result of two facts. First, the BMI approach involves Lyapunov variables whose number grows quadratically with the number of states [61, 95]. Second, the controllers need to be redesigned many times during the iterations.

Chapter 2 proposes a systematic method for designing MRC. The proposed method has two main advantages. First, this method is beneficial over the method in [29, 30], in that the sufficient number of subsets for achieving the pre-specified performance objective will be determined automatically, in contrast to the method in [29, 30] where the number of uncertainty subsets must be pre-specified by the user. Second, this method avoids the previously discussed computational problems of the MRC method in [29, 30]. Because in the proposed method, once the robust controller is designed for a subset, we do not change it, while in [29, 30], the robust controller for each subset must be synthesized many times during the iterations. In addition, because the proposed method does not involve Lyapunov variables, the design of each controller is computationally faster than when using Lyapunov variables, especially for high order plants.

Multiple robust control for LTV plants

For LTV plants, gain-scheduling controllers have been widely used [68]. The gain-scheduling controller utilizes a measurement or an estimation of time-varying parameters to enhance system performance. A key step in the gain-scheduling controller design is the search for appropriate Lyapunov functions that meet the stability condition and a performance bound for the closed-loop system [48]. The search can be conducted mainly by two approaches: a common lyapunov function (CLF) [4, 21] or a parameter-dependent Lya-

punov functions [44, 100]. The latter approach can incorporate knowledge about the rate of variation of the time-varying parameters, and therefore, leads to less conservative controllers. Nevertheless, this approach can still be conservative in two situations: for plants with a large parameter set, or for plants with uncertainty that is not possible to estimate.

To improve the performance of the traditional gain-scheduling methods for plants with a large parameter set, the parameter set can be divided into a number of subsets, and a controller is designed for each subset [78, 107]. This approach, which is similar to the MRC approach, can achieve higher performances compared to traditional methods. In this method, the controller is scheduled based on the time-varying parameters, which is different from the MRC method for LTI plants where the controller is fixed.

The methods of [78, 107] share two main shortcomings, despite of their differences. First, they lack a systematic procedure for dividing the parameter set. In the methods of [78, 107], user-specified divisions will not be updated during the controller design, which may lead to unsatisfactory results. Therefore, users should follow a trial and error method for the number of divisions and their locations to achieve the desired performance specification. Second, these methods are not suitable for uncertain plants that have both time-varying and time-invariant uncertainty.

In Chapter 4, a switching multiple robust control method is proposed that removes both of the above shortcomings. This method extends the systematic parameter set division and controller design method of Chapter 2 to time-varying uncertain plants. Hence, our method automates the parameter set division that replaces the time-consuming trial and error division process of [78, 107]. Moreover, the proposed method is applicable to both time-varying and time-invariant uncertainty plants.

1.4.2 Switched robust controllers

The stability of the closed-loop system with a switched controller is a delicate issue, because even when each controller stabilizes the plant on its own, the stability of the switched closed-loop system is not guaranteed [36, 72]. Several stability analysis methods based on the *Lyapunov stability* approach have been proposed for switched dynamic systems; e.g., [35, 51]. It is well known that Lyapunov stability does not quantify the transient behavior of systems; however, in some applications, the concern is the transient behavior of systems. The concepts of *practical stability* and *finite-time stability* were introduced to deal with the transient behavior of systems over infinite and finite time horizons, respectively [38]. In switched systems, these stability measures are of interest, especially when individual modes do not share a common equilibrium [103, 104, 109].

The introduction of uncertainty in the switched *robust* controllers design increases the complexity of the design problem because the closed-loop system should not only maintain its stability, but also its performance in the presence of arbitrary switching and plant uncertainty. Although significant progress has been made in the Lyapunov stability approach to switched robust controllers (e.g., [75, 94, 101, 102, 111]), the practical stability and the finite-time stability counterparts remained relatively untouched (e.g., [2, 38]).

In Chapter 5, following the finite-time stability approach, the notion of robust finite-time tracking (RFTT) is introduced. A novel switched robust controller design method is developed, based on the RFTT definition. The proposed method has two main advantages. First, it provides an extension to the finite-time stability concept for plants with uncertainty. Second, this method designs controllers for time-domain performance specifications without converting them into frequency-domain specifications. Most conventional robust controller design methods convert time-domain specifications into frequency-domain specifications through several approximations; e.g., [49, 56, 108]. Although time-domain specifications can be handled using

these methods, they often produce conservative results.

1.4.3 Nonsmooth optimization

The robust control synthesis problems that are considered in this thesis lead to optimization problems with a nonsmooth cost function. An example of nonsmooth functions is shown in Figure 1.6. Although the function is *globally continuous*, it is not *globally differentiable*, because in a number of points, such as x_2 and x_3, the right and left derivatives do not match. The nonsmooth analysis [31, 32] provides necessary concepts and computational tools to define the derivative, known as the *subdifferential*, at those points. As shown in Figure 1.6, at points x_2 and x_3 we define a cone of *subgradients* shown by the arrows and the arcs between them. Consequently, the tangent to the $f(x)$ (i.e. subdifferentials) also consists of a cone at those points, in contrast with regular points, such as x_1, where the tangent (i.e. derivative) is a line.

In robust control, nonsmooth optimization was first applied to H_∞ controller design problems. Since the early-1990s, the advent of linear matrix inequalitys (LMIs) introduced a new paradigm in robust control, and specifically, in H_∞ control, see for example [24, 39, 43, 90]. In the LMI-based controller design methods, Lyapunov functions are used to guarantee stability and H_∞ performance. The use of Lyapunov functions in the LMI methods

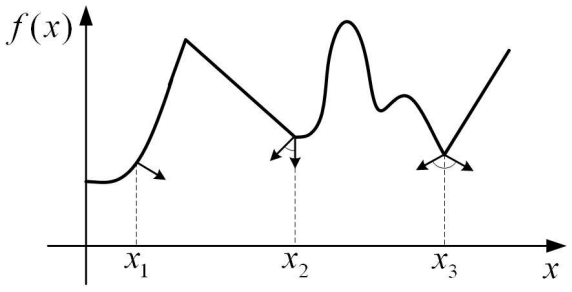


Figure 1.6: Illustration of a nonsmooth function.

converts the H_∞ controller design problem into a convex optimization problem. Nevertheless, the introduction of Lyapunov variables turned out to be a bottleneck of LMI methods for high order plants, because the number of LMI variables (i.e., the dimensions of the optimization problem) grows quadratically with the order of the plant. Consequently, other non-LMI methods were developed to eliminate this disadvantage [5, 9, 45].

One of the non-LMI methods, proposed to avoid the computational problems caused by the introduction of Lyapunov variables, is the nonsmooth controller synthesis method [7, 9]. This method checks the finiteness of H_∞-norm of the system instead of using the Lyapunov functions to ensure the closed-loop stability. Although the nonsmooth approach deals with nonconvex optimization, it has recently received much attention due to the elimination of Lyapunov variables, leading to a much faster computational method relative to traditional LMI methods.

The idea of the nonsmooth H_∞ synthesis was initially suggested by [9], developed further by [6, 12], and extended in various directions by [12, 16, 23]. The use of the nonsmooth optimization technique not only improves the computational efficiency of the controller design but also provides useful design tools, such as multiband [11] and multidisk [10] control techniques.

In this thesis, the nonsmooth optimization approach of [9, 31] is used to solve optimization problems. Further information about this approach is provided in the main body of the thesis.

1.5 Organization of the thesis

This thesis is organized as follows. Chapters 2 and 4 are devoted to Objective (O1) of the thesis, presented in Section 1.3. To be more specific, Chapter 2 proposes a computationally efficient MRC for LTI plants that meets Objective (O1-i). The proposed MRC for LTI plants is then used to design track-following controllers for a HDD read/write head positioning system in Chapter 3. The performance of the proposed method is compared to that of

a traditional robust controller by simulations and experiments. The parameter set division method of Chapter 2 is further developed for LTV uncertain plants in Chapter 4 to accomplish Objective (O1-ii). Chapter 5 formulates and solves a switched RFTT controller problem that achieves Objective (O2) of the thesis, presented in Section 1.3. Finally, Chapter 6 concludes the thesis, summarizes its major contributions, and provides directions for future research.

Chapter 2

Multiple Robust Control for Linear Time-Invariant Uncertain Plants [1]

2.1 Introduction

This chapter proposes a systematic method to design multiple robust controllers (MRC) for plants with time-invariant parametric uncertainty that achieves Objective (O1-i) of Section 1.3. This method consists of iterations of two steps. In the first step, for a given parametric uncertainty set, we choose (a) a plant within the set and (b) a controller to achieve the "best" closed-loop performance. It turns out that the problem of finding such a plant-controller pair entails to solving a nonsmooth optimization problem, to which a nonsmooth gradient-based numerical method is applied. In the second step, for a given performance and the controller designed in the first step, we design, within the original uncertainty set, a largest subset with the obtained plant as its nominal plant, so that all the plants in the subset satisfy the desired closed-loop performance. The largest uncertainty subset can be found based

[1] This chapter is based on the following publication: E. Azadi Yazdi and R. Nagamune, *"Multiple robust H_∞ controller design using the nonsmooth optimization method"*, International Journal of Robust and Nonlinear Control, 2010, Vol. 20, No. 11, pp. 1197-1212.

on the standard robustness analysis tools, such as small gain theorem and μ-analysis [85].

This chapter is organized as follows. Section 2.2 formulates the MRC design problem. In Section 2.3, we propose an algorithm to synthesize MRC. Three main steps of this algorithm are discussed in details in Section 2.3. Section 2.3.1 formulates an optimization problem to find the center (called nominal) of each uncertainty subset. Section 2.3.2 finds a subset around the nominal value such that the robust performance is guaranteed for all elements of this subset. Based on the subset, Section 2.3.3 introduces an algorithm to determine the partitions. Section 2.4 introduces a nonsmooth numerical optimization method for the optimization problem formulated in Section 2.3.1. Section 2.5 presents a numerical example to validate our algorithm.

2.2 Multiple robust controllers design problem for linear time-invariant (LTI) plants

We consider an LTI plant with real parametric uncertainty $\Delta \in \mathbf{\Delta}$ expressed by the state space equation (see Figure 2.1):

$$\begin{bmatrix} \dot{x}(t) \\ z(t) \\ y(t) \end{bmatrix} = F_u(P, \Delta) \begin{bmatrix} x(t) \\ w(t) \\ u(t) \end{bmatrix}, \tag{2.1}$$

Here, $\mathbf{\Delta} \subset \mathbb{R}^{n_\Delta \times n_\Delta}$ is a bounded set of real structured time-invariant uncertainty, $x(t) \in \mathbb{R}^n$ is the state vector, $w(t) \in \mathbb{R}^{n_w}$ the exogenous input vector, $u(t) \in \mathbb{R}^{n_u}$ the control input vector, $z(t) \in \mathbb{R}^{n_z}$ the performance vector, $y(t) \in \mathbb{R}^{n_y}$ the measurement vector, and $P \in \mathbb{R}^{(n_\Delta+n+n_z+n_y)\times(n_\Delta+n+n_w+n_u)}$ the system matrix. We denote the transfer function from $\begin{bmatrix} u_\Delta(s)^T & w(s)^T & u(s)^T \end{bmatrix}^T$ to $\begin{bmatrix} y_\Delta(s)^T & z(s)^T & y(s)^T \end{bmatrix}^T$ by $P(s)$, and the closed-loop transfer function

from $w(s)$ to $z(s)$ by $T_{w\to z}(K,\Delta)$, where K is the controller.

The MRC synthesis problem is formulated as follows.

Problem 2.2.1. *For a given uncertain LTI plant (2.1) with the uncertainty set* $\mathbf{\Delta}$ *and a desired robust performance objective* γ*, design a set of uncertainty subset and controller pairs* $\{(\mathbf{\Delta}^{(i)}, K^{(i)})\}_{i=1}^{n_d}$*, such that*

(i) $\mathbf{\Delta} = \bigcup_{i=1}^{n_d} \mathbf{\Delta}^{(i)}$,

(ii) The performance objective is satisfied in all subsets, i.e.

$$\left\| T_{w\to z}(K^{(i)}, \Delta) \right\|_\infty < \gamma \qquad , \forall \Delta \in \mathbf{\Delta}^{(i)}. \tag{2.2}$$

Remark.

- We emphasize that the number of uncertainty subsets n_d is not prespecified in this problem setting; it will be automatically determined by the proposed algorithm.
- Mass produced systems, such as hard disk drives, may have two different sources for parametric uncertainty:

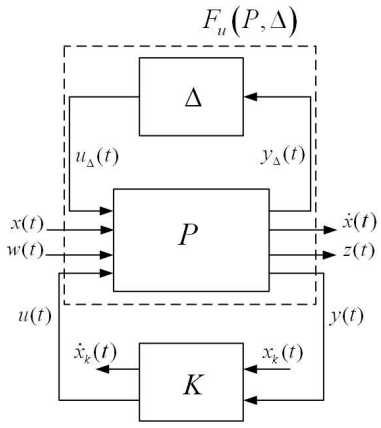


Figure 2.1: A generalized plant P, with the uncertainty Δ, and the controller K.

a) variations in the conditions of the fabrication line,

b) variations due to elapse of time.

Uncertainty caused by the former, respectively latter, source can be treated as time-invariant, respectively time-varying, uncertainty. In some systems, uncertainty set due to the former source is much larger than uncertainty set due to elapse of time. This chapter focuses on the time-invariant uncertainty of these systems. In such systems, the closed-loop performance will degrade by a large factor if the controller is designed to perform robustly against variations between different products. Chapter 4 develops a method that can deal with both time-varying and time-invariant uncertain elements.

- Since uncertainty is assumed to be time-invariant in this chapter, once a controller is chosen for a system based on its identified parameters, it will not be changed. Therefore we do not investigate switching controller scheme and its related potential issues.

For a given performance objective γ, if a robust controller can be found without partitioning $\mathbf{\Delta}$, the problem is just a standard robust control synthesis problem, and there are design techniques and software available, see, e.g., [28, 62, 96, 114]. Here, our interest is in the case when such a single robust controller cannot be found. In some applications, we are not allowed to sacrifice performance for maintaining robustness. The MRC method is one way to push forward the robust performance by introducing extra controllers and by reducing the region that one controller is assigned to.

One method to solve the MRC synthesis problem was proposed in [30]. This method tries to improve the robust performance with a pre-specified number of uncertainty subsets. Pre-specifying the number of subsets is a drawback of this method, because if the required performance objective is not met at the end of iterations, the design should be repeated with a greater number of partitions.

To overcome this drawback, our aim is to provide a systematic MRC design method to achieve a desired H_∞ norm γ over entire uncertainty set. Moreover, to decrease the computational effort, this task should be accomplished with small number of partitions.

2.3 MRC synthesis algorithm

This section proposes a MRC synthesis method. First, we present the algorithm, and then discuss each step of the algorithm in details (see Figure 2.2 for the graphical interpretation of the algorithm). In the algorithm, we denote the unpartitioned uncertainty subset by $\mathbf{\Delta}_{\text{unpart}}$.

Algorithm 2.3.1. *Suppose that the system (2.1), the uncertainty set $\mathbf{\Delta}$, and the desired robust performance objective γ are given.*

0. ***Start:*** *Set $i = 1$ and choose the unpartitioned uncertainty set as $\mathbf{\Delta}_{\text{unpart}} = \mathbf{\Delta}$.*

1. ***Find a nominal uncertainty value and a controller:*** *Find a nominal uncertainty $\Delta_{\text{nom}} \in \mathbf{\Delta}_{\text{unpart}}$ and a controller which minimizes the H_∞ norm of the performance channel (Figure 2.2(a)).*

   - *If the minimized H_∞ norm is not less than γ, terminate the algorithm; In this case, the uncertain plant cannot satisfy the desired performance.*
   - *Otherwise, go to step (2).*

2. ***Find the uncertainty subset $\mathbf{\Delta}^{(i)}$:*** *Find an uncertainty subset $\mathbf{\Delta}^{(i)} \subset \mathbf{\Delta}_{\text{unpart}}$ around the nominal value Δ_{nom} which meets the desired robust performance γ (Figure 2.2(b)).*

3. ***Partition $\mathbf{\Delta}_{\text{unpart}}$:*** *Based on the $\mathbf{\Delta}^{(i)}$, partition the $\mathbf{\Delta}_{\text{unpart}}$ with a certain rule which will be explained later (Figure 2.2(c)).*

4. ***Redefine the unpartitioned space:*** *Exclude the newly established subset* $\mathbf{\Delta}^{(i)}$ *from the uncertainty set as*

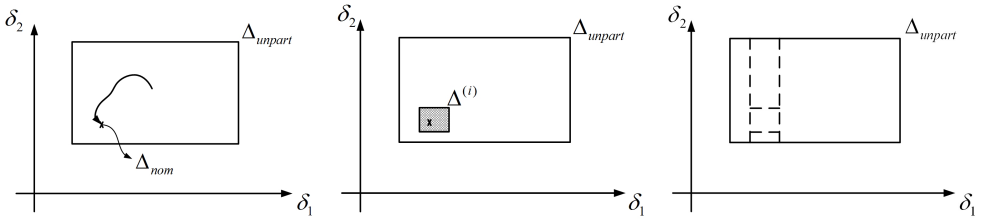
$$\mathbf{\Delta}_{\text{unpart}} := \mathbf{\Delta}_{\text{unpart}} - \mathbf{\Delta}^{(i)}. \tag{2.3}$$

5. *Set* $i = i + 1$ *and go to Step (1) until the unpartitioned uncertainty set* $\mathbf{\Delta}_{\text{unpart}}$ *becomes an empty set.*

Remark. In practice, a "very small" uncertainty subset $\mathbf{\Delta}^{(i)}$ is not favorable. Hence, a lower bound on the size of the uncertainty subset $\mathbf{\Delta}^{(i)}$ should be considered in Step 2 of Algorithm 2.3.1. Imposing the lower bound also guarantees the convergence of Algorithm 2.3.1.

2.3.1 Finding a nominal uncertainty value and a controller

This section formulates an optimization problem to find a nominal uncertainty value and a controller which minimizes the H_∞ norm of the closed-loop system. We will show that, by choosing such uncertainty value as the center of the uncertainty subset, we can achieve a larger uncertainty subset in Step (2) of Algorithm 2.3.1. This leads to a fewer number of divisions to ensure robust performance over the entire uncertainty set.



(a) Algorithm 2.3.1 - Step (1).

(b) Algorithm 2.3.1 - Step (2).

(c) Algorithm 2.3.1 - Step (3).

Figure 2.2: Graphical interpretation of Algorithm 2.3.1 for a two dimensional uncertainty set.

Let us define an operator $\mathcal{F}$ by

$$\mathcal{F}(K, \Delta_{\text{nom}}) := F_l \left(F_u \left(P(s), \begin{bmatrix} 0 & I \\ I & \Delta_{\text{nom}} \end{bmatrix} \right), K \right). \tag{2.4}$$

The operator $\mathcal{F}(K, \Delta_{\text{nom}})$ computes the closed-loop nominal system for the nominal uncertainty value Δ_{nom} and the controller K. This specific structure for the operator $\mathcal{F}(K, \Delta_{\text{nom}})$ is chosen in such a way that if we connect an uncertainty block Δ to $\mathcal{F}(K, \Delta_{\text{nom}})$, as shown in Figure 2.3, the resulting closed-loop system represents the original closed-loop system (Figure 2.1) with an uncertainty block $\Delta + \Delta_{\text{nom}}$. For a given Δ_{nom}, the admissible range of the uncertainty Δ in Figure 2.3 can be determined from the constraint $\Delta + \Delta_{\text{nom}} \in \mathbf{\Delta}$.

The following optimization problem is formulated to find the nominal

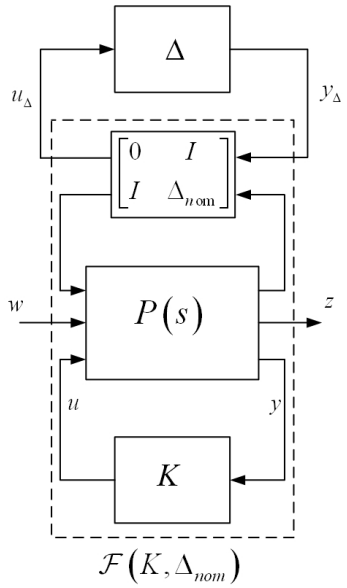


Figure 2.3: General control configuration of a plant with nominal uncertainty value Δ_{nom}.

plant and the corresponding controller in each uncertainty subset.

$$\left(K^{(i)}, \Delta_{\text{nom}}\right) = \underset{\substack{K \in \text{Stabilizing} \\ \Delta \in \mathbf{\Delta}_{\text{unpart}}}}{\boldsymbol{argmin}} \left\{ \left\| \mathcal{F}(K, \Delta) \right\|_\infty \right\}, \tag{2.5}$$

where $\mathbf{\Delta}_{\text{unpart}}$ is the unpartitioned uncertainty subset. The reason for minimizing H_∞-norm $\left\| \mathcal{F}(K, \Delta_{\text{nom}}) \right\|_\infty$ is that the smaller the H_∞-norm is, the larger the $\mathbf{\Delta}^{(i)}$ would be in Step (2), leading to the small number of subsets in the partition. This is presented in the following lemma. Hereafter, we assume that the robust performance objective γ is normalized to be one.

Lemma 2.3.2. *Suppose that Δ_1 and Δ_2 are two fixed matrices, and K_1 and K_2 are the corresponding stabilizing controllers for these matrices such that*

$$\left\| \mathcal{F}(K_1, \Delta_1) \right\|_\infty < \left\| \mathcal{F}(K_2, \Delta_2) \right\|_\infty. \tag{2.6}$$

If there exists a simply connected uncertainty set $\tilde{\mathbf{\Delta}}_2$ including zero such that

$$\left\| \Delta_{\text{aug}} \right\|_\infty \left\| \mathcal{F}(K_2, \Delta_2) \right\|_\infty < 1, \quad \Delta_{\text{aug}} := \begin{bmatrix} \Delta & \\ & \Delta_P \end{bmatrix} \tag{2.7}$$

holds for all $\Delta \in \tilde{\mathbf{\Delta}}_2$, and $\left\| \Delta_P \right\|_\infty \leq 1$, then there exists a $\tilde{\mathbf{\Delta}}_1 \supseteq \tilde{\mathbf{\Delta}}_2$ such that

$$\left\| \Delta_{\text{aug}} \right\|_\infty \left\| \mathcal{F}(K_1, \Delta_1) \right\|_\infty < 1, \tag{2.8}$$

holds for all $\Delta \in \tilde{\mathbf{\Delta}}_1$, and $\left\| \Delta_P \right\|_\infty \leq 1$.

Proof. This lemma is based on the small gain theorem. In the small gain theorem framework, the robust performance of the uncertain plant (2.1) is equivalent to robust stability of a modified plant where a full block complex matrix Δ_P is added as a feedback gain on the performance channel of the original uncertain plant. Hence, the robust performance requirement in

Figure 2.3 is

$$\|\mathcal{F}(K, \Delta_{\text{nom}})\Delta_{\text{aug}}\|_\infty < 1, \qquad \forall \|\Delta_P\|_\infty \leq 1, \tag{2.9}$$

Due to the inequality $\|AB\|_\infty \leq \|A\|_\infty \|B\|_\infty$, a conservative form of the robust performance criterion (2.9) is given by ([92])

$$\|\Delta_{\text{aug}}\|_\infty < \frac{1}{\|\mathcal{F}(K, \Delta_{\text{nom}})\|_\infty}, \qquad \forall \|\Delta_P\|_\infty \leq 1, \tag{2.10}$$

Now, supposing that $\mathcal{F}(K_1, \Delta_1)$ and $\mathcal{F}(K_2, \Delta_2)$ satisfy robust performance criterion (2.10) for all $\Delta \in \tilde{\mathbf{\Delta}}_1$, and $\Delta \in \tilde{\mathbf{\Delta}}_2$, respectively, we would like to show that $\tilde{\mathbf{\Delta}}_2 \subseteq \tilde{\mathbf{\Delta}}_1$.

Based on the condition (2.6), there exists a $k > 1$ such that the following holds:

$$\|\mathcal{F}(K_2, \Delta_2)\|_\infty = k \|\mathcal{F}(K_1, \Delta_1)\|_\infty . \tag{2.11}$$

Replacing $\|\mathcal{F}(K_2, \Delta_2)\|_\infty$ in (2.7) with the above equation,

$$\|\mathcal{F}(K_1, \Delta_1)\|_\infty \|k\Delta_{\text{aug}}\|_\infty < 1, \tag{2.12}$$

holds for all $\Delta \in \tilde{\mathbf{\Delta}}_2$. This implies:

$$\|\mathcal{F}(K_1, \Delta_1)\|_\infty \|\Delta_{\text{aug}}\|_\infty < 1, \tag{2.13}$$

for all $\frac{\Delta}{k} \in \tilde{\mathbf{\Delta}}_2$. Since $k > 1$, the above inequality means that the uncertainty set for which closed-loop system $\mathcal{F}(K_1, \Delta_1)$ satisfies the robust performance criterion includes $\tilde{\mathbf{\Delta}}_2$. □

Equation (2.9) guarantees the robust performance objective $\gamma = 1$ is met for the closed-loop system. If a different robust performance objective γ is desired, it can be included in N by scaling the performance channel z. In the remaining part of the chapter we assume that the robust performance objective γ is already included in the system matrix P.

The problem (2.5) is a nonsmooth optimization due to the nonsmoothness of the H_∞ norm. In Section 2.4, we will develop a gradient based numerical algorithm to find a local optimal solution to this problem.

2.3.2 Finding the uncertainty subset $\mathbf{\Delta}^{(i)}$

Suppose that the optimal nominal uncertainty value Δ_{nom} has been determined in (2.5), and that we would like to find the largest uncertainty subset $\mathbf{\Delta}^{(i)}$ around this nominal value which meets the robust performance cost γ (i.e. $\left\|T_{w\to z}(K^{(i)},\Delta)\right\|_\infty < \gamma$ for all $\Delta \in \mathbf{\Delta}^{(i)}$). The nominal closed-loop system must satisfy the desired performance objective; otherwise, the perturbed plant, regardless of its uncertainty subset, can not meet this performance objective.

An iterative method is used to find the largest uncertainty set which guarantees the robust performance γ. In each iteration we expand the uncertainty subset by a small amount and check the robust performance criterion (2.9). These iterations are continued until either the criterion (2.9) is violated or the uncertainty subset $\mathbf{\Delta}^{(i)}$ reaches the boundaries of the original uncertainty set. Lemma 2.3.3 provides an initial uncertainty subset which can be used in this iterative method.

Lemma 2.3.3. *In Figure 2.1, suppose that the plant $N := F_l(P,K)$ satisfies the desired performance γ, and the uncertainty matrix is*

$$\Delta = \begin{bmatrix} \delta_1 M_1 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & \delta_m M_m \end{bmatrix}, \tag{2.14}$$

where M_i is a complex matrix, and

$$|\delta_i| < \delta_{i,bnd} := \frac{1}{\overline{\sigma}(M_i)\,\|N\|_\infty} \quad , \quad i \in \{1,\ldots,m\}. \tag{2.15}$$

Then the closed loop system is robustly stable.

Proof. The maximum singular value of the uncertainty matrix is

$$\overline{\sigma}(\Delta) = \max\left\{\delta_{1,bnd}\overline{\sigma}\left(M_1\right), \cdots, \delta_{m,bnd}\overline{\sigma}\left(M_m\right)\right\}.$$

Substituting the $\delta_{i,bnd}$ from (2.15) into the above equation, we can obtain

$$\overline{\sigma}(\Delta) < \frac{1}{\|N\|_\infty}. \tag{2.16}$$

This equation is the robust performance criterion (2.10). Hence, the plant N satisfies the robust performance criterion for any uncertainty value bounded by (2.15). □

2.3.3 Partition modification

In previous sections, we discussed the problem of finding the optimal nominal plant and its corresponding largest admissible uncertainty subset $\mathbf{\Delta}^{(i)}$. This uncertainty subset establishes a new subset in the uncertainty set. If we do not modify the entire partition after finding this subset, we may end up with small gaps between the subsets after a number of iterations of Algorithm 2.3.1. Figure 2.4 shows a possible situation that may happen in the absence of the partition modification in Algorithm 2.3.1 for a system with a two dimensional uncertainty set. If we consider these small gaps as individual subsets, the number of subsets will grow significantly, which is undesirable. In this section, our objective is to introduce the partition modification step to avoid this situation.

Suppose that we are in the r^{th} iteration of the MRC design algorithm (Algorithm 2.3.1), and that the uncertainty set has m dimensions, named $\delta_1, \ldots, \delta_m$. Let the following uncertainty subset be the newly established

uncertainty subset which is found as explained in Section 2.3.2,

$$\mathbf{\Delta}^{(i)} = \{[\delta_{1,min}, \delta_{1,max}], \cdots, [\delta_{m,min}, \delta_{m,max}]\}. \tag{2.17}$$

Let $\mathbf{\Delta}_{\text{part}} = \{\mathbf{\Delta}^{(q)}, q = 1, \ldots, r\}$ denote the set of previously established partitions which satisfy the desired robust performance objective, and $\mathbf{\Delta}_{\text{unpart}} = \{\mathbf{\Delta}^{(q)}_{\text{unpart}}, q = 1, \ldots, s\}$ is the set of remaining partitions in the original uncertainty set $\mathbf{\Delta}$. The uncertainty set $\mathbf{\Delta}$ is the union of the elements of these two sets, i.e.

$$\mathbf{\Delta} = \{\cup_{q=1}^{r} \mathbf{\Delta}^{(q)}\} \bigcup \{\cup_{q=1}^{s} \mathbf{\Delta}^{(q)}_{\text{unpart}}\}.$$

Without loss of generality assume $\mathbf{\Delta}^{(i)} \subset \mathbf{\Delta}^{(1)}_{\text{unpart}}$.

The following algorithm provides a systematic method to redefine the partitions such that $\mathbf{\Delta}^{(i)}$ becomes a member of the established partitions $\mathbf{\Delta}_{\text{part}}$ and the set $\mathbf{\Delta}_{\text{unpart}}$ is modified in a way to avoid undesirable situations, as shown in Figure 2.4.

Algorithm 2.3.4. *Set* $l = 0$.

(I) Set $l = l + 1$. *Divides the* $\mathbf{\Delta}^{(1)}_{\text{unpart}}$ *into following three subsets (see*

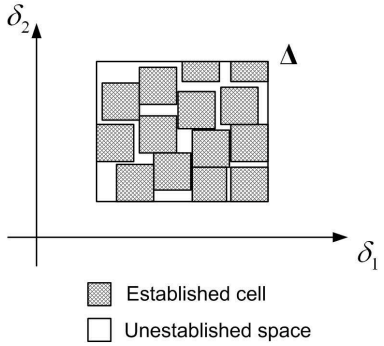


Figure 2.4: Undesirable situations that may occur in the absence of partition modification.

Figure 2.5):

$$\mathbb{D}_1 := \left\{(\delta_1, \cdots, \delta_m) \in \Delta^{(1)}_{\text{unpart}} : \delta_l \leq \delta_{l,min}\right\}, \tag{2.18a}$$

$$\mathbb{D}_2 := \left\{(\delta_1, \cdots, \delta_m) \in \Delta^{(1)}_{\text{unpart}} : \delta_{l,min} \leq \delta_l \leq \delta_{l,max}\right\}, \tag{2.18b}$$

$$\mathbb{D}_3 := \left\{(\delta_1, \cdots, \delta_m) \in \Delta^{(1)}_{\text{unpart}} : \delta_l \geq \delta_{l,max}\right\}. \tag{2.18c}$$

Note that $\mathbf{\Delta}^{(i)} \subseteq \mathbb{D}_2$.

(II) Replace the $\mathbf{\Delta}^{(1)}_{\text{unpart}}$ *with* $\mathbb{D}_2$ *and add the other two subsets (i.e.* $\mathbb{D}_1$ *and* $\mathbb{D}_3$*) to the unestablished partition set, i.e.*

$$\mathbf{\Delta}_{\text{unpart}} = \left\{\mathbb{D}_2, \mathbf{\Delta}^{(2)}_{\text{unpart}}, \ldots, \mathbf{\Delta}^{(s+2l-2)}_{\text{unpart}}, \mathbb{D}_1, \mathbb{D}_3\right\}. \tag{2.19}$$

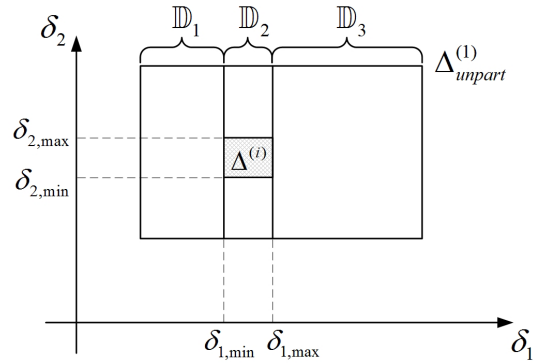
(III) If $l < m$ *go to Step (I). Otherwise, remove the first element of the* $\mathbf{\Delta}_{\text{unpart}}$ *set and add it to the established partitions set,*

$$\mathbf{\Delta}_{\text{unpart}} = \left\{\mathbf{\Delta}^{(q)}_{\text{unpart}}, q = 2, \ldots, s + 2l\right\}, \tag{2.20}$$

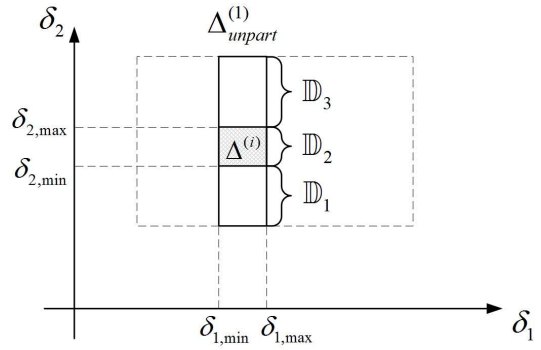
$$\mathbf{\Delta}_{\text{part}} = \left\{\mathbf{\Delta}^{(1)}, \ldots, \mathbf{\Delta}^{(r)}, \mathbf{\Delta}^{(1)}_{\text{unpart}}\right\}. \tag{2.21}$$

Terminate the algorithm.

Figure 2.5 depicts the result of Algorithm 2.3.4 for a two dimensional uncertainty set. Since in this example the uncertainty set is two dimensional ($m = 2$), this algorithm has two iterations. It should be noted that in the first iteration (Figure 2.5(a)) and the second iteration (Figure 2.5(b)) $\mathbf{\Delta}^{(i)}$ is a subset of the set $\mathbb{D}_2$. Moreover in the last iteration of the algorithm ($l = 2$ in this example) $\mathbb{D}_2 = \mathbf{\Delta}^{(i)}$. At the end of the algorithm, for this particular example, four uncertainty sets are added to the $\mathbf{\Delta}_{\text{unpart}}$ set and $\mathbf{\Delta}^{(i)}$ is removed from the $\mathbf{\Delta}_{\text{unpart}}$ set and added to the $\mathbf{\Delta}_{\text{part}}$ set.



(a) Algorithm 2.3.4 - $l = 1$.



(b) Algorithm 2.3.4 - $l = 2$.

Figure 2.5: Graphical interpretation of the result of Algorithm 2.3.4 for a two dimensional uncertainty set.

2.4 Nonsmooth H_∞ algorithm for finding the optimal plant and controller

This section explains a nonsmooth first order descent method, which can be used to solve the optimization problem (2.5). The cost function of the optimization problem (2.5) is the H_∞ norm of the closed-loop nominal plant. This cost function is a nonsmooth function for two reasons. Firstly, the H_∞ norm contains a maximum eigenvalue function which is not smooth because this function may switch between different eigenvalues. Secondly, this maximum eigenvalue function should be obtained over all frequencies, hence the change in the frequency at which the H_∞ norm is attained induces nons-

moothness in the cost function of the optimization problem (2.5). Therefore, it is not always possible to define the gradient of this cost function which is necessary for traditional methods developed for smooth functions.

The Clarke subdifferential [31] is a powerful tool to compute the gradient of a nonsmooth function. Nonsmooth functions can have several gradients at one point where smooth functions have only one gradient at any point. In the nominal nonsmooth H_∞ synthesis method, proposed in [9], the Clarke subdifferential is used to determine the set of descent directions of a cost function, which contains the H_∞ norm of a transfer function. Since this method does not use Lyapunov variables, it is faster than the traditional H_∞ synthesis methods which use Lyapunov variables.

To utilize advantages of the nominal nonsmooth H_∞ synthesis method [9], we reduce the optimization problem (2.5) to a structured controller synthesis problem. After this reduction, we use an algorithm based on nominal nonsmooth H_∞ synthesis to solve the reduced form of the optimization problem (2.5).

2.4.1 Reduction to H_∞ structured controller synthesis

The optimization problem (2.5) is similar to the nominal H_∞ controller synthesis problem, discussed in [9], because both problems seek to optimize an H_∞ norm. The difference between these two problems is that, in (2.5), the controller parameter K and nominal uncertainty value Δ are the optimization variables, while in the nominal H_∞ controller synthesis problem, the controller parameter K is the only optimization variable.

Therefore, we extend the nonsmooth H_∞ controller synthesis method, proposed in [9], to include the nominal plant optimization. We group the controller and the uncertainty matrix in one block as shown in Figure 2.6. Assume the state space representation of the nominal plant P in Figure 2.1

is given by

$$\begin{bmatrix} \dot{x}(t) \\ y_\Delta(t) \\ z(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} A & B_1 & B_2 & B_3 \\ C_1 & D_{11} & D_{12} & D_{13} \\ C_2 & D_{21} & D_{22} & D_{23} \\ C_3 & D_{31} & D_{32} & D_{33} \end{bmatrix} \begin{bmatrix} x(t) \\ u_\Delta(t) \\ w(t) \\ u(t) \end{bmatrix}. \tag{2.22}$$

In the configuration of Figure 2.6, the first and second input/output are flipped comparing to Figure 2.1. Hence the new system matrix $\tilde{P}$ is

$$\tilde{P}(s) = \left[\begin{array}{c|ccc} A & B_2 & B_1 & B_3 \\ \hline C_2 & D_{22} & D_{21} & D_{23} \\ C_1 & D_{12} & D_{11} & D_{13} \\ C_3 & D_{32} & D_{31} & D_{33} \end{array}\right]. \tag{2.23}$$

For simplicity we assume $D_{33} = 0$. Let $T_{w\to z}(K, \Delta)$ be the closed-loop transfer function of the performance channel, defined by

$$T_{w\to z}(K, \Delta) := \left[\begin{array}{c|c} \mathcal{A}(K, \Delta) & \mathcal{B}(K, \Delta) \\ \hline \mathcal{C}(K, \Delta) & \mathcal{D}(K, \Delta) \end{array}\right]. \tag{2.24}$$

For the case of a static controller (i.e. $u = Ky$) the closed-loop state space

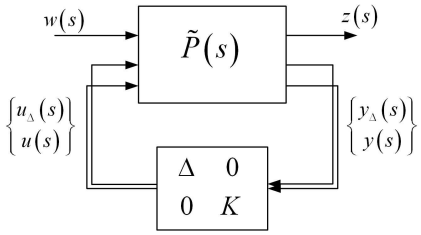


Figure 2.6: New configuration of the generalized plant.

matrices are given by

$$\begin{aligned}
\mathcal{A}(K,\Delta) &:= A + \begin{bmatrix} B_1 & B_3 \end{bmatrix} \begin{bmatrix} \Delta & 0 \\ 0 & K \end{bmatrix} \begin{bmatrix} C_1 \\ C_3 \end{bmatrix}, \\
\mathcal{B}(K,\Delta) &:= B_2 + \begin{bmatrix} B_1 & B_3 \end{bmatrix} \begin{bmatrix} \Delta & 0 \\ 0 & K \end{bmatrix} \begin{bmatrix} D_{12} \\ D_{32} \end{bmatrix}, \\
\mathcal{C}(K,\Delta) &:= C_2 + \begin{bmatrix} D_{21} & D_{23} \end{bmatrix} \begin{bmatrix} \Delta & 0 \\ 0 & K \end{bmatrix} \begin{bmatrix} C_1 \\ C_3 \end{bmatrix}, \\
\mathcal{D}(K,\Delta) &:= D_{22} + \begin{bmatrix} D_{21} & D_{23} \end{bmatrix} \begin{bmatrix} \Delta & 0 \\ 0 & K \end{bmatrix} \begin{bmatrix} D_{12} \\ D_{32} \end{bmatrix}. \qquad (2.25)
\end{aligned}$$

For a dynamic controller the closed-loop transfer function is again obtained via formula (2.24) and (2.25) by performing the substitutions:

$$\begin{aligned}
K &\to \begin{bmatrix} A_K & B_K \\ C_K & D_K \end{bmatrix}, & A &\to \begin{bmatrix} A & 0 \\ 0 & 0_K \end{bmatrix}, \\
B_1 &\to \begin{bmatrix} B_1 \\ 0 \end{bmatrix}, & B_2 &\to \begin{bmatrix} 0 & B_2 \\ I_K & 0 \end{bmatrix}, \\
C_1 &\to \begin{bmatrix} C_1 & 0 \end{bmatrix}, & C_2 &\to \begin{bmatrix} 0 & I_K \\ C_2 & 0 \end{bmatrix}, \\
D_{21} &\to \begin{bmatrix} 0 \\ D_{21} \end{bmatrix}, & D_{12} &\to \begin{bmatrix} 0 & D_{12} \end{bmatrix}. \qquad (2.26)
\end{aligned}$$

Grouping the controller and uncertainty blocks as a single block, the optimization problem (2.10) becomes similar to the nominal H_∞ synthesis. The only difference between this optimization problem and the nominal H_∞ synthesis is that in this problem, the optimization variables, which correspond to zero elements in the uncertainty block, should be forced to remain zero during optimization. Hence this optimization problem can be regarded as a *structured controller design problem.*

2.4.2 Numerical algorithm for finding the optimal nominal plant and controller

The following algorithm is an extension of the nominal nonsmooth H_∞ synthesis algorithm proposed in [9], since it reduces to the nominal H_∞ synthesis algorithm if we remove the uncertainty matrix from all equations. The proof of convergence of Algorithm 2.4.1 is similar to the nominal H_∞ synthesis algorithm in [9], and hence it is omitted in this thesis.

Algorithm 2.4.1. *Consider the system matrix $\tilde{P}$ in (2.23) and closed-loop state matrices in (2.25).*

1. ***Choose an initial nominal plant and a stabilizing controller:*** *The algorithm must be initialized with a stabilizing controller. A nonsmooth multidirectional search algorithm is introduced in [6], which can be used to design the initial stabilizing controller. Choose midpoints of the uncertainty ranges as the initial nominal plant and design a stabilizing controller for this nominal plant.*

2. ***Find a descent direction:*** *Solve the following optimization problem:*

$$\begin{aligned}\theta(x) = \sup_{\tau_i, \sum_i \tau_i = 1} \sup_{Y_i \in \mathbb{B}} \Bigg\{ & - \|T_{w\to z}(K,\Delta)\|_\infty^2 + \sum_{i=1}^{r} \tau_i \mu_i - \frac{1}{2\sigma} \\ & \Bigg\| \sum_{i=1}^{r} \tau_i \, \|T_{w\to z}(K,\Delta)\|_\infty^{-1} \\ & \times \mathrm{Re}\Big(G_{21}(K,\Delta,j\omega) T_{w\to z}^H(K,\Delta,j\omega) Q_i Y_i Q_i^H G_{12}(K,\Delta,j\omega) \Big)^T \Bigg\|^2 \Bigg\}. \end{aligned} \tag{2.27}$$

   In the above equation,

   - *the vector x consists of parameters in the controller and the uncertainty matrices,*

- $\mathbb{B} := \{X \in \mathbb{S} : \mathrm{Tr}(X) = 1\}$ *where* $\mathbb{S}$ *is the set of symmetric positive semi-definite matrices with the appropriate dimension,*
- $\mu_1 > \mu_2 > \cdots > \mu_r$ *are the eigenvalues of* $T_{w\to z}(K, \Delta, j\omega)$ *without repetition,*
- ω *is the frequency at which* $\|T_{w\to z}(K, \Delta)\|_\infty$ *is attained,*
- $\sigma > 0$ *is a constant,*
- *the columns of* Q *form an orthonormal basis for the eigenvalues of* $T_{w\to z}(K, \Delta, j\omega)\ T_{w\to z}(K, \Delta, j\omega)^H$*, and*
- *transfer functions* $G_{12}(K, \Delta, s)$ *and* $G_{21}(K, \Delta, s)$ *are defined as*

$$\begin{aligned} G_{12}(K, \Delta, s) :=& \mathcal{C}(K, \Delta)(sI - \mathcal{A}(K, \Delta))^{-1}) \begin{bmatrix} B_1 & B_3 \end{bmatrix} \\ &+ \begin{bmatrix} D_{21} & D_{23} \end{bmatrix}, \\ G_{21}(K, \Delta, s) :=& \begin{bmatrix} C_1 \\ C_3 \end{bmatrix} (sI - \mathcal{A}(K, \Delta))^{-1})\mathcal{B}(K, \Delta) + \begin{bmatrix} D_{12} \\ D_{32} \end{bmatrix}. \end{aligned}$$

Having the optimal Y_i *and* τ_i *from the solution of (2.27), the steepest descent direction is given by*

$$\begin{aligned} g(x) := -\frac{1}{\sigma} \sum_{i=1}^{r} \Bigg(& \tau_i \|T_{w\to z}(K, \Delta)\|_\infty^{-1} \\ & \times \mathrm{Re}\Big(G_{21}(K, \Delta, j\omega) T_{w\to z}^H(K, \Delta, j\omega) Q_i Y_i Q_i^H G_{12}(K, \Delta, j\omega)\Big)^T \Bigg). \end{aligned} \tag{2.28}$$

The matrix $g(x)$ *contains the descent direction of parameters in both the controller and the uncertainty.*

When the uncertainty matrix has a special structure, we constraint the change in the uncertainty matrix by multiplying a pattern matrix by its

steepest descent direction;

$$\begin{bmatrix} g_\Delta & \\ & g_K \end{bmatrix} = \begin{bmatrix} W_\Delta & 0 \\ 0 & \mathbf{1}_K \end{bmatrix} \bullet g, \tag{2.29}$$

where $\bullet$ *denotes entrywise matrix product [55], and "*$\mathbf{1}_K$*" denotes the unity matrix whose size is the same as that of* K*. The elements of the pattern matrix* W_Δ *are either 0 or 1, where* $W_\Delta(i,j) = 0$ *means that the uncertainty matrix element* $\Delta(i,j)$ *must be zero, and* $W_\Delta(i,j) = 1$ *means that uncertainty matrix element* $\Delta(i,j)$ *can change during optimization.*

3. ***Line search:*** *Perform a line search to find the optimal amount of movement in the steepest descent directions* g_K *and* g_Δ*.*

$$t(x) := \arg\max_t \{\|T_{w\to z}(K,\Delta)\|_\infty - \|T_{w\to z}(K + tg_K, \Delta + tg_\Delta)\|_\infty\} \tag{2.30}$$

4. *Iterate Steps (2) and (3) until one of the following termination criteria is satisfied. Since this algorithm is first order, it is slow in the neighborhood of the locally optimal solution. Hence various stopping criteria should be implemented to avoid unnecessary computations and to ensure that sufficient approximation of the solution is found.*

   - ***Critical point:*** *The following condition is satisfied when the descent directions are near horizontal. As in calculus, this is a property of a critical point.*

$$\begin{aligned} \|g_K\| &< \epsilon_1, \\ \|g_\Delta\| &< \epsilon_2. \end{aligned} \tag{2.31}$$

   - ***Small change in the goal function:*** *The following condition is satisfied when the* H_∞ *norm does not improve significantly in*

two subsequent iterations.

$$\begin{aligned}&\|T_{w\to z}(K,\Delta)\|_\infty - \|T_{w\to z}(K+tg_K,\Delta+tg_\Delta)\|_\infty \\ &< \epsilon_3\left(1+\|T_{w\to z}(K,\Delta)\|_\infty\right). \end{aligned} \tag{2.32}$$

- ***Small change in optimization variables:*** *The following condition is satisfied when the controller and uncertainty matrices do not change significantly in two subsequent iterations.*

$$\begin{aligned}\|t(x)g_K\| &< \epsilon_4\left(1+\|K\|\right), \\ \|t(x)g_\Delta\| &< \epsilon_5\left(1+\|\Delta\|\right). \end{aligned} \tag{2.33}$$

Remark. *In Step (2) of Algorithm 2.4.1, the descent direction of the uncertainty matrix (i.e. g_Δ) is constrained by multiplying a pattern matrix W_Δ by the original descent direction g. We do this multiplication to make sure that the nominal uncertainty moves in the appropriate direction which is consistent with the structure of the uncertainty matrix.*

Remark. *In the Step (3) of Algorithm 2.4.1, a line search is performed with one variable t to find the optimum amount of movement in the descent directions g_K and g_Δ. In practical problems, the controller parameters K and the uncertainty values Δ can be in different orders of magnitude. Therefore, it is better to introduce two different variables t_1 and t_2 in Step (3) of Algorithm 2.4.1. In this case, the following two dimensional optimization problem must be solved for t_1 and t_2 instead of (2.30):*

$$\begin{aligned}(t_1(x),t_2(x)) = \arg\max_{t_1,t_2}\big\{\ &\|T_{w\to z}(K,\Delta)\|_\infty \\ &- \|T_{w\to z}(K+t_1g_K,\Delta+t_2g_\Delta)\|_\infty\ \big\}. \end{aligned} \tag{2.34}$$

This allows the numerical solution to move different amounts in the controller parameter space and the uncertainty space, in each step. Our numeri-

cal experiments show that this modification increases the convergence rate of Algorithm 2.4.1.

2.5 Multiple robust H_∞ track-following controller synthesis in dual-stage hard disk drives (HDDs)

This section applies the proposed MRC synthesis method to the track-following controller design in HDDs. Several controllers have been proposed for the track-following purpose in HDDs over recent years, see, for instance, [33, 57, 69–71, 82]. The track-following controller should be able to perform the track-following task for all HDD products in any plausible working condition. Although track-following robustness is important, its precision should also be improved to meet the growing demand on disks with larger capacities. Since these objectives are incompatible, the MRC method can be one of the proper methods to overcome this challenge.

2.5.1 Generalized plant of a dual-stage HDD

Figure 2.7 shows the block diagram of a dual-stage positioning servo mechanism. Several positioning mechanisms have been designed for dual-stage HDDs, see, e.g., [27, 42, 58, 77]. In this work, the positioning mechanism consists of the voice coil motor (VCM) as the main actuator, and a translational microactuator (MA) as the secondary actuator for fine tuning. The control input to VCM and MA are denoted by u_v and u_m respectively. The air flow induced force w_v assumed to be normalized Gaussian white disturbance to VCM. The relative motion of the tracks on the disk relative to head w_r is normalized Gaussian white noise, caused by external shocks, mechanical imperfections, D/A quantization noise, and power amplifier noise. Measured output signals y_p and y_m are respectively the output of strain gauge mounted on the suspension, and the relative position of the microactuator to

the suspension tip. Measured output y_{PES} is the position error signal (PES), defined as the position of the read/write head with respect to the reference track.

Although there are some bias forces and nonlinearities in the HDD system, it is known that this system can be modeled as a linear system for track following controller design purpose [26]. The transfer functions for the VCM dynamics, G_V, the MA dynamics, G_{MA}, the coupling between the VCM and MA dynamics, G_C, and the track runout model, W_R, are taken from [30] and

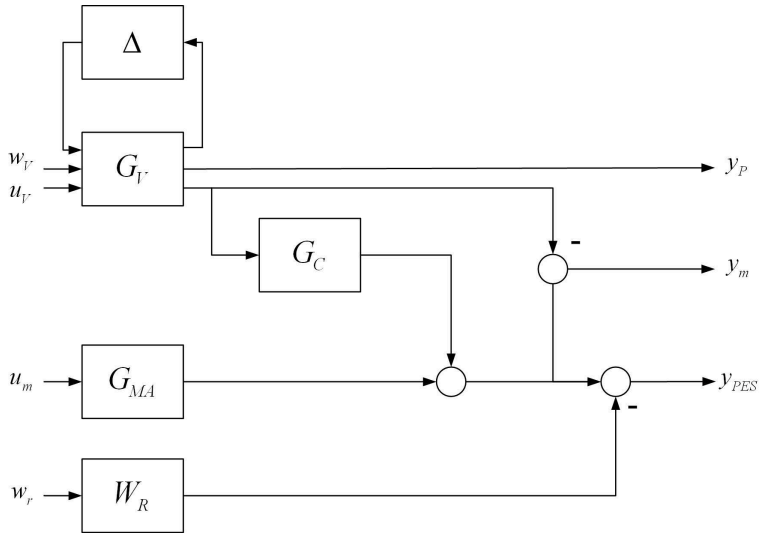


Figure 2.7: The block diagram of a dual-stage HDD servo mechanism.

Table 2.1: Nominal parameter values for dual-stage HDD servo mechanism.

q	mode description	ζ_q	ω_q	A_q
1	rigid body mode	0.5	376.99	$\begin{bmatrix} 0 & 0 \\ 0 & 1.4 \times 10^4 \end{bmatrix}$
2	butterfly mode	0.015	4.6496×10^4	$\begin{bmatrix} -0.439 & 0.48 \\ 0.768 & -0.84 \end{bmatrix}$
3	sway mode	0.015	6.7218×10^4	$\begin{bmatrix} -0.783 & 0.408 \\ 0.576 & -0.3 \end{bmatrix}$
m	MA	0.2	1.4137×10^4	0.2

respectively given by

$$G_V(s) := \sum_{i=1}^{3} \frac{A_i \omega_i^2}{s^2 + 2\zeta_i \omega_i s + \omega_i^2}, \tag{2.35a}$$

$$G_{MA}(s) := \frac{A_m \omega_m^2}{s^2 + 2\zeta_m \omega_m s + \omega_m^2}, \tag{2.35b}$$

$$G_C(s) := \frac{2\zeta_m \omega_m s + \omega_m^2}{s^2 + 2\zeta_m \omega_m s + \omega_m^2}, \tag{2.35c}$$

$$W_R(s) := \frac{1.2 \times 10^5 s^2 + 2.89 \times 10^9 s + 5.298 \times 10^{12}}{s^3 + 2684 s^2 + 1.756 \times 10^6 s + 4.703 \times 10^8}, \tag{2.35d}$$

where realistic parameters are selected as in Table 2.1 [30].

Variations in the natural frequency and damping ratio of flexible modes of the VCM result in parametric uncertainty in the VCM transfer function. Uncertainty in VCM dynamics are represented by the uncertainty block Δ in Figure 2.7. In our system, the values of ζ_2 and ζ_3, and ω_2 and ω_3 in Table 2.1 are allowed to vary up to $\pm 50\%$, and $\pm 10\%$, respectively. It is assumed that the variations in ζ_2 and ζ_3 are correlated, and so are those in ω_2 and ω_3. Therefore, the Δ block in Figure 2.7 contains parametric uncertainty; δ_1, and δ_2 stand for the variations in ζ_2 and ζ_3, and ω_2 and ω_3, respectively.

The inputs and outputs of the 11^{th} order generalized plant for dual-stage

HDD, shown in Figure 2.1, are chosen as

$$w := \begin{bmatrix} w_r \\ w_v \end{bmatrix}, \qquad u := \begin{bmatrix} u_v \\ u_m \end{bmatrix}, \qquad z := \begin{bmatrix} y_{PES} \\ u_v \\ 0.01u_m \end{bmatrix}, \qquad y := \begin{bmatrix} y_{PES} \\ y_m \\ y_p \end{bmatrix}. \tag{2.36}$$

2.5.2 MRC synthesis

For the dual-stage HDD, using nonsmooth controller synthesis method and μ-analysis, we found that the achievable H_∞ robust performance is $\gamma = 257.6$ if a single robust controller is used for the entire uncertainty region. Our algorithm has a potential for achieving a smaller robust performance. Here we chose $\gamma = 80$ which provides satisfactory performance. We also selected ϵ_i constants in our nonsmooth numerical algorithm as

$$\epsilon_1 = 1 \times 10^{-6}, \qquad \epsilon_2 = 1 \times 10^{-8}, \qquad \epsilon_3 = 1 \times 10^{-6},$$

$$\epsilon_4 = 1 \times 10^{-3}, \qquad \epsilon_5 = 1 \times 10^{-7},$$

that provide a satisfactory convergence speed and solution tolerance.

The overall optimization took about 4,200 seconds using a high-end personal computer with a 2.4 GHz Intel® Core™ 2 Duo Processor. In this example, to satisfy the robust performance objective $\gamma = 80$, 17 uncertainty subsets are sufficient. Figure 2.8 shows the partitions generated by our algorithm. Figure 2.9 shows that the desired robust H_∞ norm $\gamma = 80$ is satisfied over the entire uncertainty set. The designed dynamic output-feedback controllers are of 11^{th} order.

Figure 2.10 shows the Bode diagram of the closed-loop sensitivity transfer functions from r to Y_{PES} for 20 randomly sampled uncertainty values in all subsets. The bandwidths of the closed-loop systems for all subsets are larger than 5×10^3 (rad/sec) which is satisfactory for this system [1]. Figure 2.11

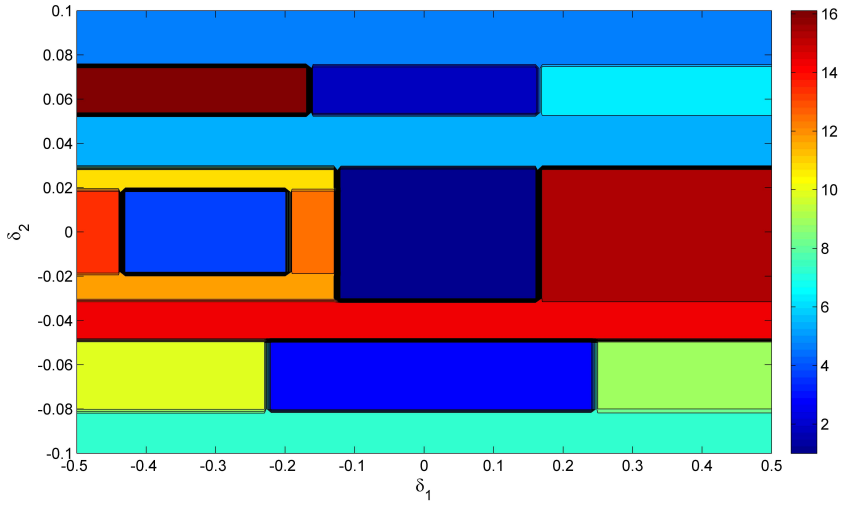


Figure 2.8: Resulting partitions for $\gamma = 80$.

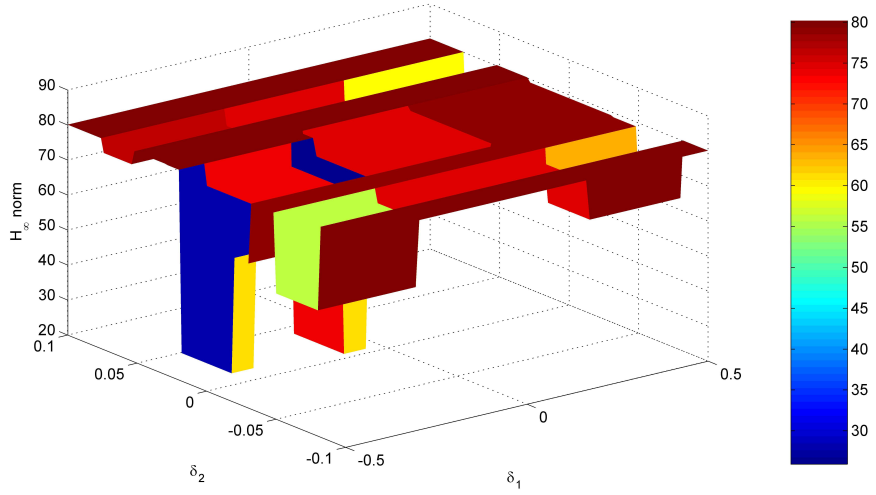


Figure 2.9: H_∞ norm over the entire uncertainty set.

shows the Bode diagram of closed-loop sensitivity transfer functions from r to Y_{PES} for 20 randomly sampled uncertainty values for the single robust controller case. It can be seen that if we use a single controller for all subsets,

the gains of sensitivity functions will have larger peaks.

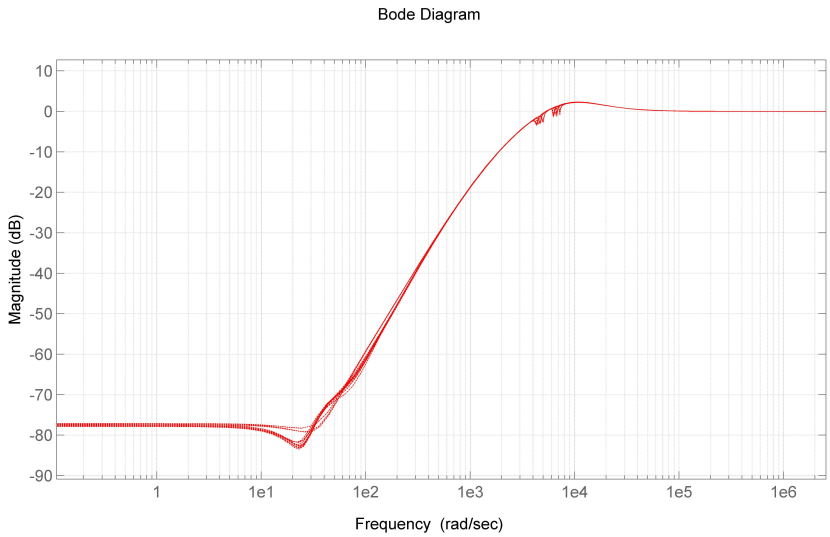


Figure 2.10: Closed-loop sensitivity transfer functions from r to y_{PES} for the MRC approach.

Although our method may result in a larger number of uncertainty subsets relative to the method of [30] for a similar robust performance, our method determines the sufficient number of subsets to achieve a specified robust performance γ *automatically*. This is a significant advantage of our method because in the method of [30], the number of divisions in each uncertainty dimension must be pre-specified by the user. To determine such appropriate number of divisions is not a trivial task, and needs to be found by trial and error.

This together with the use of nonsmooth optimization make our method computationally fast relative to the method of [30]. It should be emphasized that, although controller design is an off-line task, its computation time is of importance for control engineers. Specially, in the H_∞ controller synthesis, large computation time would make the search of appropriate weighting functions cumbersome.

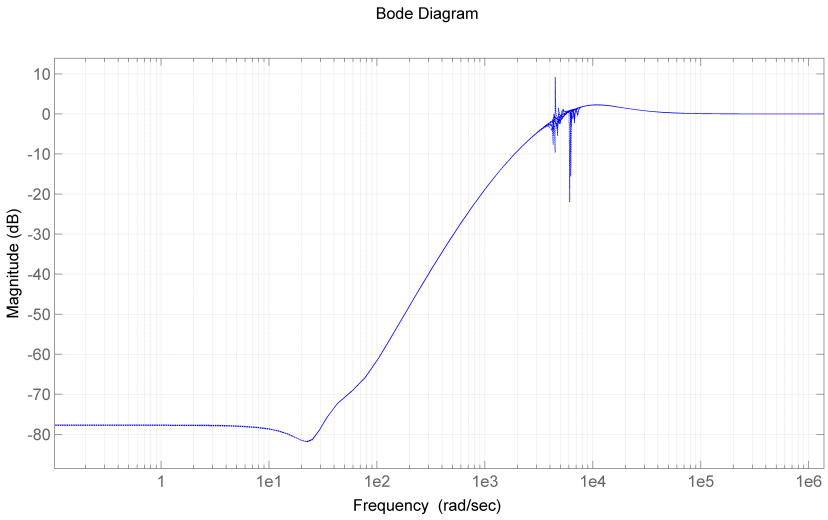


Figure 2.11: Closed-loop sensitivity transfer functions from r to y_{PES} for the single robust control approach.

2.6 Conclusions

In this chapter, we proposed a new systematic method that meets Objective (O1-i) of Section 1.3, i.e. synthesizes the MRC in a computationally efficient manner. Our algorithm achieves a pre-specified desired robust performance objective, by dividing the uncertainty set to several subsets, and by designing a robust controller to each subset. The method can design a fixed order dynamic output-feedback controller. In terms of computation time, the method has two advantages over the previous MRC method proposed in [30]:

- In the method of [30], the computation time grows significantly with the number of partitions compared to our method. This is because our method does not modify the uncertainty subsets once they have been established, while the method in [30] needs to redesign the subsets iteratively.

- In the method of [30], because of the use of Lyapunov variables, the

computation time grows quadratically with the number of states of the closed-loop system, while in our method based on nonsmooth optimization, the computation time does not change significantly for high order systems.

One practical control example has been provided to show the performance improvement with the MRC method. Since the proposed method is based on the nonsmooth H_∞ synthesis method, it can make use of advantages inherent to the nonsmooth method, such as multiband controller synthesis [11], multidisc H_∞ [10], mixed H_2/H_∞ controller design [12].

Chapter 3

Track-following Servo Control in Hard Disk Drives [1]

3.1 Introduction

hard disk drives (HDDs) played a central role in the development of recent information technology, due to their low cost, fast data access speed, and large capacity. In HDDs, a magnetic inductive head reads or writes binary bits on disks which are coated with permanent magnet. The read/write head (R/W head) is attached to a suspension arm and flies over the spinning disk, as shown in Figure 3.1. In conventional HDDs, a VCM is used to position the R/W head.

The R/W head positioning servomechanism has two modes; track following and track seeking. In the track seeking mode, the R/W head is moved from the present track to a desired destination track in minimum time using a bounded control effort. In the track following mode, the R/W head

[1] This chapter is based on the following articles: E. Azadi Yazdi, M. Sepasi, F. Sassani and R. Nagamune, *"Automated multiple robust track-following control system design in hard disk drives"*, 2010 ASME Dynamic Systems and Control Conference, Boston, MA, and E. Azadi Yazdi, M. Sepasi, F. Sassani and R. Nagamune, *"Automated multiple robust track-following control system design in hard disk drives"*, to appear in IEEE Transactions on Control System Technology.

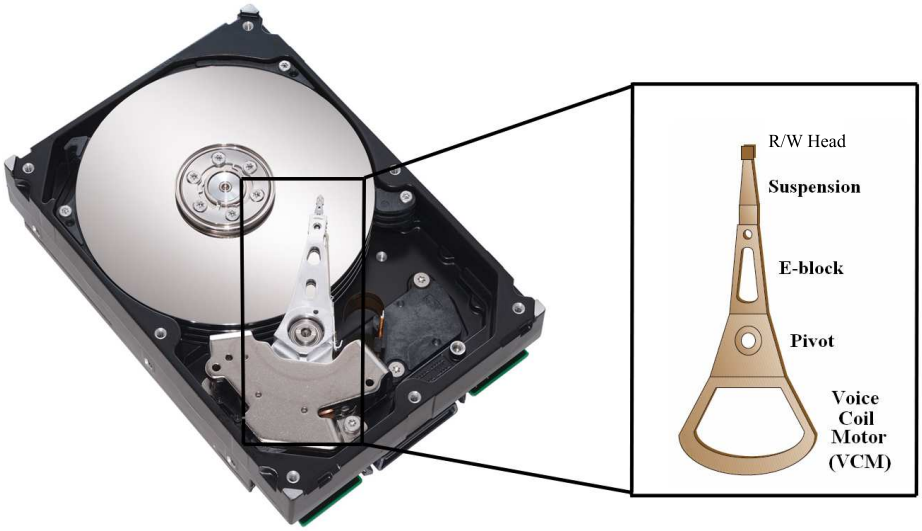


Figure 3.1: HDD servo mechanism.

should be maintained as close as possible to the destination track center while reading or writing information to the disk. The required accuracy in the track following mode depends on the distance between two neighboring data tracks, called track width.

The track width is continuously being reduced in order to meet increasing demands on disks with larger capacities. This trend has to be followed by improving the positioning precision of the R/W head. To improve the tracking precision, the R/W head positioning mechanism must be modified on both hardware and software sides. Recent developments in servomechanism hardware make use of a secondary MEMS or piezo actuator and auxiliary sensor [91, 110].

Parallel to hardware developments, on the software side, the servo controller must be modified. The HDD servo controller must have two main properties. Firstly, the controller should be designed to achieve high performances. Secondly, the controller must be designed to perform track seeking and track following tasks for all HDD products in a fabrication line, during their life time. Hence, the controller has to be *robust* against dynamic varia-

tions that come from, for example, variations in the fabrication environment, temperature change, and mechanical imperfections due to the elapse of time.

In this chapter, the MRC method of Chapter 2 is used to design track-following controllers for HDDs. Here, the variations in the product line is assumed to be the source of a time-invariant uncertainty in the HDD products. Experimental frequency response function (FRF) data sets from a batch of similar HDDs are used to determine a linear time-invariant uncertain model for the servo mechanism. To this model, the MRC method of Chapter 2 is applied. The performance of the MRC is compared to the one of a traditional robust controller through simulations and experiments.

This chapter is organized as follows. Section 3.2 formulates a robust track-following control problem in HDDs. The experimental setup is briefly described in Section 3.3. Section 3.4 explains an automated procedure for multiple robust track-following controller design. Parameter uncertainty modeling and controller design results are presented in Sections 3.5 and 3.6, respectively.

3.2 Track-following servo control in HDDs

In this section, we first address a general robust track-following control problem in HDDs. As a control system, the input of a traditional HDD is an electrical signal to the VCM, and the output is the position error signal (PES). There are disturbance signals affecting this control system, such as the windage from the spinning disk surface, the winding reference track, and the signal noise.

The robust track-following problem in HDDs considered in this chapter is described as follows:

For a large number of HDDs, design a controller (or a small number of controllers) so that each HDD paired with the designed controller (or a controller among designed controllers) generates an acceptably small PES in the face

of disturbance signals.

If the dynamic properties of a large number of HDDs are similar and the PES requirement is reasonable, it is likely that one single robust controller suffices for acceptable track-following. However, if it is difficult to design a single robust controller to solve a track-following problem due to relatively large dynamic variations of HDDs and the stringent PES requirement, multiple robust track-following controllers will be introduced.

3.3 HDD experimental setup

As a prelude to the demonstration and verification of modeling and controller design methods in the following sections, we will describe an HDD experimental setup, shown in Figure 3.2. The equipment, listed below, is quite standard for HDD servo experiments.

- A number of identical HDDs N256 (Maxtor)
- laser doppler vibrometer (LDV) OFV-5000 and OFV-551 (Polytec)
- Anti-vibration table RS3000 (Newport)
- Amplifier TA105-A14 (TRUST Automation Inc.)
- FFT Dynamic Signal Analyzer 35670A (Agilent Technologies)
- Controller Board DS1103 (dSPACE Inc.)

3.4 A proposed automated procedure for multiple robust track-following controller design

Here, we consider uncertainty in HDD systems due to product variations. Since such uncertainty is time-invariant, the method of Chapter 2 can be used to design MRC. The procedure proposed in this section forms n_d categories of HDDs, and n_d corresponding robust controllers. We emphasize

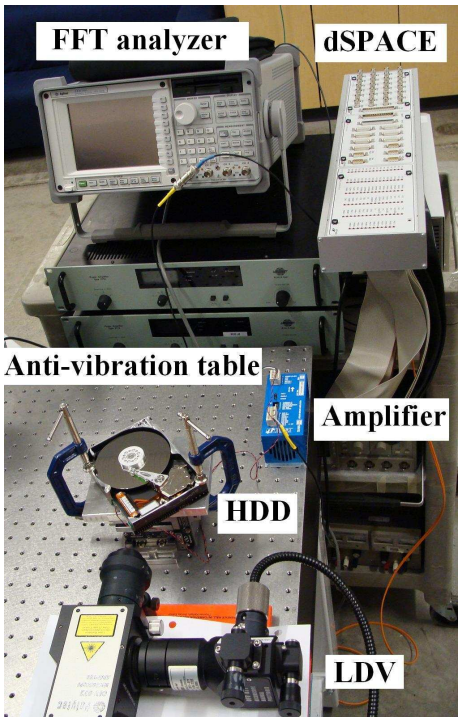


Figure 3.2: HDD experimental setup.

that the number n_d is *not* pre-specified, but automatically determined by our procedure. The procedure consists of three steps:

1. **Parameter uncertainty modeling:** First, by sampling a sufficient number of HDDs from the production line, and by taking FRF data for each sampled HDD, we build a set of transfer functions with parametric uncertainties that represents not only the sampled HDDs but also "intermediate" HDDs. Here, the model structure is not automatically determined, and is manually selected. (See Section 3.5.)

2. **MRC design:** For the obtained model with parametric uncertainties, a specified controller structure, and design specifications in the frequency-domain, we design MRC by following the method of Chapter 2. MRC is designed so that each controller is specialized to achieve

high track-following performance for only one category of HDDs.

3. **Controller selection:** At the end of the HDD production line, we set up a categorization mechanism to identify which category each HDD belongs to, by some system identification technique. After the categorization, each HDD that belongs to the i-th category automatically selects the i-th controller, where i ($i = 1, \ldots, n_d$) is the numbering of HDD categories, and n_d is the number of categories. In this way, all the HDDs can achieve high track-following performance.

3.5 Parameter uncertainty modeling results

Using the experimental setup in Section 3.3, we measured two sets of FRF data for each of the five HDDs, and thus had 10 FRF data sets in total. As can be seen in Figure 3.3(a), all 10 data sets have similar gain and phase curves, but manufacturing variations clearly exist.

By inspection, we have selected the model structure as [2]

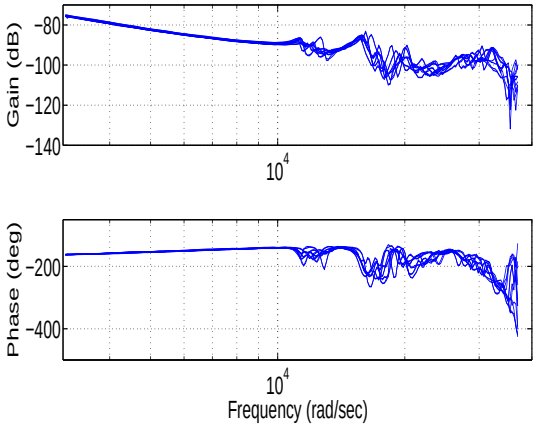
$$\mathcal{G}(\delta) := \frac{K}{s^2} \prod_{i=1}^{4} \frac{s^2 + 2\zeta_{n_i}\omega_{n_i}s + \omega_{n_i}^2}{s^2 + 2\zeta_{d_i}\omega_{d_i}s + \omega_{d_i}^2}, \tag{3.1}$$

where K and $\{\zeta_{n_i}, \omega_{n_i}, \zeta_{d_i}, \omega_{d_i}\}_{i=1}^{4}$ can be described as second order functions of the uncertain parameter $\delta \in \mathbf{\Delta} := \{\delta \in \mathbb{R}, |\delta| \leq 1\}$ by

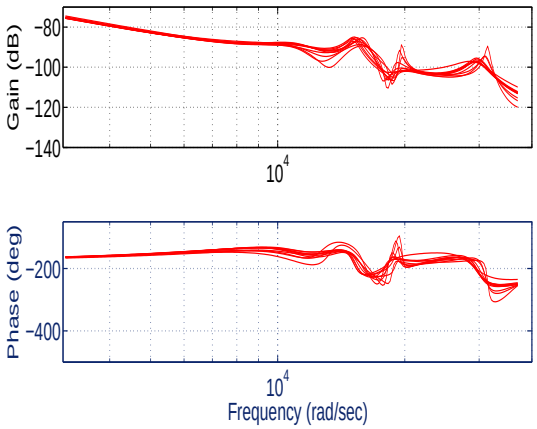
$$f(\delta) := f_0 + f_1\delta + f_2\delta^2, \tag{3.2}$$

We have obtained the coefficient vectors f_0, f_1 and f_2 of the function f as shown in Table 3.1. Figure 3.3(b) shows the Bode plots of 10 samples from the set $\mathcal{G}(\delta)$. The figure illustrates that the obtained model set captures the major characteristics of the 10 FRF data sets.

[2]The modeling and parameter identification of HDDs are not within the scope of this thesis. The results shown in this section are taken from [18].



(a) 10 FRF data (2 FRF are taken for each HDD).



(b) 10 random samples of the uncertain transfer function (3.1).

Figure 3.3: Uncertainty modeling result.

3.6 Controller design results

For the transfer function set $\mathcal{G}(\delta)$ obtained in Section 3.5, we design MRC, as well as a single robust controller designed by μ-synthesis technique of [85] for comparison. A weighted H_∞ norm is considered for controller design with

Table 3.1: Numerical values of the parameters in the transfer function (3.1).

parameter	f_0	f_1	f_2
K	1360.4	-389.1	181.1
ζ_{n_1}	0.612	0.0101	-0.6
ω_{n_1}	13345	127	-719
ζ_{d_1}	0.2104	0.0544	-0.0825
ω_{d_1}	11077	-1	15
ζ_{n_2}	0.1746	0.2306	0.5195
ω_{n_2}	14276	573	-144
ζ_{d_2}	0.0329	0.013	0.0346
ω_{d_2}	15650	148	-604
ζ_{n_3}	0.0196	-0.01	-0.007
ω_{n_3}	17746	-469	1300
ζ_{d_3}	0.059	0.011	-0.044
ω_{d_3}	18432	-420	695
ζ_{n_4}	0.466	0.0705	-0.1871
ω_{n_4}	31419	3366	2889
ζ_{d_4}	0.055	-0.0129	-0.0251
ω_{d_4}	30740	1926	-1272

the multi-band weight function given by

$$W_{mb} = \begin{cases} 0.5 & \omega \leq 5 \times 10^3, \\ 2 & 6 \times 10^3 \leq \omega \leq 4 \times 10^4, \\ 1.15 & 5 \times 10^4 \leq \omega. \end{cases} \tag{3.3}$$

The weight function W_{mb} is chosen such that, for the sensitivity transfer function, the low frequency (i.e., $\omega \leq 5 \times 10^3$) gain is less than -6 dB, the peak is less than 6 dB, and the high frequency (i.e., $\omega \geq 5 \times 10^4$) gain is less than 1.2 dB.

Based on the proposed multiple controller design procedure, two controllers are designed. The partition of $\mathbf{\Delta}$ and the achieved values of the weighted infinity norms are summarized in Table 3.2.

HDD category i	1	2
$\boldsymbol{\Delta}^{(i)}$	$[-1, 0.859]$	$[0.859, 1]$
$\max_{\delta \in \boldsymbol{\Delta}^{(i)}} \left\|W_{mb} T_{w\to z}(K^{(i)}, \Delta)\right\|_\infty$	0.869	0.885

Table 3.2: Achieved values of $\max_\delta \left\|W_{mb} T_{w\to z}(K, \Delta)\right\|_\infty$

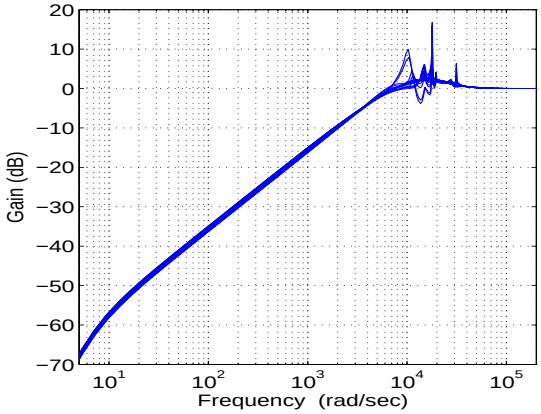
Next, we evaluate the closed-loop performances of the designed MRC in both the frequency and time domains.

3.6.1 Frequency-domain evaluations

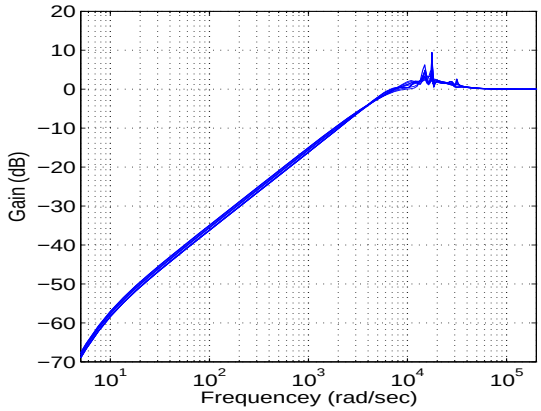
Here, by plotting the sensitivity functions, we compare the effectiveness of the MRC. Figures 3.4 and 3.5 show the simulated sensitivity functions and those in the experiments, respectively. In these figures, "Single controller" means the case when we apply the single robust controller to all five HDDs, while "Multiple controllers" means the case when the controller $K^{(i)}$, $i = 1, 2$ is applied to the HDDs belonging to the HDD category i. As can be seen in these figures, MRC generally yield lower gain peaks, leading to larger stability margins and less oscillatory time-domain responses. It should be emphasized that however the single robust controller is well tuned, it performs worse than the MRC for all of the HDDs. The reason is that the single robust controller tends to be more conservative in order to achieve robust performance for all HDDs.

3.6.2 Time-domain evaluations

root mean square (RMS) values of the R/W head position for windage generated by the spinning disk are compared in Table 3.3, where "Single" and "Multiple" indicate the same cases as in the frequency domain evaluations. This table shows which HDD belongs to which HDD category, and also indicates that MRC provides much smaller RMS values of the head position vibration signal than a single controller does for all the HDDs. This result



(a) Single controller



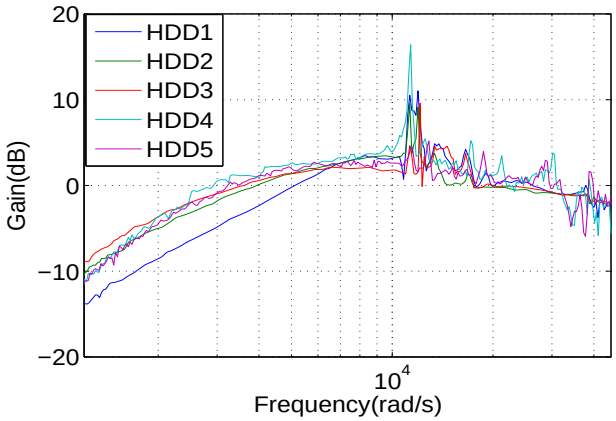
(b) Multiple controllers

Figure 3.4: Sensitivity functions by simulations.

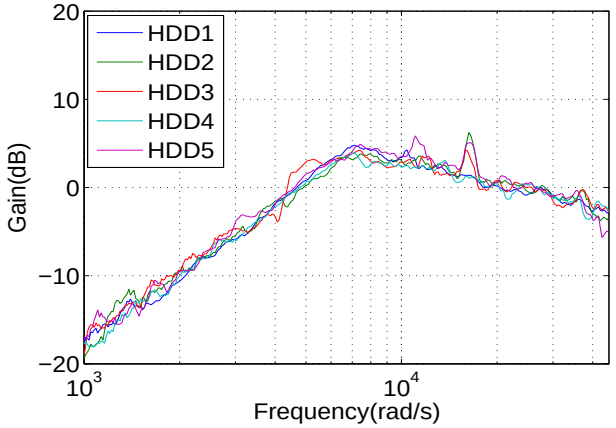
implies that multiple controllers have potential for significantly improving the vibration suppression performance of the R/W head.

Finally, for one of the five HDDs, the open-loop and the closed-loop (with a corresponding controller in the designed MRC) responses are shown in Figure 3.6. From this time domain signal, it is evident that the controller

[3]Calculated by $100(\text{RMS}_{\text{single}} - \text{RMS}_{\text{multiple}})/\text{RMS}_{\text{single}}$.



(a) Single controller



(b) Multiple controllers

Figure 3.5: Sensitivity functions by experiments.

suppresses the head vibration.

3.7 Conclusions

In this chapter, we applied the MRC design method to the track-following control problem in HDDs in order to improve the tracking performance. The procedure is automated, in the sense that given a set of experimental fre-

HDD number	1	2	3	4	5
HDD category	II	I	I	II	I
Single	55.1	58.3	89.0	60.8	57.8
Multiple	48.1	42.8	61.3	47.9	44.1
Improvement[3](%)	12.7	26.6	31.1	21.2	23.7

Table 3.3: Root mean square values of the head position for closed-loop systems (nm) and performance improvement (%).

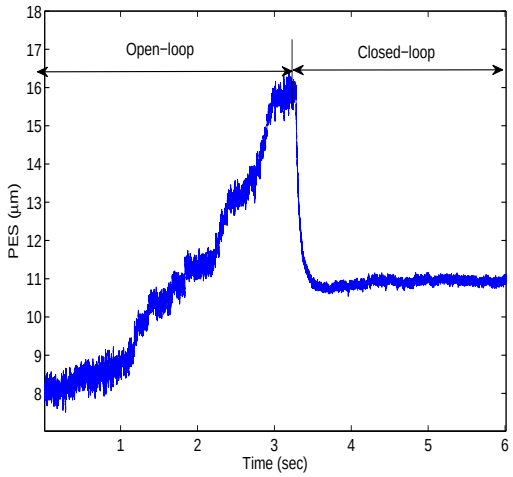


Figure 3.6: Time response of open-loop and closed-loop systems (there is an offset around 11 μm).

quency responses and user inputs (that is, a model and uncertainty structure, closed-loop performance specifications in the frequency domain, and controller structure constraints, if any), it automatically generates a MRC. Both simulations and experiments on actual HDDs demonstrated the usefulness and the efficiency of the proposed controller design method.

Chapter 4

Switching Multiple Robust Control for Linear Time-Varying Plants [1]

4.1 Introduction

This chapter proposes a switching MRC design method for linear time-varying (LTV) plants. Similar to the method of Chapter 2, the proposed controller design algorithm divides the parameter set into a number of subsets, and designs a robust time-invariant controller for each parameter subset. However, unlike the method of chapter 2 where the controller was time-invariant in implementation, here the controller may switch according to the value of time-varying parameters. Hence, stability conditions should be met not only for each controller but also for the switching controllers.

The design algorithm consists of iterations of two main steps; parameter division and switching stability verification. In the parameter set division step, by solving an optimization problem, a parameter subset is chosen and a robust controller, which meets the desired performance, is designed for

[1] This chapter is based on the following article: E. Azadi Yazdi and R. Nagamune, *"A parameter set division and switching gain-scheduling controllers design method for time-varying plants"*, submitted for publication.

that parameter subset. The integral quadratic constraint (IQC) technique is used in this step to formulate the optimization problem whose cost function is neither smooth nor convex. A gradient-based nonsmooth optimization method is introduced to find a local solution to this problem.

In the second step of the design algorithm, the closed-loop stability is verified for the switching between controllers of the recently designed parameter subset and all of its neighbors. In the case of switching instability, the recently designed parameter subset is modified by following a certain procedure. The iterations of these two steps are terminated when the entire parameter set is covered by parameter subsets.

This chapter is organized as follows. Section 4.2 formulates a switching MRC design problem for LTV plants. In Section 4.3, a controller design method is proposed to the formulated problem. The method encounters an optimization problem whose cost function is neither smooth nor convex. Hence, in Section 4.4 a nonsmooth optimization technique is used to solve this problem. Section 4.5 provides a numerical example to validate our algorithm.

4.2 Switching MRC design problem for LTV plants

4.2.1 Closed-loop system interconnection

We consider an LTV system with a real time-varying matrix $\Delta(t)$[2] expressed by the state space equation (see Figure 4.1):

$$\begin{bmatrix} \dot{x}(t) \\ z(t) \\ y(t) \end{bmatrix} = F_u(P, \Delta(t)) \begin{bmatrix} x(t) \\ w(t) \\ u(t) \end{bmatrix}. \tag{4.1}$$

[2]Other types of uncertain elements, such as parametric uncertainty, can be easily incorporated in this method by following the path of [80]. However, for simplicity of presentation, we do not consider them in the problem formulation.

In 4.1, $x(t) \in \mathbb{R}^n$ is the state vector, $w(t) \in \mathbb{R}^{n_w}$ the exogenous input vector, $u(t) \in \mathbb{R}^{n_u}$ the control input vector, $z(t) \in \mathbb{R}^{n_z}$ the performance vector, $y(t) \in \mathbb{R}^{n_y}$ the measurement vector, $\Delta(t) \in \mathbb{R}^{n_\Delta \times n_\Delta}$ the time-varying matrix, and $P \in \mathbb{R}^{(n_\Delta+n+n_z+n_y)\times(n_\Delta+n+n_w+n_u)}$ the system matrix. The mapping $\Delta : \mathbb{R}_+ \to \mathbb{R}^{n_\Delta \times n_\Delta}$ is assumed to be in the following time-varying matrix-valued function set $\boldsymbol{\Delta}_{TV}$:

$$\boldsymbol{\Delta}_{TV}(\boldsymbol{\Delta}, \boldsymbol{\Delta}_v) := \left\{ \Delta(t) := \mathrm{diag}\{\boldsymbol{\delta}(t)\} : \boldsymbol{\delta}(t) \in \boldsymbol{\Delta}, \dot{\boldsymbol{\delta}}(t) \in \boldsymbol{\Delta}_v, \forall t \in \mathbb{R}_+ \right\}, \tag{4.2}$$

where the time-varying vector $\boldsymbol{\delta}(t)$ and its derivative are assumed to be bounded as

$$\begin{aligned} \boldsymbol{\Delta} &:= \left\{ \boldsymbol{\delta}(t) \in \mathbb{R}^{n_\Delta} : \|\boldsymbol{\delta}(t)\|_\infty \le 1 \right\}, \\ \boldsymbol{\Delta}_v &:= \left\{ \dot{\boldsymbol{\delta}}(t) \in \mathbb{R}^{n_\Delta} : \left\| \mathrm{diag}(v)^{-1} \dot{\boldsymbol{\delta}}(t) \right\|_\infty \le 1 \right\}. \end{aligned} \tag{4.3}$$

In 4.3, each component of the vector $v \in \mathbb{R}^{n_\Delta}$ designates the bound of the corresponding component of $\dot{\boldsymbol{\delta}}(t)$, and is assumed to be given. Throughout this chapter, we assume that the time-varying vector $\boldsymbol{\delta}(t)$ is available in real time.

To the LTV plant (4.1), we connect an LTV controller represented by

$$\begin{bmatrix} \dot{x}_k(t) \\ u(t) \end{bmatrix} = K(t) \begin{bmatrix} x_k(t) \\ y(t) \end{bmatrix}, \tag{4.4}$$

where $x_k(t) \in \mathbb{R}^{n_k}$ is the controller state vector and $K(t) \in \mathbb{R}^{(n_k+n_u)\times(n_k+n_y)}$ a system matrix for the controller. Since the vector $\boldsymbol{\delta}(t)$ is available online, the controller parameters can be adjusted based on $\boldsymbol{\delta}(t)$ for performance improvement. Such controllers are called *gain-scheduling controllers* [89].

4.2.2 Gain-scheduling controller synthesis problem with L_2-gain performance

For the plant given in (4.1), we revisit the well-studied gain-scheduling controller synthesis problem with guaranteed L_2-gain. To be more specific, the problem is to design a controller $K(\boldsymbol{\delta}(t))$ which ensures (i) asymptotic stability, i.e. for $w(t) = 0$

$$\left[x^T(t), x_k^T(t)\right] \to 0 \text{ as } t \to \infty, \quad \forall \Delta \in \boldsymbol{\Delta}_{TV}(\boldsymbol{\Delta}, \boldsymbol{\Delta}_v), \tag{4.5}$$

and (ii) a guaranteed L_2-gain bound γ for the closed-loop system from the disturbance signal $w(t)$ to the error signal $z(t)$, that is

$$\|z\|_2 \leq \gamma \|w\|_2, \tag{4.6}$$

for any $\Delta \in \boldsymbol{\Delta}_{TV}(\boldsymbol{\Delta}, \boldsymbol{\Delta}_v)$ and zero-state initial conditions.

This problem is not new. In fact, several solutions to this problem have been proposed in the literature, see [89] and the references therein. One of

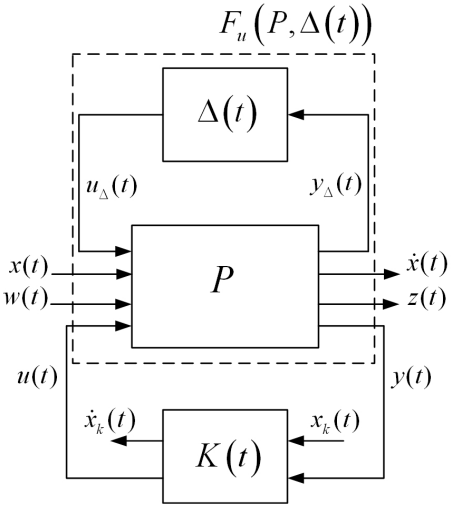


Figure 4.1: A system matrix P, the uncertainty matrix $\Delta(t)$, and the controller $K(t)$.

the popular solutions is the method of [3], which utilizes the linear matrix inequality (LMI) technique to design gain-scheduling controllers in a numerically efficient way. The disadvantage of the method of [3], as well as similar methods [4, 99], is that they can be very conservative or even infeasible for large parameter sets $\mathbf{\Delta}_{TV}$, or even infeasible for a small desired L_2-gain γ. Such conservatism is a common limitation that most of the robust control synthesis methods suffer from.

This disadvantage can be overcome by dividing the parameter set into smaller subsets and designing a controller for each subset [78, 107]. In this approach, since each controller has to be robust against parameter variations in a small subset of the parameter set, relatively small desired L_2-gains can be achieved. There are two differences between the methods in [107] and [78]; firstly, the former method switches between linear parameter varying controllers, while the latter uses LTI controllers. Secondly, in the former the divisions in the parameter set should be fully specified by users, while in the latter only the center of parameter subsets needs to be specified by users and the radius of subsets will be determined by the method.

Hence, in both methods of [78] and [107], the number of parameter divisions and their centers should be specified by the user. Although this simplifies the controller design, to prevent a conservative result, the user needs to have a prior knowledge about "difficulty" of achieving the desired robust performance in different regions of the parameter set. For example, the user should assign smaller parameter subsets in regions of the parameter set that are "difficult" to control in order to achieve the desired robust performance. Since such *a priori* knowledge is often unavailable, the parameter set division and the controller design can be a time-consuming trial and error process.

4.2.3 Design problem of the switching MRC for LTV plants

In this section, a brief description of the control configuration of switching MRC for LTV plants is provided, and a controller design problem, which is an extension to the one in [107], will be formulated.

switching multiple robust controllers (MRC)

A switching MRC consists of a finite number of LTI controllers $K^{(i)}$, and a piecewise constant switching scheme $\sigma(t)$ specifying at each time instant t which LTI controller is connected to the plant. Figure 4.2 depicts a simple static gain-scheduling control example that shows the difference between a continuous and a switching gain-scheduling controller. In switching MRC approach, each LTI controller $K^{(i)}$ satisfies the guaranteed L_2-gain performance for a parameter subset $\boldsymbol{\Delta}^{(i)}$ in $\boldsymbol{\Delta}$. When the scheduling vector $\boldsymbol{\delta}(t)$ moves from one subregion to another, a switching occurs according to the switching rule in place.

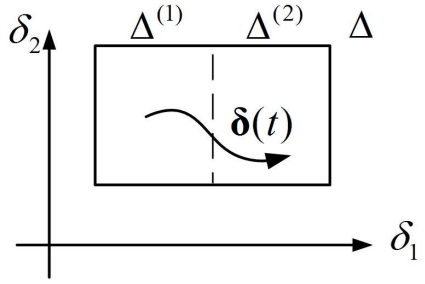
Problem formulation

The *switching MRC design problem* to be considered in this chapter is formulated as follows.

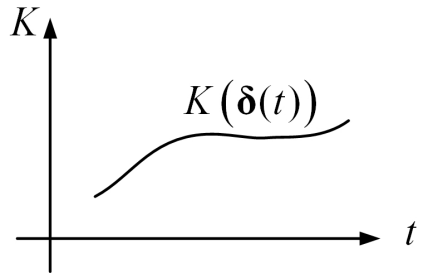
Problem 4.2.1. *For a given plant (4.1) with the time-varying matrix-valued function set $\boldsymbol{\Delta}_{TV}(\boldsymbol{\Delta}, \boldsymbol{\Delta}_v)$, and a desired L_2-gain performance objective γ, design a set of pairs of parameter subsets and LTI controllers $\{(\boldsymbol{\Delta}^{(i)}, K^{(i)})\}_{i=1}^{n_d}$ and a switching rule $\sigma(t)$, such that*

(i) The set $\boldsymbol{\Delta}$ is covered by $\{\boldsymbol{\Delta}^{(i)}\}_{i=1}^{n_d}$, i.e.

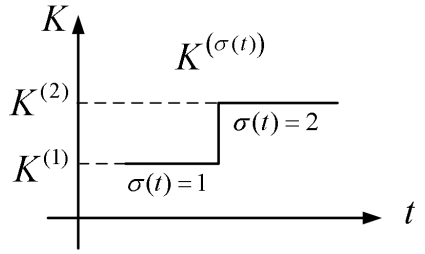
$$\boldsymbol{\Delta} = \bigcup_{i=1}^{n_d} \boldsymbol{\Delta}^{(i)}, \tag{4.7}$$



(a) Scheduling parameter trajectory.



(b) A continuous gain-scheduling static controller.



(c) A switching multiple robust static controller.

Figure 4.2: Continuous gain-scheduling vs. switching MRC.

where $\boldsymbol{\Delta}^{(i)}$ *is a hyper-cube characterized by its center* $\Delta^{(i)}_{\text{nom}} := \text{diag}\{\boldsymbol{\delta}_{\text{nom}}\}$ *and radius* $\tau^{(i)}$ *as*

$$\boldsymbol{\Delta}^{(i)} := \left\{\boldsymbol{\delta}(t) \in \mathbb{R}^{n_\Delta} : \|\boldsymbol{\delta}(t) - \boldsymbol{\delta}_{\text{nom}}\|_\infty \leq \tau^{(i)}\right\}. \tag{4.8}$$

(ii) The closed-loop system is asymptotically stability for all $\Delta \in \mathbf{\Delta}_{TV}(\mathbf{\Delta}, \mathbf{\Delta}_v)$.

(iii) For all $i = 1, \ldots, n_d$*, the* L_2*-gain performance objective*

$$\|z\|_2 \leq \gamma \|w\|_2, \tag{4.9}$$

is satisfied for all plants in $\mathbf{\Delta}_{TV}(\mathbf{\Delta}^{(i)}, \mathbf{\Delta}_v)$ *equipped with the controller* $K^{(i)}$.

There are a number of remarks for the formulated problem. Firstly, unlike Chapter 2, the designed parameter subsets $\mathbf{\Delta}^{(i)}$ are not necessarily disjoint and may be overlapping. Secondly, we emphasize that the number of parameter set divisions n_d is not pre-specified in this problem setting; it will be automatically determined by the controller design algorithm. Thirdly, only the set $\mathbf{\Delta}$ is divided into subsets and the set $\mathbf{\Delta}_v$ will not be. This is because the information $\dot{\boldsymbol{\delta}}(t)$ will not be used for gain-scheduling purpose in this thesis, due to practical invalidity [3]. Thirdly, selecting the controller at each instant of time requires a knowledge of the value of the variable $\boldsymbol{\delta}(t)$. This value can be obtained through either direct measurement of the variable or a system identification method (e.g. [22, 40, 46]). Lastly, the solution of this problem is an approximate solution to the gain-scheduling controller synthesis problem with guaranteed L_2-gain performance, because the L_2-gain condition is only guaranteed in each parameter subset and may not be satisfied in switching instances.

4.3 Controller design

This section proposes a switching MRC design method. The flowchart of this method is shown in Figure 4.3, and will be explained in detail.

The parameter d is specified by the user at the beginning of the flowchart. This parameter indicates the width of the overlapping region between two adjacent parameter subsets, and is used to guarantee switching stability. In

addition, the parameter subset to which a robust controller has not been assigned, shown by $\mathbf{\Delta}_{\text{unpart}}$, is the entire parameter set $\mathbf{\Delta}$.

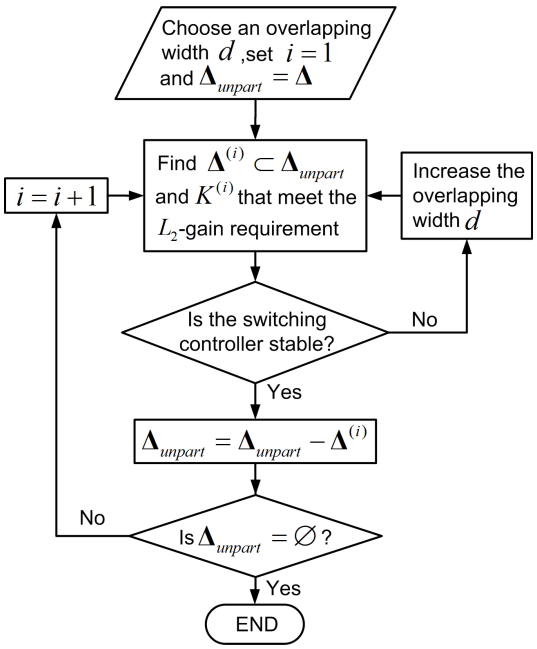


Figure 4.3: The flowchart of the switching MRC design method.

Given d and $\mathbf{\Delta}_{\text{unpart}}$, the first step in the flowchart is to find a parameter subset $\mathbf{\Delta}^{(i)} \subset \mathbf{\Delta}_{\text{unpart}}$ and an LTI controller $K^{(i)}$ which meets the desired L_2-gain requirement robustly over the parameter set $\mathbf{\Delta}_{TV}(\mathbf{\Delta}^{(i)}, \mathbf{\Delta}_v)$. The parameter set $\mathbf{\Delta}^{(i)}$ is designed to have an overlapping width d with the previously established neighbor subsets. In the next step, the switching stability between the newly designed controller and previously established ones, if any, is verified. If the stability conditions fail, the overlapping width d is increased and the recently designed controller is redesigned accordingly. Otherwise, the parameter subset $\mathbf{\Delta}^{(i)}$ is removed from the set $\mathbf{\Delta}_{\text{unpart}}$ and the partitions are modified according to Algorithm 2.3.4 which will be described later. The iteration of these steps will be continued until the set $\mathbf{\Delta}_{\text{unpart}}$ becomes an empty set.

Figure 4.4 depicts a two dimensional parameter set $\mathbf{\Delta}$ that has undergone the switching gain-scheduling controllers design algorithm explained in the flowchart of Figure 4.3. The parameter set is divided into two subsets $\mathbf{\Delta}^{(1)}$ and $\mathbf{\Delta}_{\text{unpart}}$ at the end of the first iteration as shown in Figure 4.4(a). In each iteration, one new parameter subset will be established as shown in Figure 4.4(b). The iteration is terminated when the original parameter subset is covered with parameter subsets $\mathbf{\Delta}^{(i)}$ (Figure 4.4(c)).

4.3.1 Finding a parameter subset $\mathbf{\Delta}^{(i)}$ and a robust controller $K^{(i)}$

In this section, we present an optimization method that finds a parameter subset $\mathbf{\Delta}^{(i)}$ and a stabilizing controller $K^{(i)}$ which meets the desired L_2-gain requirement for all plants in the set $\mathbf{\Delta}_{TV}(\mathbf{\Delta}^{(i)}, \mathbf{\Delta}_v)$. The optimization problem is based on a sufficient condition for an LTI controller $K^{(i)}$ to satisfy the asymptotic stability and the L_2-gain requirement for plants in the parameter set $\mathbf{\Delta}_{TV}(\mathbf{\Delta}^{(i)}, \mathbf{\Delta}_v)$, presented in the following theorem. (We omit the superscript "(i)" in the theorem.)

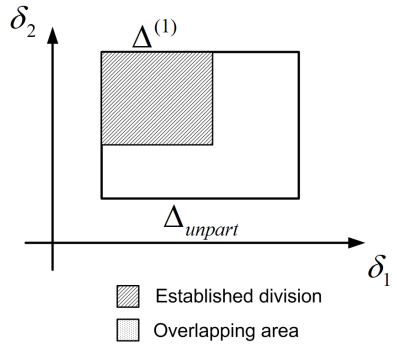
Theorem 4.3.1. *Given the plant P, the controller K, and the nominal parameter value $\Delta_{\text{nom}} \in \mathbf{\Delta}$, the closed-loop system Figure 4.1 is asymptotically stability and satisfies L_2-gain requirement $\|z\|_2 \leq \gamma\|w\|_2$ for all $\Delta \in \mathbf{\Delta}_{TV}(\mathbf{\Delta}, \mathbf{\Delta}_v)$, if there exists a multiplier $\Pi(\tau)$ defined in (B.6) such that*

$$f(K, \Delta_{\text{nom}}, \Pi(\tau)) := \max_{\omega \in [0,\infty]} \lambda_{max}(\mathcal{F}(K, \Delta_{\text{nom}}, \Pi(\tau); j\omega)) < 0, \tag{4.10}$$

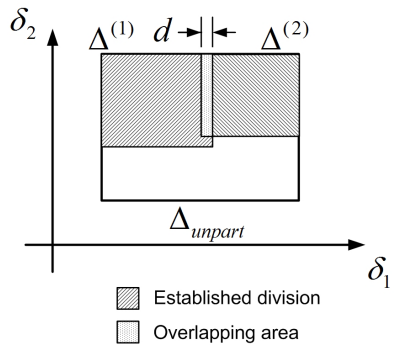
where

$$\mathcal{F}(K, \Delta_{\text{nom}}, \Pi(\tau); j\omega) := \begin{bmatrix} F_l\left(P_{\Delta_{\text{nom}}}, K\right) \\ I \end{bmatrix}^H \Pi(\tau) \begin{bmatrix} F_l\left(P_{\Delta_{\text{nom}}}, K\right) \\ I \end{bmatrix}, \tag{4.11}$$

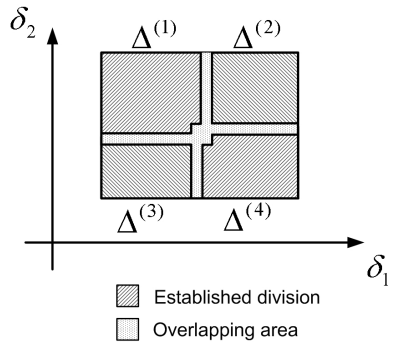
and $P_{\Delta_{\text{nom}}}$ is the frequency-domain representation of the plant corresponding



(a) The parameter set after the first iteration.



(b) The parameter set after the second iteration..



(c) Final parameter set divisions.

Figure 4.4: An illustration of the parameter set divisions of the switching MRC design method.

to the parameter Δ_{nom}.

The proof of Theorem 4.3.1 is presented in Appendix A.

Using Theorem 4.3.1, the following optimization problem is formulated to find a parameter subset $\mathbf{\Delta}^{(i)}$ and a robust controller $K^{(i)}$ as

$$\max \tau^{(i)}, \text{ subject to } f(x, \tau^{(i)}) < 0, \tag{4.12}$$

where x is a vector that contains all the parameters of $K^{(i)}$, $\Delta_{\text{nom}}^{(i)}$, and Π. In the above optimization problem x and $\tau^{(i)}$ are the optimization variables. In (4.12), the cost function represents the size of the parameter subset that is to be maximized in order to reduce the number of parameter subsets required to cover the parameter set, and hence the computation time. Meanwhile, based on Theorem 4.3.1, the constraint in (4.12) ensures the desired L_2-gain as well as asymptotic stability for all plants with the time-varying parameter $\boldsymbol{\delta}(t)$ in the set $\mathbf{\Delta}^{(i)}$ which is characterized by $\Delta_{\text{nom}}^{(i)}$ and $\tau^{(i)}$.

We simplify the constrained optimization problem (4.12) by breaking it into two nested unconstrained optimization problems. An outer optimization which maximizes the radius of the parameter subset, i.e. $\tau^{(i)}$, and an inner optimization that, for a fixed $\tau^{(i)}$, tries to meet the constraint of the optimization problem (4.12) by solving

$$\min_{x} f(x, \tau^{(i)}). \tag{4.13}$$

In the outer optimization, x is kept fixed and $\tau^{(i)}$ is the optimization variable, while in the inner optimization, $\tau^{(i)}$ is kept fixed and x is the optimization variable.

Since the outer optimization is a linear formulation, traditional linear programming methods can be used. On the other hand, the cost function of the inner optimization (4.13) is a nonsmooth function due to the following two facts. Firstly, the maximum eigenvalue function may switch between different eigenvalues. Secondly, the change in the frequency at which the

maximum eigenvalue is attained induces nonsmoothness in the cost function. A nonsmooth optimization technique will be used in Section 4.4 to find a local optimum for (4.13).

4.3.2 Hysteresis switching between controllers

The time-varying parameter vector $\boldsymbol{\delta}(t)$ may exit one parameter subset and enter another one. In such cases, the controller has to be switched properly to maintain the asymptotic stability. Therefore, the stability of the proposed switching gain-scheduling controllers should be studied. Here, we use a hysteresis switching rule that guarantees a minimum dwell time, and hence simplifies the stability analysis. See, e.g., [67, 81].

Figure 4.5 depicts a situation where multiple sequential hysteresis switchings occur between two controllers. The system has a one dimensional parameter set $\boldsymbol{\Delta}$ that is divided into two parameter subsets $\boldsymbol{\Delta}^{(1)}$ and $\boldsymbol{\Delta}^{(2)}$ with an overlapping width d, as shown in Figure 4.5.

A Lyapunov function approach is taken to analyze the stability of such hysteresis switching controllers. Multiple parameter-dependent Lyapunov functions can be defined as

$$V^{(\sigma)}(x_{cl}, \boldsymbol{\delta}) := x_{cl}^T X^{(\sigma)}(\boldsymbol{\delta}) x_{cl}, \quad \forall \boldsymbol{\delta} \in \boldsymbol{\Delta}^{(\sigma)} \tag{4.14}$$

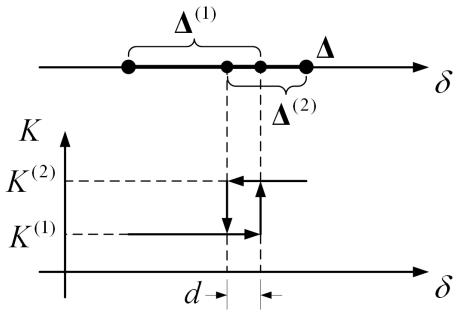


Figure 4.5: Hysteresis switching of switching MRC for a one dimensional parameter set.

where the Lyapunov variable $X^{(\sigma)}$ is a function of the time-varying parameter. The multiple parameter-dependent Lyapunov functions method is proposed in [78] to reduce the conservatism of the conventional multiple Lyapunov functions method. Following the path of [78], Theorem 4.3.2 gives a sufficient condition for stability of switching gain-scheduling controllers.

Theorem 4.3.2. *Given a set of pairs of* $\{(\boldsymbol{\Delta}^{(i)}, K^{(i)})\}_{i=1}^{n_d}$*, the switching gain-scheduling controllers internally stabilize if the following three matrix inequalities hold for all* $i, j \in \{1, \ldots, n_d\}$*,* $\boldsymbol{\delta} \in \boldsymbol{\Delta}^{(i)}$ *, and* $\hat{\boldsymbol{\delta}} \in \boldsymbol{\Delta}^{(i)} \cap \mathbb{B}(\boldsymbol{\Delta}^{(j)})$

$$X^{(i)}(\boldsymbol{\delta}) > 0, \tag{4.15}$$

$$\mathcal{A}^T(K^{(i)}, \boldsymbol{\delta}) X^{(i)}(\boldsymbol{\delta}) + X^{(i)}(\boldsymbol{\delta}) \mathcal{A}(K^{(i)}, \boldsymbol{\delta}) + \sum_{k=1}^{n_\Delta} \dot{\delta}_k \frac{\partial X^{(i)}(\boldsymbol{\delta})}{\partial \delta_k} < 0, \tag{4.16}$$

$$X^{(i)}(\hat{\boldsymbol{\delta}}) \leq X^{(j)}(\hat{\boldsymbol{\delta}}), \tag{4.17}$$

where $\mathcal{A}(K^{(i)}, \boldsymbol{\delta})$ *is the closed-loop A-matrix of the plant equipped with the controller* $K^{(i)}$*, and* $\mathbb{B}(\boldsymbol{\Delta})$ *is the set of all boundary elements of the compact set* $\boldsymbol{\Delta}$*.*

Proof. In a switching system, two conditions should be satisfied by a positive definite Lyapunov function to conclude stability; it should decrease in each mode, and it should not increase in switching instances. The conditions of Theorem 4.3.2 represent these conditions. The inequality (4.15) requires the Lyapunov function to be positive definite. The derivative of the Lyapunov function should be negative definite so that it decreases in each mode, that is guaranteed by (4.16). The Lyapunov function should not increase in switching instances, that is (4.17). □

By parameterizing the Lyapunov variables as $X^{(i)}(\boldsymbol{\delta}) = X_0^{(i)} + \sum_{j=1}^{n_\Delta} X_1^{(i)} \delta_j$, the conditions of Theorem 4.3.2 are LMIs with respect to the parameters of the Lyapunov variable $X^{(i)}(\boldsymbol{\delta})$ for each $\boldsymbol{\delta}$. According to the flowchart in Figure 4.3, the switching stability LMIs (4.15)-(4.17) should be satisfied for all

established parameter subsets. If these LMIs do not hold, the overlapping distance d should be increased and the recently designed controller should be redesigned. According to [67], the larger the overlapping distance d, the larger the minimum dwell time, and hence the less probable the switching stability condition is to fail.

4.4 Nonsmooth optimization

This section proposes a gradient-based method for optimization problem (4.13). Recently, nonsmooth optimization methods have been used to solve several control problems [9]. We follow the path of [9] for developing a nonsmooth descent algorithm. The nonsmooth descent algorithm to solve (4.13) can be summarized as follows.

Algorithm 4.4.1. *Consider the LTI plant (4.1) and the cost function (4.13).*

1. ***Choose an initial controller*** K***, an initial nominal parameter*** Δ_{nom}***, and an initial multiplier*** Π***.***

2. ***Find a descent direction:*** *Find a descent direction h for the cost function (4.13) with respect to optimization parameter x.*

3. ***Perform a line search:*** *Perform a line search to find the optimal amount of movement in the descent directions.*

$$\alpha^* := \arg\min_{\alpha} \left\{ f(x + \alpha h, \tau^{(i)}) \right\} \tag{4.18}$$

4. *Iterate Steps (2)-(3) until a local optimum is achieved.*

In order to use Algorithm 4.4.1, the derivative ∂f of the functional f, defined in (4.10), should be calculated with respect to the controller K, the nominal parameter Δ_{nom}, and the multiplier Π. Since the function f is a nonsmooth function, Clarke subdifferential is used to present its derivative

∂f. The Clarke subdifferential expression turns out to be very long. Hence, in this section, we just provide a procedure to derive the Clarke subdifferential of (4.10).

The function $f(x, \tau^{(i)})$ is a composition of two functions $\max_{\omega\in[0,\infty]} \lambda_{max}(\cdot)$ and $\mathcal{F}(\cdot)$. We obtain the subdifferential of f using the chain rule

$$\partial f = \mathcal{F}'^{*}\Lambda_Y, \tag{4.19}$$

where $\mathcal{F}'$ is the Fréchet derivative of the transfer function $\mathcal{F}(x, \tau^{(i)}; j\omega)$ in (4.11) with respect to parameters x that represents the controller parameters K, the nominal parameter Δ_{nom}, and the multiplier Π, and Λ_Y represents the subdifferential of the function $\max_{\omega\in[0,\infty]}\ \lambda_{max}(\cdot)$. Because of the complexity in the presentation of the adjoint space of $\mathcal{F}'$, the adjoint operator $\mathcal{F}'^{*}$ in (4.19) is not easy to compute. However, the definition of an adjoint, i.e.

$$\langle \mathcal{F}'^{*}\Lambda_Y, \delta x\rangle = \langle \mathcal{F}'(\delta x, \tau^{(i)}; j\omega), \Lambda_Y\rangle, \tag{4.20}$$

provides us with an alternative way of computing $\mathcal{F}'^{*}\Lambda_Y$. The right hand side of (4.20) is an expression that only contains the standard scalar product and is easy to compute. The following theorem presents a method to compute the right hand side of (4.20).

Theorem 4.4.2. *The scalar product on the right hand side of (4.20) can be computed by*

$$\langle \mathcal{F}'(\delta x, \tau^{(i)}; j\omega), \Lambda_Y\rangle = \sum_{\omega\in\Omega(\mathcal{F})} \mathrm{Tr}(Q_\omega Y_\omega Q_\omega^H \mathcal{F}'(\delta x)). \tag{4.21}$$

- *The set $\Omega(\mathcal{F})$, called the set of active frequencies, is the set of frequen-*

cies at which the maximum eigenvalue is attained in (4.10), i.e.,[3]

$$\Omega(\mathcal{F}) := \left\{\omega \in [0,\infty) : \lambda_{max}(\mathcal{F}(x,\tau^{(i)};j\omega)) = f(x,\tau^{(i)})\right\}. \qquad (4.22)$$

- *For every $\omega \in \Omega(\mathcal{F})$, let Q_ω denote a matrix whose columns form an orthogonal basis for the eigenspace of $\mathcal{F}(x,\tau^{(i)};j\omega)$ associated with its largest eigenvalue.*

- *$Y_\omega \geq 0$ are Hermitian matrices that satisfy $\sum_{\omega\in\Omega(\mathcal{F})} \mathrm{Tr}(Y_\omega) = 1$.*

Using Theorem 4.4.2, the following procedure is used to compute the Clarke subdifferential ∂f.

1. Compute the Fréchet derivative of the transfer function $\mathcal{F}(x,\tau^{(i)};j\omega)$ in (4.11) with respect to the parameter x.

$$\mathcal{F}'(\delta x) := \lim_{t\to 0} \frac{\mathcal{F}(x+t\delta x,\tau^{(i)};j\omega) - \mathcal{F}(x,\tau^{(i)};j\omega)}{t}. \qquad (4.23)$$

2. Using Theorem 4.4.2, compute $\langle \mathcal{F}'(\delta x), \Lambda_Y\rangle$ by

$$\langle \mathcal{F}'(\delta x), \Lambda_Y\rangle := \sum_{\omega\in\Omega(\mathcal{F})} \mathrm{Tr}(Q_\omega Y_\omega Q_\omega^H \mathcal{F}'(\delta x)). \qquad (4.24)$$

3. Manipulate the right hand side of the above expression with the trace properties such as $\mathrm{Tr}(AB) = \mathrm{Tr}(BA)$ and $\mathrm{Tr}(A) = \mathrm{Tr}(A^T)$ to obtain a linear form with respect to δx:

$$\langle \mathcal{F}'(\delta x), \Lambda_Y\rangle = \sum_{\omega\in\Omega(F)} \mathrm{Tr}(\delta x \Psi_Y). \qquad (4.25)$$

[3] In practice, it is recommended to use an extended frequency set Ω_e instead of Ω that contains not only the frequencies at which the maximum eigenvalue is attained, but also the frequencies at which the largest eigenvalue is close to the maximum eigenvalue. See [10] for more information about the extended frequency set.

4. Considering the adjoint definition (4.20), the Clarke subdifferential is given by

$$\partial f := \mathcal{F}'^{*}\Lambda_Y = \Psi_Y. \tag{4.26}$$

Because of the freedom on the variable Y_ω, the calculated Clarke subdifferential is a set of the gradient direction of the function f, among which the steepest descent direction should be extracted. Following the line in [87], [9], a semi-definite program (SDP) must be solved to find the steepest descent direction h. See Section 2.4.2 for the steepest descent calculations.

4.5 Mass-spring-damper example

In this section, we apply the proposed switching MRC synthesis method to a mass-spring-damper system with time-varying parameters. As shown in Figure 4.6, the control input u is a force exerted on the mass m_1, and the measurement y is the position of the mass m_2. The parameters of the system are:

$$\begin{aligned} m_1 = m_2 &= 0.5 \text{ kg}, \\ k(t) &= 1 + \delta_k(t) \text{ kg/m}, \\ c(t) &= 0.05 + \delta_c(t) \text{ Ns/m}, \end{aligned} \tag{4.27}$$

where $|\delta_k(t)| \leq 0.11$, $|\dot{\delta}_k(t)| \leq 0.001$, $|\delta_c(t)| \leq 0.011$, and $|\dot{\delta}_c(t)| \leq 0.0001$.

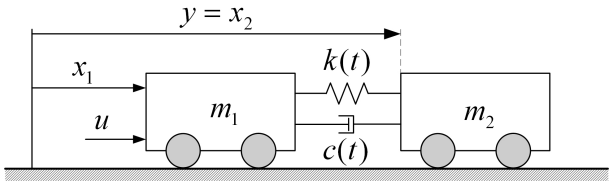


Figure 4.6: Mass-spring-damper system.

With these parameters, the mass-spring-damper system is poorly damped. The nominal controller design techniques are very likely to fail, because they tend to include the inverse of the weakly damped dynamics of the plant in the controller. Note that this problem has previously been studied, see, e.g. [8, 98].

The system dynamics can be described as

$$\begin{cases} \dot{\boldsymbol{x}} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{k}{m_1} & \frac{k}{m_1} & \frac{c}{m_1} & -\frac{c}{m_1} \\ \frac{k}{m_2} & -\frac{k}{m_2} & -\frac{c}{m_2} & \frac{c}{m_2} \end{bmatrix} \boldsymbol{x} + \begin{bmatrix} 0 \\ 0 \\ \frac{1}{m_1} \\ 0 \end{bmatrix} u, \\ y = \begin{bmatrix} 0 & 1 & 0 & 0 \end{bmatrix} \boldsymbol{x}, \end{cases} \tag{4.28}$$

where $\boldsymbol{x} = \begin{bmatrix} x_1 & x_2 & \dot{x_1} & \dot{x_2} \end{bmatrix}^T$ is the state vector.

The following sections present design results of the single robust controller and the switching gain-scheduling controllers, and compare them in various aspects. Inspired by the method of [11], a multiband frequency response shaping technique is used in both controller design problems. Unlike traditional frequency shaping techniques, where weighting functions should be rational, in the multiband frequency response shaping technique the weighting function is a piecewise constant function. For this problem, the multiband weighting function is defined by

$$W_{mb} = \begin{cases} 0.5 & \omega \leq 0.4 \\ 2 & 0.4 \leq \omega \leq 10 \\ 1.15 & 10 \leq \omega \end{cases} \tag{4.29}$$

The weighting function W_{mb} is chosen such that, the low frequency (i.e. $\omega \leq 0.4$) L_2-gain is less than -6 dB, the peak is less than 6 dB, and the high

frequency (i.e. $\omega \geq 10$) L_2-gain is less than 1.2 dB.

4.5.1 Single robust controller

For the time-varying mass-spring-damper system, using IQC method, the achievable L_2-gain is $\gamma = 30$ if a single robust controller is used for the entire parameter set. The performance IQCs that are provided by Theorem B.2.1 are used to design the controller. These IQCs are tackled by using the nonsmooth optimization algorithm that is proposed in Section 4.4. The function $\lambda_{max}(\mathcal{F}(K, \Delta_{\text{nom}}, \Pi(\tau); j\omega))$ is shown in Figure 4.7. Each curve in this figure shows the largest eigenvalue of the function $\mathcal{F}(K, \Delta_{\text{nom}}, \Pi(\tau); j\omega)$, defined in (4.11), in the selected frequency range for 3 consecutive iterations of Algorithm 4.4.1. On each curve, the frequencies of the points that are shown by '*' represent the extended frequency set Ω_e at each iteration.

Figure 4.8 shows the closed-loop sensitivity Bode diagram for a number of randomly sampled plants. In this figure, time-varying parameters are kept constant. Since parameters are time-varying under our assumption, this figure shows only a lower bound on the worst-case L_2-gain of the closed-loop plant.

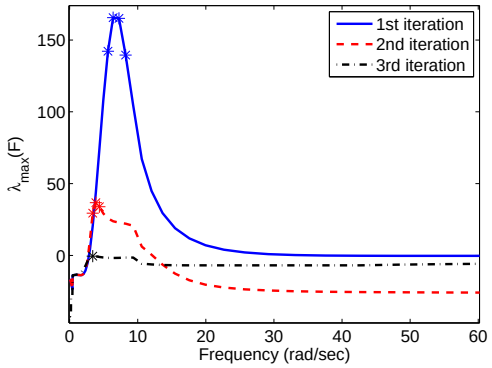


Figure 4.7: The largest eigenvalue of the FDI (B.2).

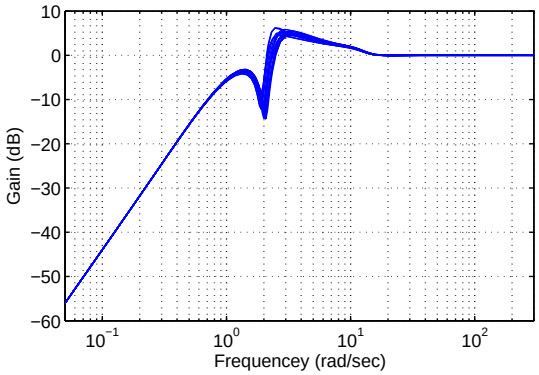


Figure 4.8: The closed-loop sensitivity Bode diagram for a single robust controller.

4.5.2 Switching MRC

In this section, we verify that the proposed switching MRC have a potential to achieve higher performances than the single robust controller. Here, we set the desired L_2-gain to $\gamma = 10$ which leads to an improvement in the closed-loop performance relative to the single robust controller. The initial partition overlapping distance d is chosen to be 10% of the parameter range.

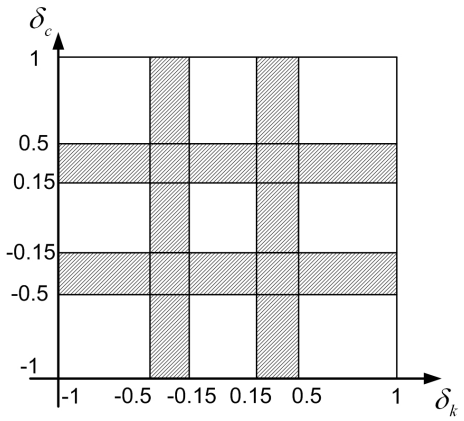


Figure 4.9: Parameter set divisions.

The algorithm divides the parameter set into 9 subsets in order to achieve the desired L_2-gain. Figure 4.9 shows the parameter subsets. In this figure the shaded areas represent the overlapping regions. Figure 4.10 shows the Bode diagrams of the designed LTI controllers. The main difference between the controllers is in low and medium frequency ranges, because the high frequency response depends mainly on masses that are assumed to be the same for all plants.

Figure 4.11 shows the closed-loop sensitivity Bode diagram for a number of randomly sampled plants equipped with LTI controllers. In this figure, time-varying parameters are kept fixed just like Figure 4.8, and thus, this figure is also a lower bound on the L_2-gain of the closed-loop plant. The comparison between Figure 4.8 and Figure 4.11 shows an improvement in terms of the peak and the cross-over frequency of the sensitivity transfer function when switching MRC are used.

The time response of the time-varying plant equipped with switching MRC and the corresponding control action are shown in Figure 4.12(a) and Figure 4.12(b), respectively. For this simulation, the time-varying parameters are assumed to vary according to Figure 4.12(c). To identify the active

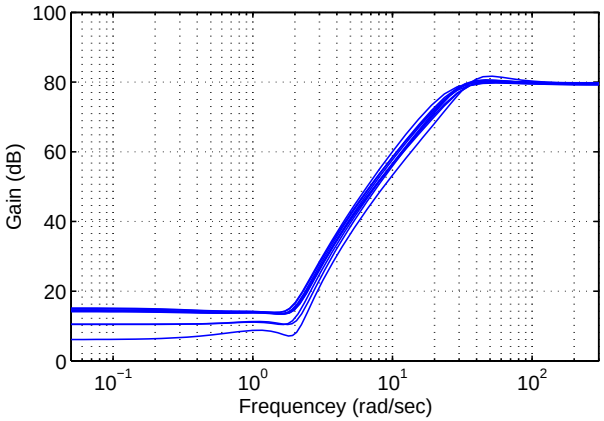


Figure 4.10: Bode diagram of switching MRC.

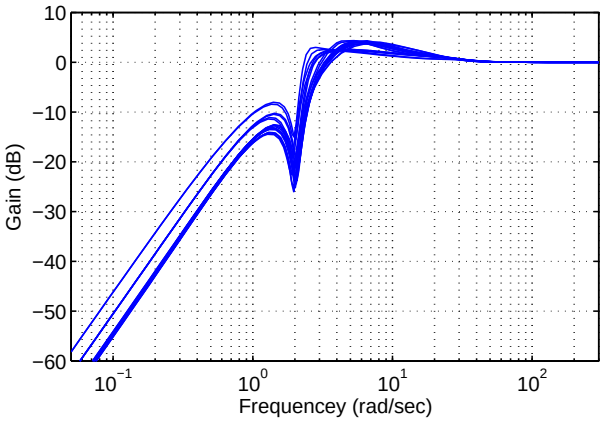
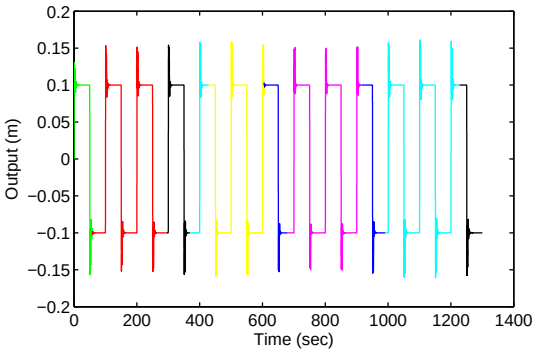


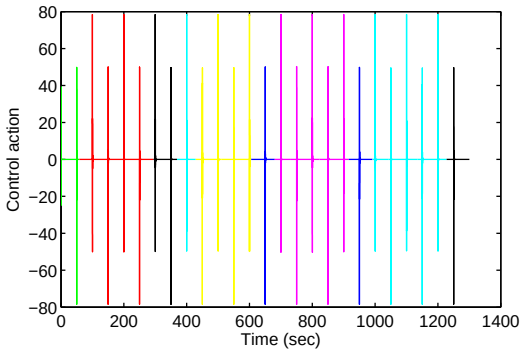
Figure 4.11: The closed-loop sensitivity Bode diagram for switching MRC.

controller at each time instant, various line colors are used in Figures 4.12(a) and 4.12(c). The hysteresis switching scheme can be easily seen in Figure 4.12(c). The time response of the system shows a satisfactory settling time of 13 seconds.

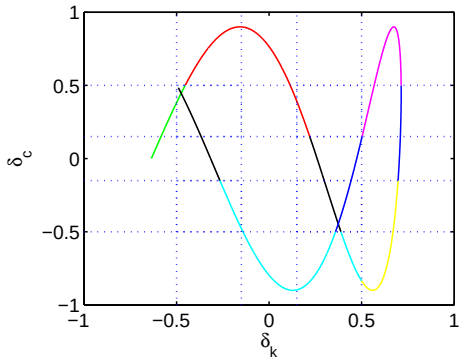
As far as the computational time is concerned, the overall optimization for single robust control and switching MRC took respectively 290 seconds and 1270 seconds on a personal computer with a 2.4 GHz Intel® Core™ 2 Duo Processor.



(a) The closed-loop response.



(b) The control action.



(c) The parameter variations.

Figure 4.12: The time-domain simulation of the switching MRC.

4.6 Conclusions

In this chapter, a parameter set division and switching MRC design method are proposed for plants with time-varying parameters. A user-specified robust L_2-gain is guaranteed with this controller. The main advantage of this method over previous similar methods, such as [78, 107], is the systematic parameter set division method. This method automatically determines the sufficient number of parameter set divisions to achieve a given robust L_2-gain performance, while the previous method requires the user to pre-specify parameter set divisions. Therefore, in practical examples where the designer needs to achieve a performance objective but has no idea on how to divide the parameter set, our method is a more natural choice than the ones of [107] and [78].

Chapter 5

Robust Finite-Time Tracking with Switched Controllers [1]

5.1 Introduction

This chapter proposes a robust switched controller design method that meets Objective (O2) of Section 1.3. The new notion of *robust finite-time tracking (RFTT)* is introduced that helps us to easily modify most of the currently available switched controller design methods for plants without uncertainty in order to apply them to uncertain switched systems. Roughly speaking, RFTT ensures that the distance between trajectories of an uncertain system and its nominal trajectory (i.e. the trajectory in the absence of uncertainty) is bounded at all times. The switched robust finite-time tracking controllers (RFTTC) design problem is formulated as an optimization problem, to be solved by a nonsmooth optimization method.

This chapter is organized as follows. Section 5.2 formulates a switched RFTTC design problem. Section 5.3 provides sufficient conditions for a closed-loop system in order to achieve RFTT property. Based on these condi-

[1] This chapter is based on the following articles: E. Azadi Yazdi and R. Nagamune, *"Switched tracking controller design for uncertain systems"*, 2010 ASME Dynamic Systems and Control Conference, Boston, MA, and E. Azadi Yazdi and R. Nagamune, *"Robust finite-time tracking with switched controllers"*, submitted for publication.

tions, a controller design optimization problem is formulated in Section 5.4. To the controller design optimization problem, a nonsmooth optimization technique is applied in Section 5.5. An application of the proposed method is presented in Section 5.6, and numerically validated in Section 5.7.

5.2 Problem formulation

5.2.1 Uncertain plant

We consider a LTI plant expressed by the state space equation:

$$P(\Delta A) : \begin{bmatrix} \dot{x}(t) \\ y(t) \end{bmatrix} = \begin{bmatrix} A + \Delta A & B_1 & B_2 \\ C & D_1 & D_2 \end{bmatrix} \begin{bmatrix} x(t) \\ w(t) \\ u(t) \end{bmatrix}, \tag{5.1}$$

where $\Delta A \in \mathbf{\Delta}$ is a real parametric uncertainty, $\mathbf{\Delta} \subset \mathbb{R}^{n \times n}$ is a bounded set of structured real matrices, $x(t) \in \mathbb{R}^n$ is the state vector, and as shown in Figure 5.1, $w(t) \in \mathbb{R}^{n_w}$ the reference input vector, $u(t) \in \mathbb{R}^{n_u}$ the control input vector, $y(t) \in \mathbb{R}^{n_y}$ the measurement vector, and all the matrices in (5.1) are of appropriate dimensions.

For this plant, we assume the following in this chapter.

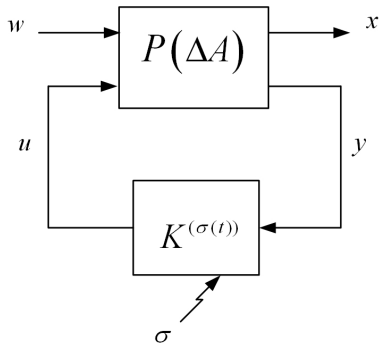


Figure 5.1: An uncertain plant $P(\Delta A)$ with a switched controller $K^{(m)}$.

(A1) $\Delta A = 0$ belongs to $\mathbf{\Delta}$.

(A2) $D_2 = 0$.

(A3) (A, B_2) is stabilizable and (A, C) is detectable.

Assumptions (A1) and (A2) are without loss of generality, and are for simplifying the calculations. $P(0_{n\times n})$ is called a *nominal plant* in this chapter. Assumption (A3) is necessary and sufficient for the existence of a stabilizing dynamic output-feedback controller for the nominal plant.

5.2.2 Switched controller

To the uncertain plant $P(\Delta A)$, we connect a switched controller $K^{(\sigma(t))}$, characterized by a set of LTI dynamic output feedback controllers $\mathbb{K} := \{K^{(m)}\}_{m=1}^{M}$ and an external switching signal $\sigma(t)$. We assume that all of the controllers in $\mathbb{K}$ are of the same order n_c, and described by

$$K^{(m)}: \quad \begin{bmatrix} \dot{x}_m(t) \\ u(t) \end{bmatrix} = \mathcal{K}^{(m)} \begin{bmatrix} x_m(t) \\ y(t) \end{bmatrix}, \tag{5.2}$$

where $\mathcal{K}^{(m)} \in \mathbb{R}^{(n_c+n_u)\times(n_c+n_y)}$ consists of (A, B, C, D)-matrices of m-th controller $K^{(m)}$. To indicate a switching rule, we utilize a map $\sigma \in \Sigma$, where

$$\Sigma := \{\sigma : \mathbb{R}^+ \to \mathbb{Z}\,[1, M] \mid \sigma \text{ is piecewise constant}\}. \tag{5.3}$$

$\mathbb{R}^+$ and $\mathbb{Z}\,[1, M]$ respectively represent the time and the mode index. For $\sigma \in \Sigma$, define a set of switching instances $T_{switch} := \{t \mid \sigma(t) \text{ is discontinuous}\}$.

5.2.3 Closed-loop switched system

Using the feedback interconnection as shown in Figure 5.1, the closed-loop system equations, with the closed-loop state vector $x_{cl}(t) := \left[x^T(t),\ x_\sigma^T(t)\right]^T$,

are given by

$$\dot{x}_{cl}(t) = \begin{bmatrix} \mathcal{A}(\mathcal{K}^{(\sigma)}, \Delta A) & \mathcal{B}(\mathcal{K}^{(\sigma)}) \end{bmatrix} \begin{bmatrix} x_{cl}(t) \\ w(t) \end{bmatrix} , \forall t \notin T_{switch}, \tag{5.4}$$

$$x_{cl}(t) = \begin{bmatrix} x(t^-) \\ R(\sigma(t), \sigma(t^-)) x_{cl}(t^-) \end{bmatrix} , \forall t \in T_{switch}, \tag{5.5}$$

where $x(t^-) = \lim_{\varepsilon \to 0^+} x(t - \varepsilon)$, and $\mathcal{A}(\mathcal{K}^{(\sigma)}, \Delta A)$ and $\mathcal{B}(\mathcal{K}^{(\sigma)})$ are the closed-loop system matrices.[2]

The function $R(\sigma(t), \sigma(t^-)) \in \mathbb{R}^{n_c \times (n+n_c)}$ is called a *reset map*. Given a controller set $\mathbb{K}$ and a reset map R, the closed-loop state trajectory x_{cl}, which is the solution to (5.4) and (5.5), depends on: the initial condition $x_{cl}(0) := x_{cl}^0$, the input vector $w(t)$, the switching signal $\sigma(t)$, and the uncertainty ΔA. Throughout this chapter, we will explicitly show this dependency as $x_{cl}^{\left[x_{cl}^0, w, \sigma, \Delta A\right]}$.

5.2.4 Robust finite-time tracking

Let us introduce the notion of RFTT for a dynamic system that consists of a single mode.

Definition: An uncertain system $\dot{x} = (A + \Delta A)x + Bw$ with a given input $w_{ref}(t)$ has the *RFTT* property with respect to a real positive pair (r_x, r_{x_0}) in the time horizon $[t_0, t_0 + t_f]$, if

$$\|x(t) - x^{nom}(t)\| \leq r_x, \ \forall \left\|x^0 - x^{nom,0}\right\| \leq r_{x_0}, \\ \forall \Delta A \in \mathbf{\Delta}, \ \forall t \in [t_0, t_0 + t_f], \tag{5.6}$$

where $x^{nom,0}$ is the initial condition of the trajectory $x^{nom}(t)$ of the nominal system and x^0 is the initial condition of the trajectory $x(t)$ of the uncertain system.

[2]Without loss of generality we assume that $x_{cl}(t)$ is continuous from above at every point, i.e., $x_{cl}(t) = \lim_{\varepsilon \to 0^+} x_{cl}(t + \varepsilon)$.

The above definition is similar to the finite-time stability definition in [38], in the sense that both RFTT and finite-time stability concern the boundedness of the transient trajectories and not the convergence of them. This is in contrast with the Lyapunov stability and robust tracking, where the convergence of the trajectories is the main consideration.

Definition: The switched system (5.4) and (5.5) satisfies *RFTT* if the following condition holds for any arbitrary switching signal $\sigma \in \Sigma$.

$$\begin{aligned} \|x_{cl}(t) - x_{cl}^{nom}(t)\| \leq r_x,\ \forall \left\|x_{cl}^0 - x_{cl}^{nom,0}\right\| \leq r_{x_0}, \\ \forall \Delta A \in \mathbf{\Delta},\ \forall t \in [t_0, t_0 + t_f], \end{aligned} \tag{5.7}$$

where

$$\begin{aligned} x_{cl}^{nom}(t) &:= x_{cl}^{\left[x_{cl}^{nom,0}, w_{ref}, \sigma, 0\right]}(t), \\ x_{cl}(t) &:= x_{cl}^{\left[x_{cl}^{0}, w_{ref}, \sigma, \Delta A\right]}(t). \end{aligned} \tag{5.8}$$

Remark. Without loss of generality, we assume $t_0 = 0$.

An illustration of a plant equipped with a controller set of three members, that has RFTT property is depicted in Figure 5.2. For the given switching signal $\sigma_1(t)$ in Figure 5.2(a), the perturbed trajectory stays in a sphere of radius r_x around the nominal trajectory for all positive times.

Let us formulate a *switched RFTTC synthesis problem* as follows.

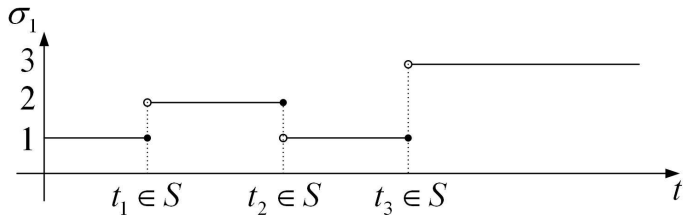
Problem 5.2.1. *Given an uncertain LTI plant (5.1) with the uncertainty set $\mathbf{\Delta}$ and a reference input $w_{ref} \in L_\infty^{n_w}$, design a specified number "M" of controllers $\mathbb{K}$ and a reset map R such that, for any switching signal $\sigma \in \Sigma$ and uncertainty $\Delta A \in \mathbf{\Delta}$, the closed-loop system has the RFTT property with respect to a given (r_x, r_{x_0}).*

Problem 5.2.1 imposes a constraint on the distance between the nominal reference trajectory and the perturbed trajectories. In some application, it is not only desirable to have all the trajectories in a bounded neighborhood

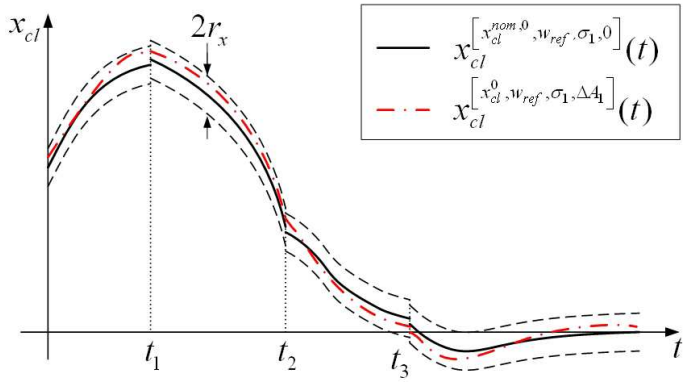
of the nominal trajectory but also favorable to have the nominal trajectory in a bounded neighborhood of a given desired trajectory $x_{des}(t)$, that is

$$\|x_{cl}^{nom} - x_{des}\|_{\infty} \leq \gamma, \tag{5.9}$$

where γ is a positive real number. In such problems, the desired trajectory $x_{des}(t)$ can be considered as an external signal to the plant as shown in Figure 5.3. For the new plant $P_e(\Delta A)$ shown in Figure 5.3, the described problem can be decomposed into a RFTT problem and a nominal L_∞ problem $\frac{1}{\gamma}\|x_e^{nom}\|_{\infty} \leq 1$. Following the path of [23], the nominal L_∞ can be easily combined with our method. This will be discussed in more details in the example section.



(a) Switching signal $\sigma_1(t)$.



(b) Closed-loop state response $x_{cl}(t)$.

Figure 5.2: Switched RFTT problem illustration.

5.3 Main results

We will derive sufficient conditions that a set of controllers $\mathbb{K}$ and a reset map R have only to meet, in order for the resulting switching controller to solve Problem 5.2.1. The conditions for $\mathbb{K}$ and R are derived independently as follows.

Condition (1)- RFTT in each mode: The RFTT problem is addressed in periods of time with a constant σ (i.e. $t \in \{[0, t_f] - T_{switch}\}$). Sufficient conditions for a set of controllers $\mathbb{K}$ are derived such that starting in a sphere of radius r_x centered with nominal initial conditions, trajectories of the uncertain system remain in a sphere of radius r_x around the nominal trajectory.

Condition (2)- RFTT during switching instances: A sufficient condition for the reset map R is derived which ensures RFTT at switching instances (i.e. $t \in T_{switch}$). This condition guarantees that, at any switching instance, initial conditions of the next mode are in a sphere of radius r_x of the nominal initial conditions, provided that in the current mode the final states are in a sphere of radius r_x centered with the

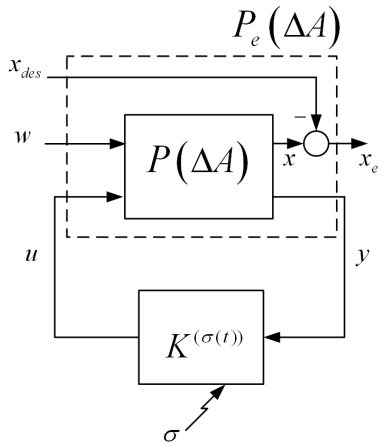


Figure 5.3: A variation of the RFTT problem.

final nominal states.

With these two conditions for a switching controller, it is guaranteed that trajectories of the closed-loop uncertain system with the switching controller stays in a sphere of radius r_x of the nominal trajectory for all $t \in \{\{[0, t_f] - T_{switch}\} \bigcup T_{switch}\} = [0, t_f]$.

Remark. *To derive condition (1), we use a method similar to [59]. In [59], a RFTTC method is developed which uses an observer-based output feedback controller structure. However, our method covers the general class of output feedback controller structure and can render less conservative controllers relative to the method of [59].*

5.3.1 Contraction mapping and its property

Contraction maps and their properties will be used to derive Condition (1). Therefore, let us introduce the definition of a contraction mapping and a useful lemma for these mappings. Letting X be a normed linear space, a contraction mapping can be defined as follows.

Definition: An operator $\Phi : X \to X$ is called a *contraction mapping* in $\mathbb{S}(\hat{x}, r) = \{x : \|x - \hat{x}\| \leq r\}$ if there exists a constant $0 < \theta < 1$ such that

$$\|\Phi x_1 - \Phi x_2\| \leq \theta \|x_1 - x_2\| \quad \forall\ x_1, x_2 \in \mathbb{S}(\hat{x}, r), \tag{5.10}$$

where θ is the *contraction factor* of Φ in $\mathbb{S}(\hat{x}, r)$.

Lemma 5.3.1. *[74] Let $(X, \|\cdot\|)$ be a Banach space, and let $\mathbb{S}(\hat{x}, r) = \{x | \|x - \hat{x}\| \leq r\}$. Suppose $\Phi : X \to X$ be an operator satisfying the following conditions:*

(i) Φ is a contraction on $\mathbb{S}(\hat{x}, r)$,

(ii) $\|\Phi(\hat{x}) - \hat{x}\| \leq r(1 - \theta)$, where θ satisfies (5.10).

Then Φ maps $\mathbb{S}(\hat{x}, r)$ back to itself, and there is a unique fixed point $x^* \in \mathbb{S}(\hat{x}, r)$ such that $\Phi(x^*) = x^*$.

5.3.2 Condition (1): RFTT in each mode

In this step, Lemma 5.3.1 is used to show that in each mode the closed-loop uncertain trajectories remain in an r_x sphere of the reference trajectory if its initial state is in an r_x sphere of the reference initial state. We choose $X := L_{t_f}^{(n+n_c)}$ (that is a Banach space), $x := x_{cl}$, $\hat{x} := x_{cl}^{nom}$, and $r := r_x$ to fit our problem to that lemma. Next, we will clarify what the operator Φ in Lemma 5.3.1 would be in our problem setting.

Suppose the switching signal is an arbitrary constant value $\sigma(t) = \hat{\sigma}$, i.e. $t \in \{[0, t_f] - T_{switch}\}$. Let us rewrite the closed-loop system equation (5.4) as

$$\dot{x}_{cl}(t) = \mathcal{A}(\mathcal{K}^{(\hat{\sigma})}, 0)x_{cl}(t) + \Delta\mathcal{A}x_{cl}(t) + \mathcal{B}(\mathcal{K}^{(\hat{\sigma})})w(t), \tag{5.11}$$

where

$$\Delta\mathcal{A} := \begin{bmatrix} \Delta A & 0 \\ 0 & 0_{n_c} \end{bmatrix}. \tag{5.12}$$

If all the eigenvalues of the matrix $\mathcal{A}(\mathcal{K}^{(\hat{\sigma})}, 0)$ are in the open left-half complex plane (i.e. each controller stabilizes the nominal plant), then the solution to (5.11) can be written in the operator form as[3]

$$x_{cl}(t) = \mathcal{L}_{\hat{\sigma}}\Delta\mathcal{A}x_{cl}(t) + \mathcal{L}_{\hat{\sigma}}\mathcal{B}(\mathcal{K}^{(\hat{\sigma})})w(t) + e^{\mathcal{A}(\mathcal{K}^{(\hat{\sigma})},0)t}x_{cl}^0, \tag{5.13}$$

where the operator $\mathcal{L}_{\hat{\sigma}}$ is defined as a linear mapping from $L_{t_f}^{(n+n_c)}$ into itself:

$$(\mathcal{L}_{\hat{\sigma}}v)(t) := \int_0^t e^{\mathcal{A}(\mathcal{K}^{(\hat{\sigma})},0)(t-\tau)}v(\tau)d\tau. \tag{5.14}$$

[3]Equation (5.11) can be retrieved by taking the derivative of both sides of (5.13).

Equation (5.13) can be further transformed into the following form:

$$\begin{aligned} x_{cl}(t) =& \mathcal{L}_{\hat{\sigma}}\mathcal{B}(\mathcal{K}^{(\hat{\sigma})})w(t) + e^{\mathcal{A}(\mathcal{K}^{(\hat{\sigma})},0)t}x_{cl}^{nom,0} \\ &+ \mathcal{L}_{\hat{\sigma}}\Delta\mathcal{A}x_{cl}(t) \\ &+ e^{\mathcal{A}(\mathcal{K}^{(\hat{\sigma})},0)t}(x_{cl}^{0} - x_{cl}^{nom,0}), \end{aligned} \tag{5.15}$$

where the first, second, and third terms of the right-hand side represent respectively the nominal trajectory, perturbations due to parametric uncertainty, and perturbations due to uncertainty in initial conditions.

The right-hand side of (5.15) is denoted by Φx_{cl}. The operator Φ maps $L_{t_f}^{(n+n_c)}$ into itself. Since $L_{t_f}^{(n+n_c)}$ is a Banach space, we can apply Lemma 5.3.1 to the operator Φ. If conditions of Lemma 5.3.1 hold for the defined Φ, the operator Φ maps $\mathbb{S}(x_{cl}^{\left[x_{cl}^0, w_{ref}, \hat{\sigma}, \Delta A\right]}, r_x)$ into itself. This means that solutions of (5.15) stay in $\mathbb{S}(x_{cl}^{nom}, r_x)$ for all initial conditions in $\mathbb{S}(x_{cl}^{nom,0}, r_x)$. Therefore, each individual mode satisfies RFTT with respect to (r_x, r_x), provided that conditions of Lemma 5.3.1 hold.

To verify conditions of Lemma 5.3.1, first we have to show that Φ is a contraction map. Let $\bar{x}_{cl}$ and $\tilde{x}_{cl}$ be two arbitrary closed-loop state trajectories starting from $\bar{x}_{cl}^0$ and $\tilde{x}_{cl}^0$ in a sphere $\mathbb{S}(x_{cl}^{nom,0}, r_x)$ respectively. By definition, the contraction factor θ is given by

$$\begin{aligned} \theta &= \sup_{\Delta A,\ \bar{x}_{cl},\ \tilde{x}_{cl}} \frac{\|\Phi\bar{x}_{cl} - \Phi\tilde{x}_{cl}\|_{t_f}}{\|\bar{x}_{cl} - \tilde{x}_{cl}\|_{t_f}} \\ &= \sup_{\Delta A,\ \bar{x}_{cl},\ \tilde{x}_{cl}} \frac{\|\mathcal{L}_{\hat{\sigma}}\Delta\mathcal{A}(\bar{x}_{cl} - \tilde{x}_{cl})\|_{t_f}}{\|\bar{x}_{cl} - \tilde{x}_{cl}\|_{t_f}}. \end{aligned} \tag{5.16}$$

where $\bar{x}_{cl} := \bar{x}_{cl}^{\left[\bar{x}_{cl}^0, w_{ref}, \hat{\sigma}, 0\right]}$ and $\tilde{x}_{cl} := \tilde{x}_{cl}^{\left[\tilde{x}_{cl}^0, w_{ref}, \hat{\sigma}, 0\right]}$ are two closed-loop trajectories that satisfy $\bar{x}_{cl}, \tilde{x}_{cl} \in \mathbb{S}(x_{cl}^{nom}, r_x)$. The above contraction factor is non-zero whenever $\bar{x}_{cl}^0 \neq \tilde{x}_{cl}^0$. For a map with a non-zero contraction factor θ, if the condition (ii) of Lemma 5.3.1 holds, the condition (i) is automatically met (i.e. $0 < \theta < 1$). Therefore, if Φ satisfies the second condition of

Lemma 5.3.1, we can apply the lemma to the operator Φ.

An upper bound for the left-hand side of condition (ii) in Lemma 5.3.1 is given by

$$\begin{aligned}\|\Phi x_{cl}^{nom} - x_{cl}^{nom}\|_{t_f} &\leq \|\mathcal{L}_{\hat{\sigma}}\Delta\mathcal{A}x_{cl}^{nom}\|_{t_f} \\ &+ \left\|(x_{cl}^{0} - x_{cl}^{nom,0})\right\|_{t_f} \sup_{\|\delta x_0\|_\infty \leq 1} \left\| e^{\mathcal{A}(\mathcal{K}^{(\hat{\sigma})},0)t}\delta x_0 \right\|_{t_f}. \end{aligned} \tag{5.17}$$

For a time-varying matrix $M(t)$ and a constant vector v[4],

$$\|M(t)v\|_{t_f} \leq \|v\|_\infty \sup_{\|u\|_\infty \leq 1} \|M(t)u\|_{t_f}. \tag{5.18}$$

The initial conditions are in a sphere of radius r_x around the nominal initial conditions (i.e., $\|(x_{cl}^{0} - x_{cl}^{nom,0})\|_\infty \leq r_x$). Therefore, (5.17) can be written as

$$\begin{aligned}\|\Phi x_{cl}^{nom} - x_{cl}^{nom}\|_{t_f} &\leq \|\mathcal{L}_{\hat{\sigma}}\Delta\mathcal{A}x_{cl}^{nom}\|_{t_f} \\ &+ r_x \sup_{\|\delta x_0\|_\infty \leq 1} \left\| e^{\mathcal{A}(\mathcal{K}^{(\hat{\sigma})},0)t}\delta x_0 \right\|_{t_f}. \end{aligned} \tag{5.19}$$

Supposing that the right-hand side of (5.19) is bounded by $r_x(1-\theta)$ with θ in (5.16), the condition (ii) of Lemma 5.3.1 holds. The supposition can be expressed as

$$\begin{aligned}&\|\mathcal{L}_{\hat{\sigma}}\Delta\mathcal{A}x_{cl}^{nom}\|_{t_f} \\ &+ \left\|(x_{cl}^{0} - x_{cl}^{nom,0})\right\|_{\infty} \sup_{\|\delta x_0\|_\infty \leq 1} \left\| e^{\mathcal{A}(\mathcal{K}^{(\hat{\sigma})},0)t}\delta x_0 \right\|_{t_f} \\ &\leq r_x \left(1 - \sup_{\Delta A \in \mathbf{\Delta}} \frac{\|\mathcal{L}_{\hat{\sigma}}\Delta\mathcal{A}(\bar{x}_{cl} - \tilde{x}_{cl})\|_{t_f}}{\|\bar{x}_{cl} - \tilde{x}_{cl}\|_{t_f}}\right). \end{aligned} \tag{5.20}$$

Hence, a sufficient condition for the condition (ii) of Lemma 5.3.1 is given

[4]The inequality (5.18) is trivial, because if in the supremum we choose $u = v$ then the equality holds.

by

$$\eta_{un}(\mathcal{K}^{(\hat{\sigma})}) + \eta_{in}(\mathcal{K}^{(\hat{\sigma})}) + \eta_{con}(\mathcal{K}^{(\hat{\sigma})}) \leq 1, \tag{5.21}$$

where

$$\eta_{un}(\mathcal{K}^{(\hat{\sigma})}) := \frac{1}{r_x} \sup_{\Delta A \in \mathbf{\Delta}} \|\mathcal{L}_{\hat{\sigma}} \Delta \mathcal{A} x_{cl}^{nom}\|_{t_f}, \tag{5.22}$$

$$\eta_{in}(\mathcal{K}^{(\hat{\sigma})}) := \sup_{\|\delta x_0\|_\infty \leq 1} \left\| e^{\mathcal{A}(\mathcal{K}^{(\hat{\sigma})},0)t} \delta x_0 \right\|_{t_f}, \tag{5.23}$$

$$\eta_{con}(\mathcal{K}^{(\hat{\sigma})}) := \sup_{\bar{x}_{cl}^0, \tilde{x}_{cl}^0, \Delta A \in \mathbf{\Delta}} \frac{\|\mathcal{L}_{\hat{\sigma}} \Delta \mathcal{A}(\bar{x}_{cl} - \tilde{x}_{cl})\|_{t_f}}{\|\bar{x}_{cl} - \tilde{x}_{cl}\|_{t_f}}. \tag{5.24}$$

According to Lemma 5.3.1, if the above inequality holds and $\|x_{cl}^0 - x_{cl}^{nom}\|_\infty \leq r_x$, the operator Φ maps $\mathbb{S}(x_{cl}^{\left[x_{cl}^0, w_{ref}, \hat{\sigma}, \Delta A\right]}, r_x)$ into itself. Therefore, each individual mode satisfies RFTT with respect to (r_x, r_x) if it is stable and (5.21) holds.

5.3.3 Condition (2): RFTT during switching instances

In switched controllers, the initial state of each mode is a function of the final condition of the previous mode given by (5.5). If $\|R(\sigma_1, \sigma_2)\|_\infty \leq 1$, then following holds.

$$\|x_{cl}(t) - x_{cl}^{nom}(t)\| \leq \|x_{cl}(t^-) - x_{cl}^{nom}(t^-)\|, \tag{5.25}$$

for any $t \in T_{switch}$. Inequality (5.25) means that, if in a mode $\sigma(t^-)$, the final conditions are in a sphere of radius r_x of the nominal final conditions, then in the next mode $\sigma(t)$, the initial conditions are in a sphere of radius r_x of the nominal initial conditions.

The following theorem summarizes sufficient conditions for a set of LTI controllers $\mathbb{K}$ and a reset map R which are derived in Condition (1) and (2)

above.

Theorem 5.3.2. *A set of controllers* $\mathbb{K}$ *and a reset map* $R(\sigma_1, \sigma_2)$ *satisfying the following conditions solves Problem 5.2.1.*

(C1) all eigenvalues of $\mathcal{A}(\mathcal{K}^{(\hat{\sigma})}, 0)$ *have negative real parts,*

(C2) the following inequality constraints hold,

$$\eta_{un}(\mathcal{K}^{(\hat{\sigma})}) + \eta_{in}(\mathcal{K}^{(\hat{\sigma})}) + \eta_{con}(\mathcal{K}^{(\hat{\sigma})}) \leq 1, \quad \forall \hat{\sigma} \in \mathbb{Z}\,[1, M] \tag{5.26}$$

(C3) $\|R(\sigma_1, \sigma_2)\|_\infty \leq 1$, *for all* $\sigma_1 \neq \sigma_2 \in \mathbb{Z}\,[1, M]$.

5.4 Controller design

This section formulates an optimization problem to find a controller set $\mathbb{K}$ that satisfies conditions (C1) and (C2) of Theorem 5.3.2, and hence meets the controller related sufficient conditions in Theorem 5.3.2 for solving the RFTT problem.

For a given admissible radius r_x of the state trajectory deviation from a nominal trajectory, the following optimization problem takes into account simultaneously conditions (C1) and (C2) of Theorem 5.3.2.

$$\begin{aligned} &\min_{\mathcal{K}^{(1)},\ldots,\mathcal{K}^{(m)}} \max_{\hat{\sigma}\in\mathbb{Z}[1,M]} \left\{\eta_{un}(\mathcal{K}^{(\hat{\sigma})}) + \eta_{in}(\mathcal{K}^{(\hat{\sigma})}) + \eta_{con}(\mathcal{K}^{(\hat{\sigma})})\right\} \\ &\text{subject to } Re(\lambda_i(\mathcal{A}(\mathcal{K}^{(\hat{\sigma})}, 0))) < 0 \ \forall \hat{\sigma} \in \mathbb{Z}\,[1, M]\,, \ \forall i. \end{aligned} \tag{5.27}$$

If the optimum value of the cost function in (5.27) is less than one, the conditions (C1) and (C2) of Theorem 5.3.2 are met. The cost function of the optimization problem (5.27) is not a smooth function because of the appearance of infinity norms. Besides, it is nonconvex in general. A nonsmooth optimization technique can be used to find a local optimum of this optimization problem [9]. Next section explains the nonsmooth optimization method that has been used.

5.5 Numerical nonsmooth descent optimization

In this section, a similar method to [9] will be used in this section to solves the optimization problem (5.27). In (5.27), the cost function contains the maximum of several functions (i.e. $\max_{\hat{\sigma}}\{\eta_{un}(\mathcal{K}^{(\hat{\sigma})})+\eta_{in}(\mathcal{K}^{(\hat{\sigma})})+\eta_{con}(\mathcal{K}^{(\hat{\sigma})})\}$). In each optimization step, the mode $\hat{\sigma}$ that attains the maximum is called the *active mode*. Suppose $\mathcal{K}^{(\hat{\sigma})}$ is given. Let us define the cost function of (5.27) as $f_\infty(\mathcal{K}^{(\hat{\sigma})})$ given by

$$\begin{aligned} f_\infty(\mathcal{K}^{(\hat{\sigma})}) :=& \eta_{un}(\mathcal{K}^{(\hat{\sigma})}) + \eta_{in}(\mathcal{K}^{(\hat{\sigma})}) + \eta_{con}(\mathcal{K}^{(\hat{\sigma})}) \\ =& \frac{1}{r_x}\|\xi_{un}(\mathcal{K}^{(\hat{\sigma})}, t)\|_{t_f} + \|\xi_{in}(\mathcal{K}^{(\hat{\sigma})}, t)\|_{t_f} \\ &+ \frac{\|n(\mathcal{K}^{(\hat{\sigma})}, t)\|_{t_f}}{\|d(\mathcal{K}^{(\hat{\sigma})}, t)\|_{t_f}}, \end{aligned} \tag{5.28}$$

where controller-dependent time functions are defined by

$$\xi_{un}(\mathcal{K}^{(\hat{\sigma})}, t) := \mathcal{L}_{\hat{\sigma}}\Delta\mathcal{A}^*_{\xi_{un}} x_{cl}^{nom}(t), \tag{5.29}$$

$$\xi_{in}(\mathcal{K}^{(\hat{\sigma})}, t) := e^{\mathcal{A}(\mathcal{K}^{(\hat{\sigma})},0)t}\delta x_0^*, \tag{5.30}$$

$$n(\mathcal{K}^{(\hat{\sigma})}, t) := \mathcal{L}_{\hat{\sigma}}\Delta\mathcal{A}^*_n(\bar{x}^*_{cl}(t) - \tilde{x}^*_{cl}(t)), \tag{5.31}$$

$$d(\mathcal{K}^{(\hat{\sigma})}, t) := \bar{x}^*_{cl}(t) - \tilde{x}^*_{cl}(t). \tag{5.32}$$

Here, $\Delta\mathcal{A}^*_{\xi_{un}}$, δx_0^*, and $\bar{x}^*_{cl}(t)$, $\tilde{x}^*_{cl}(t)$, and $\Delta\mathcal{A}^*_n$ correspond to the maximizers of optimization problems in (5.22), (5.23) and (5.24), respectively. Note that the maximizers $\bar{x}_{cl} := \bar{x}_{cl}^{\left[\bar{x}^0_{cl}, w_{ref}, \hat{\sigma}, 0\right]}$ and $\tilde{x}_{cl} := \tilde{x}_{cl}^{\left[\tilde{x}^0_{cl}, w_{ref}, \hat{\sigma}, 0\right]}$ will be determined by performing a search for their initial conditions $\bar{x}^0_{cl}$ and $\tilde{x}^0_{cl}$, respectively.

The nonsmooth descent method to solve (5.27) can be summarized as follows.

Algorithm 5.5.1. *Consider the LTI plant (5.1) and the cost function (5.27).*

1. ***Choose a set of stabilizing initial controllers.***

2. ***Find the active mode in (5.27)****: Find the active mode $\hat{\sigma}$ for which the "max" function in (5.27) attains.*

3. ***Find a descent direction:*** *Find a descent direction for the cost function (5.27) with respect to controller parameters. This descent direction is denoted by $h(\mathcal{K}^{(\hat{\sigma})})$.*

4. ***Line search:*** *Perform a line search to find the optimal amount of movement in the descent directions $h(\mathcal{K}^{(\hat{\sigma})})$.*

$$\alpha^* := \arg\max_{\alpha} \left\{ f_\infty(\mathcal{K}^{(\hat{\sigma})}) - f_\infty(\mathcal{K}^{(\hat{\sigma})} + \alpha h(\mathcal{K}^{(\hat{\sigma})})) \right\} \tag{5.33}$$

The new controller parameter matrix is given by

$$\mathcal{K}^{(\hat{\sigma},new)} = \mathcal{K}^{(\hat{\sigma})} + \alpha^* h(\mathcal{K}^{(\hat{\sigma})}). \tag{5.34}$$

5. ***Stability check:*** *If the optimization constraint $Re(\lambda_i(\mathcal{A}(\mathcal{K}^{(\hat{\sigma})}, 0))) < 0$ is violated, decrease α^* until the stability constraint is satisfied.*

6. *Iterate Steps (2)-(5) until one of the following termination criteria is satisfied.*

$$\begin{aligned} \left\| h(\mathcal{K}^{(\hat{\sigma})}) \right\| &< \epsilon_1, \\ f_\infty(\mathcal{K}^{(\hat{\sigma})}) - f_\infty(\mathcal{K}^{(\hat{\sigma},new)}) &< \epsilon_2 f_\infty(\mathcal{K}^{(\hat{\sigma})}), \\ \left\| \alpha^* h(\mathcal{K}^{(\hat{\sigma})}) \right\| &< \epsilon_3 \left\| \mathcal{K}^{(\hat{\sigma})} \right\|, \end{aligned}$$

where $\epsilon_1, \epsilon_2, \epsilon_3 > 0$ are optimization parameters.

The next two subsections describe a method to compute the gradient of the cost function (5.27) with respect to the controller parameters, i.e. $h(\mathcal{K}^{(\hat{\sigma})})$ in Step (3) of the above algorithm.

5.5.1 Nonsmooth descent direction

In this section, we address the calculation of the descent direction $h(\mathcal{K}^{(\hat{\sigma})})$. In order to use the nonsmooth descent optimization technique, the Clarke subdifferential of the functional $f_\infty(\mathcal{K}^{(\hat{\sigma})})$ in (5.28) with respect to controller parameters should be calculated. For a vector-valued function $v : [0, t_f] \to L_\infty^n$, let us define a set $\mathbb{T}(v)$ by

$$\mathbb{T}(v) := \left\{(t, i) : |e_i^T v(t)| = \|v\|_{t_f}\right\} \subset [0, t_f] \times \mathbb{Z}[1, n], \tag{5.35}$$

where $e_i \in \mathbb{R}^n$ is the unit vector in i^{th} direction.

The following lemma provides the Clarke subdifferential of the functional $f_\infty(\mathcal{K}^{(\hat{\sigma})})$.

Lemma 5.5.2. *Assume $D_1 = 0$ in (5.1). The Clarke subdifferential of $f_\infty(\mathcal{K}^{(\hat{\sigma})})$ is given by*

$$\begin{aligned}\partial f_\infty(\mathcal{K}^{(\hat{\sigma})}) =& \frac{1}{r_x} \operatorname*{conv}_{(t,i)\in\mathbb{T}(\eta_{un})} \left\{\nabla_{\mathcal{K}^{(\hat{\sigma})}} e_i^T \xi_{un}(\mathcal{K}^{(\hat{\sigma})}, t)\right\} \\ &+ \operatorname*{conv}_{(t,i)\in\mathbb{T}(\eta_{in})} \left\{\nabla_{\mathcal{K}^{(\hat{\sigma})}} e_i^T \xi_{in}(\mathcal{K}^{(\hat{\sigma})}, t)\right\} \\ &+ \frac{1}{\|d\|_{t_f}^2}\Bigg(\|d\|_\infty \operatorname*{conv}_{(t,i)\in\mathbb{T}(n)} \left\{\nabla_{\mathcal{K}^{(\hat{\sigma})}} e_i^T n(\mathcal{K}^{(\hat{\sigma})}, t)\right\} \\ &\quad - \|n\|_{t_f} \operatorname*{conv}_{(t,i)\in\mathbb{T}(d)} \left\{\nabla_{\mathcal{K}^{(\hat{\sigma})}} e_i^T d(\mathcal{K}^{(\hat{\sigma})}, t)\right\}\Bigg).\end{aligned} \tag{5.36}$$

*where "*conv*" represents the convex combination of its arguments.*

Proof. Since the numerator $\|n(\mathcal{K}^{(\hat{\sigma})}, t)\|_{t_f}$ and denominator $\|d(\mathcal{K}^{(\hat{\sigma})}, t)\|_{t_f}$ of the last term in the expression (5.28) are Lipschitz, the Clarke subdifferential

of $f_\infty(\mathcal{K}^{(\hat{\sigma})})$ satisfies (see Proposition 2.3.14 in [31])

$$\begin{aligned}\partial f_\infty(\mathcal{K}^{(\hat{\sigma})}) \subseteq & \frac{1}{r_x}\partial\|\xi_{un}(\mathcal{K}^{(\hat{\sigma})},t)\|_{t_f} + \partial\|\xi_{in}(\mathcal{K}^{(\hat{\sigma})},t)\|_{t_f} \\ & + \frac{1}{\|d\|_{t_f}}\partial\|n(\mathcal{K}^{(\hat{\sigma})},t)\|_{t_f} \\ & - \frac{\|n\|_{t_f}}{\|d\|_{t_f}^2}\partial\|d(\mathcal{K}^{(\hat{\sigma})},t)\|_{t_f}. \end{aligned} \tag{5.37}$$

If, in addition, the numerator $\|n(\mathcal{K}^{(\hat{\sigma})},t)\|_{t_f}$ and denominator $\|d(\mathcal{K}^{(\hat{\sigma})},t)\|_{t_f}$ in (5.28) are regular at $\mathcal{K}^{(\hat{\sigma})}$, then the equality in (5.37) holds. Now we have to calculate subdifferentials $\partial\|\xi_{un}(\mathcal{K}^{(\hat{\sigma})},t)\|_{t_f}$, $\partial\|\xi_{in}(\mathcal{K}^{(\hat{\sigma})},t)\|_{t_f}$, $\partial\|n(\mathcal{K}^{(\hat{\sigma})},t)\|_{t_f}$, and $\partial\|d(\mathcal{K}^{(\hat{\sigma})},t)\|_{t_f}$. Let us calculate the subdifferential of the first term $\partial\|\xi_{un}(\mathcal{K}^{(\hat{\sigma})},t)\|_{t_f}$. Assuming that $\xi_{un}(\mathcal{K}^{(\hat{\sigma})},t)$ is a continuously differentiable function with respect to controller parameters, its Clarke subdifferential is given by

$$\partial\|\xi_{un}(\mathcal{K}^{(\hat{\sigma})},t)\|_{t_f} = \underset{(t,i)\in\mathbb{T}(\eta_{un})}{\operatorname{conv}} \left\{\nabla_{\mathcal{K}^{(\hat{\sigma})}} e_i^T \xi_{un}(\mathcal{K}^{(\hat{\sigma})},t)\right\}. \tag{5.38}$$

The Clarke subdifferentials of the rest of the terms in (5.37) can be derived in a similar way. Hence (5.37) can be rewritten as in (5.36). □

Remark. *In the above lemma, we assumed that $D_1 = 0$ for simplicity. It is possible to extend the results of this thesis to cases where $D_1 \neq 0$.*

Section 5.5.2 proposes a numerical method to compute gradients that appeared in (5.36). For the time being, let us assume that these gradients are already computed. Following the line in [9, 87], the following SDP can be formulated to find the nonsmooth descent direction at point $\mathcal{K}^{(\hat{\sigma})}$:

$$\begin{aligned}\max_{\lambda_j\geq 0,\sum\lambda_j=1} \sum_{(t_j,i_j)\in T_e(\mathcal{K}^{(\hat{\sigma})})} \Big(&\lambda_j\left(f(\mathcal{K}^{(\hat{\sigma})},t_j,i_j) - f_\infty(\mathcal{K}^{(\hat{\sigma})})\right) \\ & - \frac{1}{2}\beta\|h(\mathcal{K}^{(\hat{\sigma})})\|_2^2\Big), \end{aligned} \tag{5.39}$$

where $\beta > 0$ is an optimization parameter to be specified by the user, and

$$\begin{aligned} T_e(\mathcal{K}^{(\hat{\sigma})}) :=& \mathbb{T}(\xi_{un}(\mathcal{K}^{(\hat{\sigma})}, t)) \cup \mathbb{T}(\xi_{in}(\mathcal{K}^{(\hat{\sigma})}, t)) \cup \\ & \mathbb{T}(n(\mathcal{K}^{(\hat{\sigma})}, t)) \cup \mathbb{T}(d(\mathcal{K}^{(\hat{\sigma})}, t)), \end{aligned} \tag{5.40}$$

$$\begin{aligned} f(\mathcal{K}^{(\hat{\sigma})}, t_j, i_j) :=& \frac{1}{r_x} e_{i_j}^T \eta_{un}(\mathcal{K}^{(\hat{\sigma})}, t_j) + e_{i_j}^T \eta_{in}(\mathcal{K}^{(\hat{\sigma})}, t_j) \\ & + \frac{e_{i_j}^T n(\mathcal{K}^{(\hat{\sigma})}, t_j)}{e_{i_j}^T d(\mathcal{K}^{(\hat{\sigma})}, t_j)}, \end{aligned} \tag{5.41}$$

$$h(\mathcal{K}^{(\hat{\sigma})}) := -\frac{1}{\beta} \sum_j \lambda_j \Phi_{\mathcal{K}^{(\hat{\sigma})}}(t_j, i_j), \tag{5.42}$$

and $\Phi_{\mathcal{K}^{(\hat{\sigma})}}(t_j, i_j)$ is an element of set $\partial f_\infty(\mathcal{K}^{(\hat{\sigma})})$ that corresponds to (t_j, i_j).

5.5.2 The gradient calculations

This section explains a method to calculate the gradients that appeared in Lemma 5.5.2. Since analytical expressions to the gradients that appeared in Clarke subdifferential (5.36) cannot be found, we present a computational method to compute the gradients. For simplicity of the presentation, we assume that the output-feedback controllers are static, but the results can be extended to dynamic output-feedback controllers. The following lemma introduces a formula for computing the first gradient in (5.36). A similar method can be used to compute other gradient expressions in (5.36).

Lemma 5.5.3. *The (k, l)-element of the gradient $\nabla_{\mathcal{K}^{(\hat{\sigma})}} e_i^T \xi_{un}(\mathcal{K}^{(\hat{\sigma})}, t)$ is given by*

$$\left[\nabla_{\mathcal{K}^{(\hat{\sigma})}} e_i^T \xi_{un}(\mathcal{K}^{(\hat{\sigma})}, t)\right]_{kl} := e_i^T \frac{\partial \hat{z}}{\partial \mathcal{K}_{kl}^{(\hat{\sigma})}}, \tag{5.43}$$

where $\frac{\partial \hat{z}}{\partial \mathcal{K}_{kl}^{(\hat{\sigma})}}$ is a solution to the following system of ordinary differential equa-

tions with zero initial conditions.

$$\frac{d}{dt}\begin{bmatrix}\frac{\partial \hat{z}}{\partial \mathcal{K}^{(\hat{\sigma})}}(t)\\ \frac{\partial z}{\partial \mathcal{K}^{(\hat{\sigma})}}(t)\end{bmatrix} = \begin{bmatrix}A+B_2\mathcal{K}^{(\hat{\sigma})}C & \Delta\mathcal{A}^*_{\xi_{un}}\\ 0 & A+B_2\mathcal{K}^{(\hat{\sigma})}C\end{bmatrix}\begin{bmatrix}\frac{\partial \hat{z}}{\partial \mathcal{K}^{(\hat{\sigma})}}(t)\\ \frac{\partial z}{\partial \mathcal{K}^{(\hat{\sigma})}}(t)\end{bmatrix} + \begin{bmatrix}B_2e_ke_l^TC\hat{z}(t)\\ B_2e_ke_l^TCz(t)+B_2e_ke_l^TD_1w_{ref}(t)\end{bmatrix}, \tag{5.44}$$

where

$$z(t) = x_{cl}^{nom}(t), \tag{5.45}$$

$$\hat{z}(t) = \mathcal{L}_{\hat{\sigma}}\Delta\mathcal{A}^*_{\xi_{un}}x_{cl}^{nom}. \tag{5.46}$$

Proof. Consider the system in Figure 5.4. It can be shown that if the input signal w of Figure 5.4 is chosen exactly same as the input signal w_{ref} of the original LTI plant (5.1), then (5.45) and (5.46) hold.

Substituting the definition (5.29) of $\xi_{un}(\mathcal{K}^{(\hat{\sigma})},t)$ into the gradient expression $\nabla_{\mathcal{K}^{(\hat{\sigma})}}e_i^T\xi_{un}(\mathcal{K}^{(\hat{\sigma})},t)$, we have

$$\nabla_{\mathcal{K}^{(\hat{\sigma})}}e_i^T\xi_{un}(\mathcal{K}^{(\hat{\sigma})},t) = e_i^T\nabla_{\mathcal{K}^{(\hat{\sigma})}}\hat{z}(t). \tag{5.47}$$

Now, using a graphical approach, we will show that $\frac{\partial \hat{z}}{\partial \mathcal{K}_{ij}^{(\hat{\sigma})}}$ is the solution of the set of differential equations (5.44).

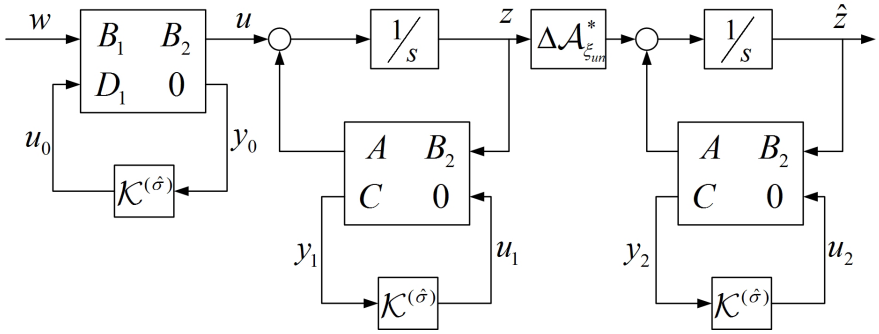


Figure 5.4: Equivalent plant for gradient calculations.

Differentiating all the signals and blocks of Figure 5.4 with respect to $\mathcal{K}_{ij}^{(\hat{\sigma})}$, Figure 5.5 can be obtained. It can be easily verified that (5.44) is the governing differential equation of the block diagram Figure 5.5. The states of the plant (5.44), or equivalently Figure 5.5, should be calculated numerically to find the gradient $\nabla_{\mathcal{K}^{(\hat{\sigma})}} e_i^T \xi_{un}(\mathcal{K}^{(\hat{\sigma})}, t)$. □

Remark. *The method that is used in this section to compute the gradient of an output signal is similar to the one in iterative feedback tuning (IFT) [54], [53]. Our approach here differs in the performance cost function and the nonsmooth optimization algorithm.*

5.6 A sample extension: combination of switched nominal stability and RFTT

The proposed switching controller design method for the RFTT problem can be used to extend available nominal controller design methods to robust controller design methods. To show the expediency of the method, this section describes one extension, where the nominal controller design method is adopted from Hespanha and Morse [51]. The following lemma summarizes the main result of [51].

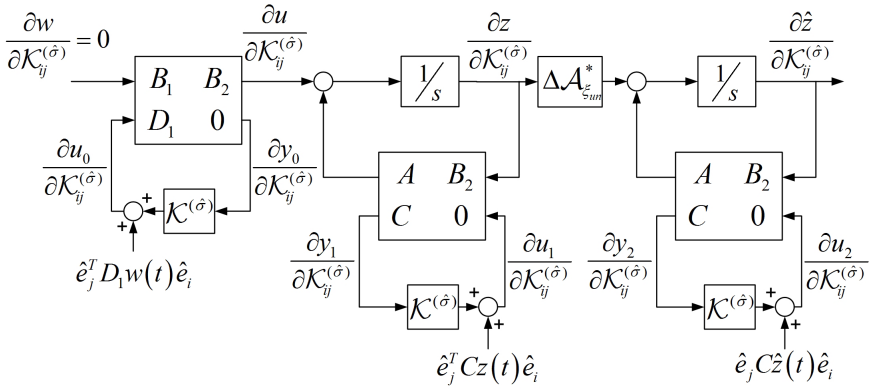


Figure 5.5: Derivative of the equivalent plant with respect to $\mathcal{K}_{ij}^{(\hat{\sigma})}$.

Lemma 5.6.1. *Given a set of stabilizing controllers $\tilde{\mathbb{K}}$ for a nominal LTI plant P, there exist a controller set $\mathbb{K}$ whose elements are realizations of elements in $\tilde{\mathbb{K}}$ and a reset map $R(\sigma_1, \sigma_2)$ that exponentially stabilize the plant P for any arbitrary switching signal.*

The extension of this method to uncertain systems with our method is straightforward, because the condition of the above lemma (i.e. nominal stability) is already met in our controller design optimization problem (5.27). The extension consists of the following three steps:

(1) Solve the optimization problem (5.27) and obtain a set of RFTTC $\tilde{\mathbb{K}}$.

(2) Apply the method of [51] on the controller set $\tilde{\mathbb{K}}$ and obtain a new set of realizations $\mathbb{K}$ and a reset map $R(\sigma_1, \sigma_2)$.

(3) If condition (C3) of Theorem 5.3.2 is not satisfied, normalize the computed reset map:

$$R(\sigma_1, \sigma_2) := \frac{R(\sigma_1, \sigma_2)}{\|R(\sigma_1, \sigma_2)\|_\infty}. \tag{5.48}$$

Step (3) in the above procedure is not a restrictive condition on the result of [51], because it can be easily shown that the normalization does not affect the nominal stability condition (Equation (37) in [51]).

5.7 An inverted pendulum example

In this section, we apply the proposed switched RFTTC design method to an inverted pendulum. The response of the closed-loop system with this controller is compared to a robust controller that does not take switching into account.

The control objective in the inverted pendulum system (Figure 5.6) is to hold the pendulum in a perpendicular position. The force F is exerted by a DC motor. The physical parameters of the DC motor are the conversion

gain ρ from the motor output torque to the force F and the motor back EMF constant c_a.

After linearizing the system dynamics around the unstable perpendicular equilibrium point, the system dynamics can be described as

$$\begin{cases} \dot{\boldsymbol{x}} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -L\omega_1^2 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \omega_2^2 & 0 \end{bmatrix} \boldsymbol{x} + \begin{bmatrix} 0 \\ \gamma \\ 0 \\ -\gamma/L \end{bmatrix} u, \\ y = \begin{bmatrix} 1 & 0 & 1 & 0 \end{bmatrix} \boldsymbol{x}, \end{cases} \tag{5.49}$$

where $\boldsymbol{x} = \begin{bmatrix} x & \dot{x} & \phi & \dot{\phi} \end{bmatrix}^T$ is the state vector, u is the input current to the motor, and ω_1, ω_2 an γ are defined as

$$\omega_1 := \sqrt{\frac{m_p g}{L m_c}}, \quad \omega_2 := \sqrt{\frac{(m_p + m_c) g}{L m_c}}, \quad \gamma := \frac{c_a}{\rho m_c}. \tag{5.50}$$

In this model, the nominal values of the parameters are given as $L = 1\ m,\ m_p = 0.2\ kg,\ m_c = 0.5\ kg,\ \rho = 0.02\ m,\ c_a = 0.116\ Vs$. The values of

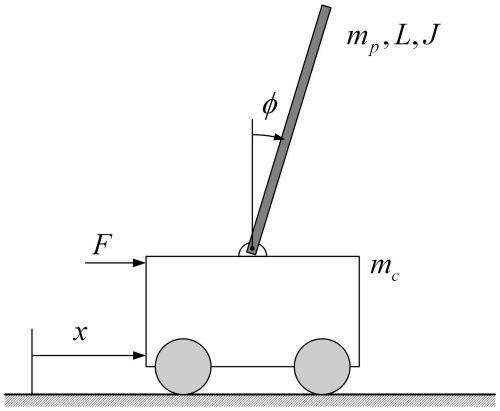


Figure 5.6: Inverted pendulum.

ω_1 and ω_2 are assumed to be uncertain up to $\pm 5\%$, and $\pm 10\%$, respectively.

Using the extension provided in Section 5.6, two 4^{th} order dynamic output-feedback switched RFTTC are designed for two different nominal performance objectives. One of these controllers is designed to have smaller nominal overshoot than the other one.[5] A unit step function is considered as the reference signal w_{ref} for both controllers. The admissible radius of the perturbed weighted states sphere is chosen as $r_x = 0.35$ for the time horizon of $[0, 20]$.

The following 4^{th} order dynamic output-feedback controllers are designed:

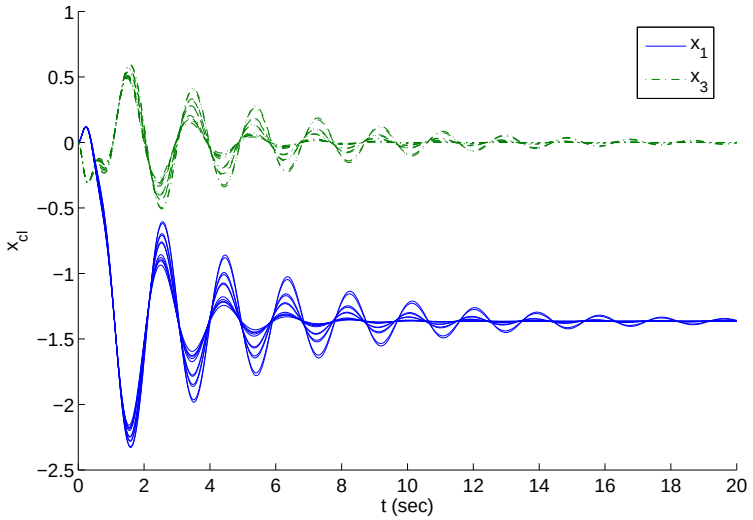
$$\mathcal{K}^{(1)} = \left[\begin{array}{cccc|c} 0.61 & 2.21 & 0.01 & 0.06 & 1.1 \\ -13.26 & 0.65 & 0.004 & 0.017 & 14 \\ -0.008 & 0.009 & -153.5 & -63.5 & -66.3 \\ 0.045 & 0.05 & 63.5 & -1 & 4.38 \\ \hline -7.83 & 0.81 & -66.3 & -4.38 & 0.07 \end{array}\right], \tag{5.51}$$

$$\mathcal{K}^{(2)} = \left[\begin{array}{cccc|c} 0.67 & 2.19 & 0 & 0 & 1.2 \\ -13.2 & 0.67 & 0 & 0 & 14 \\ 0 & 0 & -153.5 & -63.5 & -66.3 \\ 0 & 0 & 63.5 & -0.9 & 4.3 \\ \hline -7.9 & 0.74 & -66.3 & -4.38 & 0 \end{array}\right]. \tag{5.52}$$

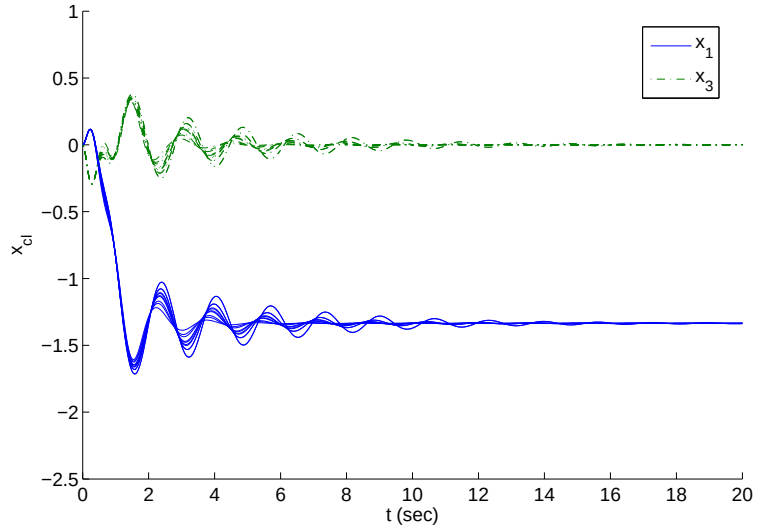
Figures 5.7(a) and 5.7(b) show, respectively for the controllers $\mathcal{K}^{(1)}$ and $\mathcal{K}^{(2)}$ without any switching, the cart position $x_1(t)$ and the pendulum angle $x_3(t)$ for closed-loop systems with 15 randomly perturbed plants. One can see that $\mathcal{K}^{(1)}$ gives more oscillatory state trajectories than does $\mathcal{K}^{(2)}$.

To verify that the requirements of Problem 5.2.1 are satisfied by the designed controllers with any arbitrary switching, a randomly generated switch-

[5] A nominal l_∞ performance is used to incorporate a maximum nominal overshoot constraint in the controller design. The method of [23] provides us with the optimization problem formulation of such a nominal overshoot specification. This method can be easily incorporated with our method, because both of the methods use a nonsmooth optimization approach.



(a) Closed-loop states for controller $\mathcal{K}^{(1)}$.



(b) Closed-loop states for controller $\mathcal{K}^{(2)}$.

Figure 5.7: Closed-loop states for the inverted pendulum without switching.

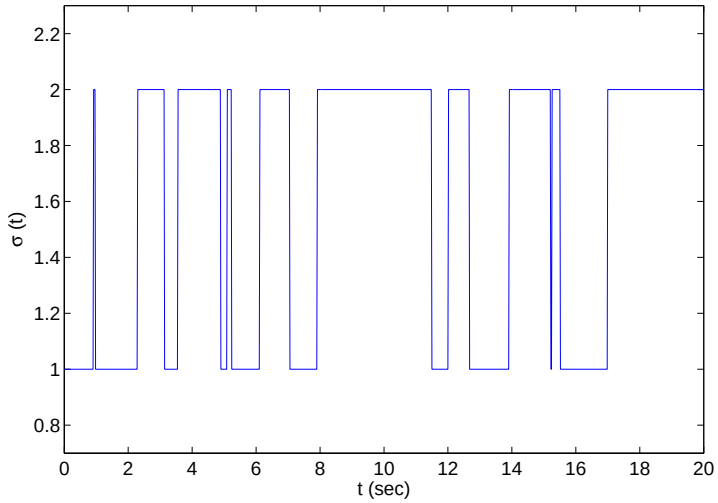
ing signal, shown in Figure 5.8(a), is applied to the closed-loop system. The closed-loop pendulum angle $x_1(t)$ and the cart position $x_3(t)$ trajectories for 15 randomly perturbed for the switching signal $\sigma(t)$ plants are shown in Figure 5.8(b). As can be seen in this figure, the closed-loop trajectories of the uncertain plants remain in $\mathbb{S}(x_{cl}^{[0,w_{ref},\sigma,0]}, r_x)$ at all times $t > 0$.

To investigate effectiveness of the proposed switched RFTTC, we compare its performance with the one designed by the method of [59], that does not take switching into account. In the method of [59], the controller satisfies a condition similar to requirement of Problem 5.2.1, but *without making any consideration about the switching controller*. Two robust controllers are designed using the method of [59] with different nominal l_∞-norm performance specifications, and these controllers are combined to make a switching controller. Applying the method of [51] on these controllers, new realizations and a reset map are obtained to guarantee the nominal stability of the switching controller. Figure 5.9(a) shows a switching signal $\sigma(t)$ that is applied to that switching controller. As shown in Figure 5.9(b), the nominal closed-loop response is stable that is consistent with the result of [51], while the perturbed trajectory is unstable. This indicates that the stability of each mode on its own is not a sufficient condition for robust stability of a switching uncertain system, even when all of the controllers are designed to be robust.

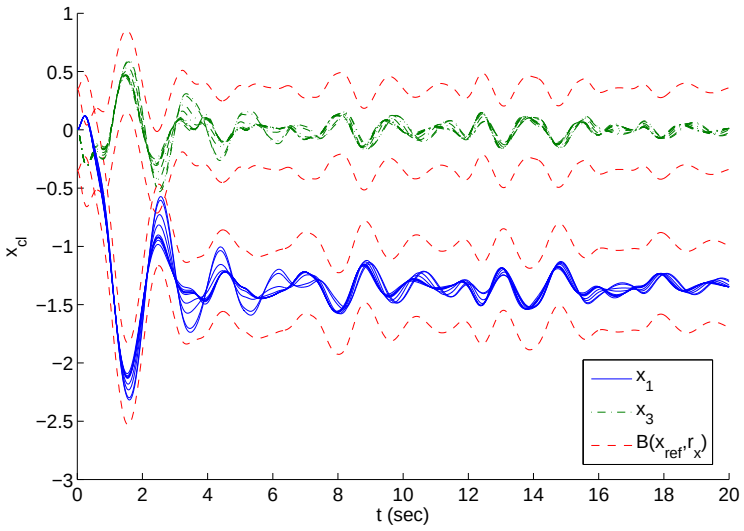
5.8 Conclusions

In this chapter, we proposed a switched RFTTC design method for LTI systems with parametric uncertainty. This method designs a set of fixed order dynamic output-feedback controllers which guarantees the RFTT in the presence of parametric uncertainty for any arbitrary switching signal. The proposed controller design method includes an optimization problem. A nonsmooth optimization approach was proposed to find a local optimal solution to the optimization problem.

The main advantage of this method is that, it can be used to extend the

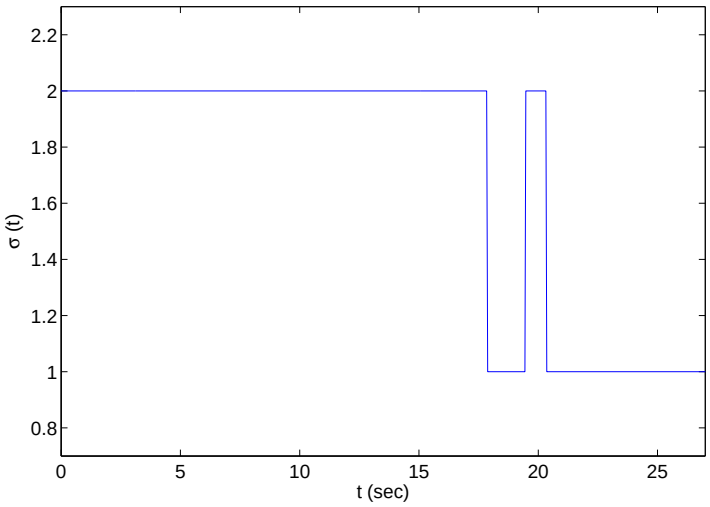


(a) The switching signal $\sigma(t)$.

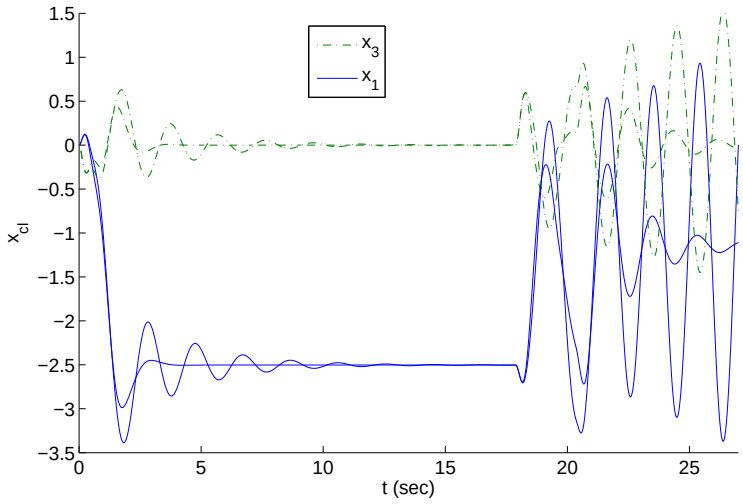


(b) The closed-loop trajectories for the switching signal $\sigma(t)$.

Figure 5.8: Closed-loop states for the inverted pendulum with switching.



(a) The switching signal $\sigma(t)$.



(b) The nominal closed-loop trajectory (solid) and a sampled perturbed trajectory (dashed) for the switching signal $\sigma(t)$.

Figure 5.9: An unstable closed-loop trajectory for the inverted pendulum with switching.

currently available results on switched nominal controller design methods to switched RFTTC where the plant uncertainty is taken into account. The proposed method is a natural choice for problems with time domain specifications, because by avoiding conservative frequency-domain approximations of time-domain specifications, our method has the potential to improve the robust performance.

Due to use of nonsmooth optimization algorithm in our method, other performance constraints such as H_∞ and H_2 constraints can also be readily mixed with the original robust performance constraint by following the path of [10, 12].

Chapter 6

Conclusions and Future Research Directions

6.1 Conclusions

We considered among several theoretical and practical limitations of conventional robust controllers, the robustness-imposed limitations and limitations due to the use of a linear time-invariant (LTI) controller for LTI uncertain plants in this thesis.

The former performance limitations were avoided by introducing multiple robust controllers (MRC). With this method, we divide the uncertainty set of an uncertain linear plant into a number of subsets, and assign a robust controller to each subset. As a consequence, because of the smaller uncertainty set that each controller must be robust against, the performance is enhanced versus the conventional robust controllers. Nevertheless, the performance improvement comes at the cost of both computational complexity associated with the division of the uncertainty set into a number of subsets, as well as requiring robust controller synthesis for each subset.

We proposed two computationally efficient methods for the design of MRC in Chapters 2 and 4. Our methods lay out the design of a set of

uncertainty subsets, and a corresponding set of fixed-order dynamic output-feedback controllers for time-invariant and time-varying uncertain plants, respectively. The methods differ in two main ways. First, in implementation, the former method uses an LTI controller, whereas the time-varying characteristic of the plant in the latter method may employ a switching controller that requires additional consideration for stability. Second, while the former method performs the calculations for the design of each controller and each parameter subset in a subsequent manner, the latter method does these calculations simultaneously. However, we note that the simultaneous calculations increase the computational complexity of the latter design method. Despite these differences, both of the methods for controller design encounter optimization problems that are non-convex and nonsmooth. We have developed nonsmooth optimization algorithms to find a local optimum for these problems in a computationally efficient manner.

We have investigated the computational performance of these algorithms, as well as the performance of the designed controllers through various numerical examples. The comparison of these results to those of previous methods reveals significant improvement in the computation speed, as discussed in Section 2.5, and as explained in more detail in Section 6.2. Our method was also applied to the practical example of the track-following controller design problem in hard disk drives (HDDs). Our experiments show significant improvement (more than 10% in the root mean square (RMS) of the position error signal (PES)) in the performance of the proposed controller when compared to a conventional robust controller.

The latter limitations; i.e., due to the use of an LTI controller for LTI uncertain plants, have been resolved by using switched robust finite-time tracking controllers (RFTTC). We introduce for these controllers the notion of robust finite-time tracking, which requires that all trajectories of an uncertain system to stay within a bounded neighborhood of a reference trajectory for any arbitrary switching signal, has been introduced for such controllers.

We proposed a controller synthesis method that ensures, in the sense of robust finite-time tracking, the robustness of a closed-loop switched system. Our controller design includes finding a locally-optimal solution by employing a nonsmooth optimization approach. We provided numerical example to show the advantages of our method over previous methods.

6.2 Summary of contributions

The contributions of this thesis are as follows: first, *we developed two MRC design methods that*

- *determine the number of uncertainty divisions sufficient to achieve a user-specified performance bound*, in contrast to previous methods [29, 30, 78, 107], where the designer has to specify the number of divisions. This gives our methods a significant advantage in practical applications where the designer needs to achieve a performance objective, but has no knowledge about the number of divisions required. Consequently, the previous methods would lead to a cumbersome trial and error process.

- *are computationally faster than those cited in the literature* for two reasons. First, in our methods, once an uncertainty subset and its corresponding controller are designed, they are not modified, while the alternative methods of [29, 30] modify the uncertainty set divisions and controllers iteratively. Second, in the methods of [29, 30], the presence of Lyapunov variables increases the number of optimization variables in the controller design, but in the proposed method, the Lyapunov variables are avoided by using an alternative stability analysis technique that does not affect the size of the optimization problem. In Table 6.1, we compare the effects of various system parameters on the computation time of MRC design methods of [29, 30] and of those proposed in this thesis.

system parameter	previous methods	proposed methods
number of dimensions of the uncertainty set	quadratic	quadratic
order of the plant	quadratic	not affected
number of uncertainty divisions	quadratic	linear

Table 6.1: The effect of various system parameters on the rate of change in computation time.

- *can make use of advantages inherent to the nonsmooth analysis, such as multiband controller synthesis, multidisc H_∞, mixed H_2/H_∞ controller design.* These design methods significantly enhance the practical aspects of our proposed MRC method.

Moreover, *the switching MRC design method of Chapter 4 is the first MRC method in the literature that can be applied to plants with both time-invariant and time-varying parametric uncertainties.* Traditional MRC design methods are applicable to plants with either time-invariant uncertainty [29, 30] or time-varying uncertainty [78, 107].

Second, *we proposed a switched RFTTC design method that*

- *extends the currently available "finite-time stability" concept to tracking in plants with uncertainty.* This approach is useful in those problems where transient response is more important than the steady-state response.

- *designs the controllers in time-domain.* Hence, our proposed solution is a natural choice for those problems with time domain specifications. Traditional controller design methods usually approximate time-domain specifications by frequency-domain specifications, which normally leads to conservative results. By avoiding such conservatism, our method has the potential to improve the robust performance for time-domain specifications.

Finally, *we suggested a novel systematic procedure for track-following controller design in HDDs*. Our design procedure uses the idea of MRC to enhance the track-following performance. Experimental results demonstrated more than a 10% improvement in the RMS of the PES, with this method.

6.3 Future research directions

This section recommends a number of potential future research directions.

6.3.1 Multi-rate and multi-sensing MRC

In practical examples, I/O channels may have different sampling/hold rates. For example, in dual stage HDDs, the sampling rate of the position error signal depends on the disk spinning speed, while the piezo vibration sensor has a flexible sampling rate. The most trivial approach to this problem is to use the slowest sampling rate for all sensors. Nevertheless, this approach degrades the closed-loop performance. Thus, a number of methods have been proposed for multi-rate and multi-sensing controller design; e.g., [63, 65, 66, 83].

The proposed design method for MRC can be extended to deal with multi-rate and multi-sensing control systems. Often, in multi-rate and multi-sensing controller design methods, a closed-loop system is converted to an equivalent periodic time-varying system [83]. Hence, the MRC method that we developed in Chapter 4 for time-varying systems, is suitable for this application.

6.3.2 Optimal switching instance and smooth switching

Consider the switching MRC method presented in Chapter 4, the controller may switch according to the value of the time-varying parameter. Since the uncertainty subsets are overlapped, switching might occur at any point in

the overlapping region. Therefore, the stability margin and the transient response of the closed-loop system can be improved by finding an optimal switching instance. Various on-line and off-line techniques are available for achieving this goal.

Moreover, a smooth controller transition strategy should be developed to prevent an unfavorable oscillatory transient response at controller switching instances. A smooth transition can be accomplished in two ways:

Smooth change of controller parameters: Instead of instantaneously switching the controller, the controller parameters can be changed, after switching instances, from one to another in a timely manner by interpolating two controllers for a short period of time. However, using this approach, special attention should be paid to the stability of the time-varying controller.

Initial value compensation: The initial values of the controller states at the instant of switching are chosen such that any transient oscillations are minimized.

The sequential use of each approach mentioned above may improve the transient response of the MRC of Chapter 4.

6.3.3 Parameter estimation in the switching MRC

In the proposed switching MRC method, we assume that the value of the time-varying parameter $\boldsymbol{\delta}(t)$ is available in real time. However, in some application the parameter $\boldsymbol{\delta}(t)$ may not be directly measurable. For those applications, a method should be developed to estimate $\boldsymbol{\delta}(t)$. The estimation method can either estimate the exact value of the parameter [13] or find the uncertainty subset that contains $\boldsymbol{\delta}(t)$ with the highest probability [22, 40, 46]. The estimation should be performed on-line, hence a computationally efficient algorithm should be developed.

6.3.4 Extensions of switched RFTTC

As explained earlier, one advantage of the method is that it designs the controllers in time-domain. The method can deal with a vast variety of practically important time-domain performance requirements, such as percent over-shoot, rise time, and settling time. Among these time-domain performances, the percent over-shoot is incorporated into the proposed method in the example section of Chapter 5. Other useful controller design methods can be developed by integrating various time-domain performances with the proposed switched RFTTC approach.

6.3.5 Supervisory controller design for the switched RFTTC

The proposed switched controllers design method guarantees stability and robust finite-time tracking for *any arbitrary switching signal*. The assumption of an arbitrary switching signal has an advantage in that any choice of the supervisory controller (that is responsible for generating the switching signal as shown in Figure 1.4) does not violate the closed-loop stability and the robust finite-time tracking. Nevertheless, in those applications where information about the supervisory controller is available, this assumption may result in conservative controllers. Consequently, two possible research directions might be explored:

- Subsequent to the design of switched RFTTC, the next issue is the design of a supervisory controller for various performance objectives. The advantage of this approach is that with regard to the design of the supervisory controller, closed-loop stability is not a concern. Because the stability has already been proven regardless of any choice of the supervisory controller in the design of the switched RFTTC.

- Next we must formulate and solve a problem that includes the design of both the supervisory controller and the switched RFTTC. In this new

problem, the switching signal is essentially constrained by the supervisory controller. Therefore, the performance of the switched RFTTC can be improved by considering the switching signal to be known.

Bibliography

[1] A. Al Mamun and G. Guo, *Hard Disk Drive: Mechatronics and Control*, CRC Press, 2006.

[2] F. Amato, M. Ariola, and P. Dorato, *Finite-time control of linear systems subject to parametric uncertainties and disturbances*, Automatica **37** (2001), no. 9, 1459 – 1463.

[3] P. Apkarian and R.J. Adams, *Advanced gain-scheduling techniques for uncertain systems*, IEEE Transactions on Control Systems Technology **6** (1998), no. 1, 21 –32.

[4] P. Apkarian, P. Gahinet, and G. Becker, *Self-scheduled H_∞ control of linear parameter-varying systems: A design example*, Automatica **31** (1995), no. 9, 1251 – 1261.

[5] P. Apkarian and D. Noll, *Spectral bundle methods for non-convex maximum eigenvalue functions: first-order methods*, Mathematical Programming **104** (2005), no. 2, 701–727.

[6] ———, *Controller design via nonsmooth multi-directional search*, SIAM Journal on Control and Optimization **44** (2006), no. 6, 1923–1949.

[7] ———, *Frequency domain H_∞ synthesis using nonsmooth techniques*, IEEE Conference on Industrial Electronics and Applications (Singapore, Singapore), 2006.

[8] ———, *IQC analysis and synthesis via nonsmooth optimization*, Systems & Control Letters **55** (2006), no. 12, 971 – 981.

[9] ———, *Nonsmooth H_∞ synthesis*, IEEE Transactions on Automatic Control **51** (2006), no. 1, 71–86.

[10] ———, *Nonsmooth optimization for multidisk H_∞ synthesis*, European Journal of Control **12** (2006), no. 3, 229–248.

[11] ———, *Nonsmooth optimization for multiband frequency domain control design*, Automatica **43** (2007), no. 4, 724–731.

[12] P. Apkarian, D. Noll, and A. Rondepierre, *Nonsmooth optimization algorithm for mixed H_2/H_∞ synthesis*, Proceedings of the IEEE Conference on Decision and Control (New Orleans, LA, United States), 2008, pp. 4110 – 4115.

[13] K.J. Åström and B. Wittenmark, *Adaptive control*, Pearson Education India, 2006.

[14] E. Azadi Yazdi and R. Nagamune, *A parameter set division and switching gain-scheduling controllers design method for time-varying plants*, Submitted for publication.

[15] ———, *Robust practical tracking with switched controllers*, Submitted for publication.

[16] ———, *Multiple robust H_∞ controller design using the nonsmooth optimization method*, International Journal of Robust and Nonlinear Control **20** (2010), no. 11, 1197–1212.

[17] ———, *Switched tracking controller design for uncertain systems*, ASME Dynamic Systems and Control Conference (Boston, MA), September 2010, pp. 4015(1)–4015(8).

[18] E. Azadi Yazdi, M. Sepasi, F. Sassani, and R. Nagamune, *Automated multiple robust track-following control system design in hard disk drives*, To appear in IEEE Transactions on Control System Technology.

[19] ———, *Automated multiple robust track-following control system design in hard disk drives*, ASME Dynamic Systems and Control Conference (Boston, MA), September 2010, pp. 4163(1)–4163(6).

[20] S. Bashash and N. Jalili, *Robust multiple frequency trajectory tracking control of piezoelectrically driven micro/nanopositioning systems*, IEEE Transactions on Control Systems Technology **15** (2007), no. 5, 867–878.

[21] G. Becker and A. Packard, *Robust performance of linear parametrically varying systems using parametrically-dependent linear feedback*, Systems & Control Letters **23** (1994), no. 3, 205–215.

[22] A. Bemporad, J. Roll, and L. Ljung, *Identification of hybrid systems via mixed-integer programming*, IEEE Conference on Decision and Control, vol. 1, 2001, pp. 786–792.

[23] V. Bompart, P. Apkarian, and D. Noll, *Control design in the time and frequency domain using nonsmooth techniques*, Systems & Control Letters **57** (2008), no. 3, 271–282.

[24] S. Boyd, L. E. Ghaoui, E. Feron, and V. Balakrishnan, *Linear Matrix Inequalities in System and Control Theory*, vol. 15, Society for Industrial and Applied Mathematics, 1994.

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

[26] B.M. Chen, T.H. Lee, K. Peng, and V. Venkataramanan, *Hard Disk Drive Servo Systems*, 2nd ed., Advances in Industrial Control, Springer, Germany, 2006.

[27] T.-L. Chen and R. Horowitz, *Design, fabrication and dynamic analysis of a PZT-actuated silicon suspension*, Proceedings of the 2001 American Control Conference, vol. 2, 2001, pp. 1235–1240.

[28] M. Chilali, P. Gahinet, and P. Apkarian, *Robust pole placement in LMI regions*, IEEE Transactions on Automatic Control **44** (1999), no. 12, 2257–2270.

[29] J. Choi, R. Nagamune, and R. Horowitz, *Multiple robust controller design based on parameter dependent lyapunov functions*, Proceedings of 17th International Symposium on Mathematical Theory of Networks and Systems, 2006.

[30] ———, *Multiple robust track-following controller design in hard disk drives*, International Journal of Adaptive Control and Signal Processing **22** (2007), no. 4, 359–373.

[31] F. H. Clarke, *Optimization and Nonsmooth Analysis*, Classics in applied mathematics, vol. 5, SIAM, 1990.

[32] F.H. Clarke, Y.S. Ledyaev, R.J. Stern, and P.R. Wolenski, *Nonsmooth Analysis and Control Theory*, Springer-Verlag, New York, 1998.

[33] R. Conway and R. Horowitz, *A μ-synthesis approach to guaranteed cost control in track-following servos*, The International Federation of Automatic Control, 2008, pp. 833–838.

[34] M.L. Corradini and G. Orlando, *An observer based switching control strategy for the stabilization of uncertain multi-input-multi-output plants*, International Journal of Control **80** (2007), no. 4, 583–592.

[35] J. Daafouz, P. Riedinger, and C. Iung, *Stability analysis and control synthesis for switched systems: a switched Lyapunov function approach*, IEEE Transactions on Automatic Control **47** (2002), no. 11, 1883–1887.

[36] R.A. Decarlo, M.S. Branicky, S. Pettersson, and B. Lennartson, *Perspectives and results on the stability and stabilizability of hybrid systems*, Proceedings of the IEEE **88** (2000), no. 7, 1069–1082.

[37] C.A. Desoer and M. Vidyasagar, *Feedback Systems: Input-output Properties*, Society for Industrial Mathematics, 2009.

[38] P. Dorato, *Current trends in nonlinear systems and control*, ch. An Overview of Finite-Time Stability, pp. 185–194, Birkhuser Boston, 2006.

[39] J.C. Doyle, K. Glover, P.P. Khargonekar, and B.A. Francis, *State-space solutions to standard H_2 and H_∞ control problems*, IEEE Transactions on Automatic Control **34** (1989), no. 8, 831–847.

[40] S. Fekri, M. Athans, and A. Pascoal, *Robust multiple model adaptive control (rmmac): a case study*, International Journal of Adaptive Control and Signal Processing **21** (2007), no. 1, 1–30.

[41] A. Feuer, G.C. Goodwin, and M. Salgado, *Potential benefits of hybrid control for linear time invariant plants*, Proceedings of the American Control Conference, vol. 5, 1997, pp. 2790–2794.

[42] H. Fujita, K. Suzuki, M. Ataka, and S. Nakamura, *A microactuator for head positioning system of hard disk drives*, IEEE Transactions on Magnetics **35** (1999), 1006–1010.

[43] P. Gahinet and P. Apkarian, *A linear matrix inequality approach to H_∞ control*, International Journal of Robust and Nonlinear Control **4** (1994), no. 4, 421–448.

[44] P. Gahinet, P. Apkarian, and M. Chilali, *Affine parameter-dependent lyapunov functions and real parametric uncertainty*, IEEE Transactions on Automatic Control **41** (1996), no. 3, 436 –442.

[45] S. Gumussoy, M. Millstone, and ML Overton, *H_∞ strong stabilization via HIFOO: A package for fixed-order controller design*, IEEE Conference on Decision and Control, Cancun, Mexico, 2008.

[46] P.D. Hanlon and PS Maybeck, *Multiple-model adaptive estimation using a residual correlation Kalman filter bank*, IEEE Transactions on Aerospace and Electronic Systems **36** (2002), no. 2, 393–406.

[47] R.L. Hecker, G.M. Flores, Q. Xie, and R. Haran, *Servocontrol of machine-tools: A review*, Latin American applied research **38** (2008), 85–94.

[48] A. Helmersson, *Methods for robust gain scheduling*, Ph.D. thesis, Linkoping University, Sweden, 1995.

[49] J.W. Helton and A. Sideris, *Frequency response algorithms for H_∞ optimization with time domain constraints*, IEEE Transactions on Automatic Control **34** (1989), no. 4, 427–434.

[50] J.P. Hespanha, D. Liberzon, and A.S. Morse, *Logic-based switching control of a nonholonomic system with parametric modeling uncertainty*, Systems and Control Letters **38** (1999), no. 3, 167–178.

[51] J.P. Hespanha and A.S. Morse, *Switching between stabilizing controllers*, Automatica **38** (2002), no. 11, 1905–1917.

[52] J.P. Hespanha, P. Naghshtabrizi, and Y. Xu, *A survey of recent results in networked control systems*, Proceedings of the IEEE **95** (2007), no. 1, 138–162.

[53] H. Hjalmarsson, *Efficient tuning of linear multivariable controllers using iterative feedback tuning*, International Journal of Adaptive Control and Signal Processing **13** (1999), no. 7, 553–572.

[54] H. Hjalmarsson, S. Gunnarsson, and M. Gevers, *A convergent iterative restricted complexity control design scheme*, Proceedings of the 33rd IEEE Conference on Decision and Control, vol. 2, 1994, pp. 1735–1740.

[55] K. Hoffman and R.A. Kunze, *Linear Algebra*, Prentice-Hall Englewood Cliffs, NJ, 1971.

[56] I.M. Horowitz, *Synthesis of feedback systems with nonlinear time-varying uncertain plants to satisfy quantitative performance specifications*, Proceedings of the IEEE **64** (1976), no. 1, 123–130.

[57] F.Y. Huang, T. Semba, W. Imaino, and F. Lee, *Active damping in HDD actuator*, IEEE Transactions on Magnetics **37** (2001), no. 2, 847–849.

[58] T. Imamura, M. Katayama, Y. Ikegawa, T. Ohwe, R. Koishi, and T. Koshikawa, *MEMS-based integrated head/actuator/slider for hard disk drives*, IEEE/ASME Transactions on Mechatronics **3** (1998), no. 3, 166–174.

[59] S. Jayasuriya, *Robust tracking for a class of uncertain linear systems*, International Journal of Control **45** (1987), no. 3, 875–892.

[60] U. Jonsson, *Robustness analysis of uncertain and nonlinear systems*, Ph.D. thesis, Department of Automatic Control, Lund Institute of Technology, 1996.

[61] S. Kanev, C. Scherer, M. Verhaegen, and B. D. Schutter, *Robust output-feedback controller design via local BMI optimization*, Automatica **40** (2004), no. 7, 1115–1127.

[62] M. V. Kothare, V. Balakrishnan, and M. Morari, *Robust constrained model predictive control using linear matrix inequalities*, Automatica **32** (1996), 1361–1379.

[63] S. Lall and G. Dullerud, *An LMI solution to the robust synthesis problem for multi-rate sampled-data systems*, Automatica **37** (2001), no. 12, 1909–1922.

[64] K. Lau and R. Middleton, *On the use of switching control for systems with bounded disturbances*, Proceedings of the 39th IEEE Conference on Decision and Control, vol. 4, 2000, pp. 3598–3603.

[65] S.H. Lee, *Multirate digital control system design and its application to computer disk drives*, IEEE Transactions on Control Systems Technology **14** (2006), no. 1, 124–133.

[66] S.H. Lee, C.C. Chung, and S.M. Suh, *Multirate digital control for high track density magnetic disk drives*, IEEE Transactions on Magnetics **39** (2003), no. 2, 832–837.

[67] S.H. Lee and J.T. Lim, *Switching control of H_∞ gain scheduled controllers in uncertain nonlinear systems*, Automatica **36** (2000), 1067–1074.

[68] D. J. Leith and W. E. Leithead, *Survey of gain-scheduling analysis and design*, International Journal of Control **73** (2000), no. 11, 1001–1025.

[69] Y. Li and R. Horowitz, *Mechatronics of electrostatic microactuators for computer diskdrive dual-stage servo systems*, IEEE/ASME Transactions on Mechatronics **6** (2001), no. 2, 111–121.

[70] ———, *Design and testing of track-following controllers for dual-stage servo systems with PZT actuated suspensions*, Microsystem Technologies **8** (2002), 194–205.

[71] Y. Li, F. Marcassa, R. Horowitz, R. Oboe, and R. Evans, *Track-following control with active vibration damping of a PZT-actuated suspension dual-stage servo system*, ASME Journal of Dynamic Systems, Measurement, and Control **128** (2006), 568–576.

[72] D. Liberzon, *Switching in Systems and Control*, Birkhauser, Boston, 2003.

[73] H. Lin and P.J. Antsaklis, *Stability and stabilizability of switched linear systems: A short survey of recent results*, Proceedings of the 2005 IEEE International Symposium on Intelligent Control, Cyprus, 2005, pp. 24–29.

[74] P. Linz, *Theoretical Numerical Analysis: An Introduction to Advanced Techniques*, Courier Dover Publications, 2001.

[75] B. Liu, K.L. Teo, and X. Liu, *Robust exponential stabilization for large-scale uncertain impulsive systems with coupling time-delays*, Nonlinear Analysis **68** (2008), no. 5, 1169–1183.

[76] Z.Z. Liu, F.L. Luo, and M. Rahman, *Robust and precision motion control system of linear-motor direct drive for high-speed X-Y table positioning mechanism*, IEEE Transactions on Industrial Electronics **52** (2005), no. 5, 1357–1363.

[77] Y. Lou, P. Gao, B. Qin, G. Guo, E.H. Ong, A. Takada, and K. Okada, *Dual-stage servo with on-slider PZT microactuator for hard disk drives*, IEEE Transactions on Magnetics **38** (2002), no. 5, 2183– 2185.

[78] B. Lu and F. Wu, *Switching LPV control designs using multiple parameter-dependent Lyapunov functions*, Automatica **40** (2004), no. 11, 1973–1980.

[79] N.H. McClamroch and I. Kolmanobsky, *Performance benefits of hybrid control design for linear and nonlinear systems*, Proceedings of the IEEE **88** (2000), no. 7, 1083–1096.

[80] A. Megretski and A. Rantzer, *System analysis via integral quadratic constraints*, IEEE Transactions on Automatic Control **42** (1997), no. 6, 819–830.

[81] A. S. Morse, *Supervisory control of families of linear set-point controllers - part 1: Exact matching*, IEEE Transaction on Automatic Control **41** (1998), 1413–1431.

[82] R. Nagamune, X. Huang, and R. Horowitz, *Robust control synthesis techniques for multirate and multi-sensing track-following servo systems in HDDs*, ASME Journal of Dynamic Systems, Measurement and Control.

[83] ———, *Multirate track-following control with robust stability for a dual-stage multi-sensing servo system in HDDs*, 44th IEEE Conference on Decision and Control, 2005, pp. 3886–3891.

[84] C. Okwudire and Y. Altintas, *Minimum tracking error control of flexible ball screw drives using a discrete-time sliding mode controller*, Journal of Dynamic Systems, Measurement, and Control **131** (2009), 051006.

[85] A. Packard and J. Doyle, *The complex structured singular value*, Automatica **29** (1993), no. 1, 71–109.

[86] A. Paul and M.G. Safonov, *Model reference adaptive control using multiple controllers and switching*, Proceedings of the 42nd IEEE Conference on Decision and Control, vol. 4, 2003, pp. 9–12.

[87] E. Polak, *Optimization: Algorithms and Consistent Approximations*, Springer, 1997.

[88] V.M. Popov, *Absolute stability of nonlinear control systems*, Automation and Remote Control **22** (1961), no. 8, 857–875.

[89] WJ Rugh and J.S. Shamma, *Research on gain-scheduling*, Automatica **36** (2000), no. 10, 1401–1425.

[90] C. Scherer, P. Gahinet, and M. Chilali, *Multiobjective output-feedback control via LMI optimization*, IEEE Transactions on Automatic Control **42** (1997), no. 7, 896–911.

[91] S. J. Schroceck and W. C. Messner, *On compensator design for linear time-invariant dual-input single-output systems*, IEEE/ASME Transactions on Mechatronics **6** (2001), no. 1, 50–57.

[92] S. Skogestad and I. Postlethwaite, *Multivariable Feedback Control: Analysis and Design*, 2 ed., Wiley, Chichester, 2005.

[93] K.K. Tan, T.H. Lee, and S. Huang, *Precision Motion Control: Design and Implementation*, Springer Verlag, 2008.

[94] A.R. Teel, *Robust Hybrid Control Systems: An Overview of Some Recent Results*, Lecture notes in control and information sciences **353** (2007), 279–302.

[95] J.B. Thevenet, D. Noll, and P. Apkarian, *Nonlinear spectral SDP method for BMI-constrained problems: Applications to control design*, Informatics in Control, Automation and Robotics **1** (2005), 61–72.

[96] J.G. VanAntwerp, R.D. Braatz, and N.V. Sahinidis, *Globally optimal robust control for systems with time-varying nonlinear perturbations*, Computers and Chemical Engineering **21** (1997), 125–130.

[97] M. Vidyasagar and N. Viswanadham, *Reliable stabilization using a multi-controller configuration*, Automatica **21** (1985), no. 5, 599–602.

[98] B. Wie and D. Bernstein, *Benchmark problems for robust control design*, Journal of Guidance, Control, and Dynamics **15** (1992), no. 5, 1057–1059.

[99] F. Wu and K. Dong, *Gain-scheduling control of LFT systems using parameter-dependent Lyapunov functions*, Automatica **42** (2006), no. 1, 39–50.

[100] F. Wu, X.H. Yang, A. Packard, and G. Becker, *Induced L_2-norm control for LPV systems with bounded parameter variation rates*, International Journal of Robust and Nonlinear Control **6** (1998), no. 9-10, 983–998.

[101] Z. Xiang and R. Wang, *Robust L_∞ reliable control for uncertain nonlinear switched systems with time delay*, Applied Mathematics and Computation **210** (2009), no. 1, 202–210.

[102] D. Xie, L. Wang, F. Hao, and G. Xie, *Robust stability analysis and control synthesis for discrete-time uncertain switched systems*, Proceedings of the 42nd IEEE Conference on Decision and Control, vol. 4817, Maui, Hawaii USA, 2003.

[103] X. Xu, S. He, and G. Zhai, *Practical Uniform Asymptotic Stabilizability of Switched Systems with Time-Varying Subsystems*, American Control Conference, 2007, pp. 681–686.

[104] X. Xu and G. Zhai, *New results on practical stabilization and practical reachability of switched systems*, Proceedings of the American Control Conference, vol. 6, June 2005, pp. 3784–3789.

[105] X. Xue and J. Tang, *Robust and high precision control using piezoelectric actuator circuit and integral continuous sliding mode control design*, Journal of Sound and Vibration **293** (2006), no. 1-2, 335–359.

[106] V.A. Yakubovich, *Frequency conditions for the absolute stability of control systems with several nonlinear or linear nonstationary blocks*, Automation and Remote Control **28** (1967), no. 6, 5–30.

[107] P. Yan and H. Ozbay, *On switching H_∞ controllers for a class of linear parameter varying systems*, Systems & Control Letters **56** (2007), no. 7-8, 504–511.

[108] O. Yaniv and I.M. Horowitz, *A quantitative design method for MIMO linear feedback systems having uncertain plants*, IEEE Conference on Decision and Control **24** (1985), 882–887.

[109] G. Zhai and A.N. Michel, *On practical stability of switched systems*, Proceedings of the 41st IEEE Conference on Decision and Control, vol. 3, Dec. 2002, pp. 3488–3493.

[110] H. Zhang, X. Huang, G. Peng, and M. Wang, *Dual-stage HDD head positioning using an H_∞ almost disturbance decoupling controller and a tracking differentiator*, Journal of Mechatronics **19** (2009), no. 5, 788–796.

[111] L. Zhang, P. Shi, C. Wang, and H. Gao, *Robust H_∞ filtering for switched linear discrete-time systems with polytopic uncertainties*, International Journal of Adaptive Control and Signal Processing **20** (2006), no. 6, 291–304.

[112] J. Zhong and B. Yao, *Adaptive robust precision motion control of a piezoelectric positioning stage*, Control Systems Technology, IEEE Transactions on **16** (2008), no. 5, 1039–1046.

[113] K. Zhou and J.C. Doyle, *Essentials of Robust Control*, Prentice Hall Upper Saddle River, NJ, 1998.

[114] K. Zhou, P.P. Khargonekar, J. Stoustrup, and H.H. Niemann, *Robust performance of systems with structured uncertainties in state space*, Automatica **31** (1995), 249–255.

Appendix A

Proof of Theorem 4.3.1

We use integral quadratic constraint (IQC) approach to guarantee the robust stability and robust performance by regarding the time-varying parameter $\boldsymbol{\delta}(t)$ as a time-varying uncertainty. This assumption is essential for using IQC approach, but it results in conservative results, because the information about the value of the parameter will not be used in the analysis. Even though results are conservative, it provides a unified robust stability and performance analysis framework for various types of uncertainty, such as time-invariant and time-varying parametric uncertainty, sector bounded nonlinearity and saturation nonlinearity.

The IQC robust performance theorem is presented in Theorem B.2.1 of B. To use Theorem B.2.1, the closed-loop system configuration must be similar to the feedback interconnection shown in Figure B.1. Therefore $G(j\omega)$ in Theorem B.2.1 is replaced by $F_l\left(P_{\Delta_{\text{nom}}}, K\right)$, where $P_{\Delta_{\text{nom}}}$, represents frequency-domain representation of the plant with the nominal uncertainty

Δ_{nom} given by[1]

$$P_{\Delta_{\text{nom}}} := F_u \left(P, \begin{bmatrix} 0 & I & 0 \\ I & \Delta_{\text{nom}} & 0 \\ 0 & 0 & \frac{1}{s} \end{bmatrix} \right). \tag{A.1}$$

Theorem B.2.1 can be applied to the system that gives us (4.10) as the robust asymptotic stability and performance condition. It should be emphasized that since the uncertainty set is not normalized, Lemma B.2.2 should be used to define the multiplier.

[1]This equation is found by a similar method that is explained in Section 2.3.1 for time-invariant uncertain plants.

Appendix B

IQC Robust Performance Theorem

This appendix provides preliminary results for the switching gain-scheduling controllers synthesis algorithm in Section 4.3. The IQC approach is taken to guarantee robust stability and performance in Chapter 4 of this thesis. The IQC method, originated in [106], is a combination of the input/output stability theory [37] and the absolute stability theory [88]. The advantage of the IQC method is that it provides the stability condition for various types of uncertainty, such as time-invariant and time-varying parametric uncertainty, sector bounded nonlinearity and saturation nonlinearity, in a unified framework. See for example [60, 80].

B.1 IQC definition

Let us define the *IQC*.

Definition B.1.1. *Let* $\Pi_\Delta : j\mathbb{R} \to \mathbb{C}^{(2n_\Delta \times 2n_\Delta)}$ *be a bounded function that takes Hermitian values. An operator* Δ *is said to satisfy the IQC defined by*

Π_Δ *if the following holds for all square integrable signals* u_Δ*:*

$$\int_{-\infty}^{+\infty} \begin{bmatrix} u_\Delta(j\omega) \\ \Delta(u_\Delta)(j\omega) \end{bmatrix}^H \Pi_\Delta(j\omega) \begin{bmatrix} u_\Delta(j\omega) \\ \Delta(u_\Delta)(j\omega) \end{bmatrix} d\omega \geq 0. \tag{B.1}$$

We use the notation $\Delta \in \mathrm{IQC}(\Pi_\Delta)$ *to show that a map* Δ *satisfies the IQC defined by* Π_Δ*.*

B.2 IQC robust performance theorem

For the uncertainty set $\mathrm{IQC}(\Pi_\Delta)$ characterized by Π_Δ, we present the robust performance theorem.

Theorem B.2.1. *Assume the system interconnection shown in Figure B.1 where* $G(s)$ *is an LTI system and* Δ *an uncertainty operator. The system is uniformly internally stable* [1] *(and thus internally stable) and the* L_2 *gain condition* $\|z\|_2 \leq \gamma \|w\|_2$ *holds if there exists there exists a Hermitian-matrix-valued bounded function* $\Pi_\Delta := \begin{bmatrix} \Pi_{\Delta_{11}} & \Pi_{\Delta_{12}} \\ \Pi_{\Delta_{12}}^H & \Pi_{\Delta_{22}} \end{bmatrix}$ *such that* $\Delta \in \mathrm{IQC}(\Pi_\Delta)$*, and the frequency domain inequality (FDI)*

$$\begin{bmatrix} G(j\omega) \\ I \end{bmatrix}^H \Pi(j\omega) \begin{bmatrix} G(j\omega) \\ I \end{bmatrix} < 0, \quad \forall \omega \in [0, \infty], \tag{B.2}$$

is satisfied, where Π *is defined by*

$$\Pi := \begin{bmatrix} \Pi_{\Delta_{11}} & & \Pi_{\Delta_{12}} & \\ & \gamma^{-1} I & & 0 \\ \Pi_{\Delta_{12}}^H & & \Pi_{\Delta_{22}} & \\ & 0 & & -\gamma I \end{bmatrix}. \tag{B.3}$$

[1] The system in Figure B.1 is uniformly exponentially stable if there exist $m, \alpha > 0$, such that $|x(t)| \leq m e^{-\alpha t} |x(0)|$ for all $t \in \mathbb{R}_+$, where $x(t)$ is the state vector of G.

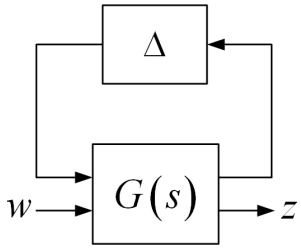


Figure B.1: Feedback interconnection for Theorem B.2.1.

In [60], it is shown that for a scaler time-varying uncertainty $\Delta(t) = \delta(t)$ with a derivative constraint $|\dot{\delta}(t)| \leq v$ for all $t \in \mathbb{R}_+$, the following Π_Δ satisfies the inequality (B.1).[2]

$$\Pi_\Delta := \begin{bmatrix} R^H R + v\Gamma(R,S)^H \Gamma(R,S) & S - S^H \\ S^H - S & -R^H R + v\Upsilon(R)^H \Upsilon(R) \end{bmatrix}, \quad \text{(B.4)}$$

where $R, S \in \mathbb{RH}_\infty$ with the realizations $R(s) = \left[\begin{array}{c|c} A_R & B_R \\ \hline C_R & D_R \end{array}\right]$ and $S(s) = \left[\begin{array}{c|c} A_S & B_S \\ \hline C_S & D_S \end{array}\right]$,

$$\begin{aligned} \Gamma(R,S) &= \left[R_B^H, S_B^H, S_C\right]^H, & \Upsilon(R) &= R_C^H R, \\ R_B(s) &= (sI - A_R)^{-1} B_R, & R_C(s) &= C_R (sI - A_R)^{-1}, \\ S_B(s) &= (sI - A_S)^{-1} B_S, & S_C(s) &= C_S (sI - A_S)^{-1}. \end{aligned} \quad \text{(B.5)}$$

The above results are valid for an uncertainty set $\mathbf{\Delta}_{TV}(\mathbf{\Delta}, \mathbf{\Delta}_v)$ with a normalized uncertain parameter. However, in Section 4.3, we need to apply these results on plants with an unnormalized uncertainty set. The following lemma can be used to extend the above results to such plants.

Lemma B.2.2. *Given a normalized uncertainty set* $\mathbf{\Delta}$ *and an unnormalized*

[2]This result can be easily extended to systems with a higher dimensional uncertainty set.

uncertainty set $\tilde{\mathbf{\Delta}}$ *such that* $\tilde{\mathbf{\Delta}} = \tau\mathbf{\Delta}$ *for some scaler* $\tau > 0$*. Then, the condition* $\mathbf{\Delta} \in \mathrm{IQC}(\Pi_\Delta)$ *is equivalent to the condition* $\tilde{\mathbf{\Delta}} \in \mathrm{IQC}(\Pi(\tau))$*, where*

$$\Pi(\tau) = \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{\tau} \end{bmatrix} \Pi_\Delta \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{\tau} \end{bmatrix}. \tag{B.6}$$

Proof. Starting from the left hand side of (B.6)

$$\begin{aligned}
&\mathbf{\Delta} \subset \mathrm{IQC}(\Pi_\Delta) \\
&\Leftrightarrow\ \tau\mathbf{\Delta} \in \tau\mathrm{IQC}(\Pi_\Delta) := \left\{ \tau\Delta : \int_{-\infty}^{+\infty} \begin{bmatrix} u_\Delta \\ \Delta(u_\Delta) \end{bmatrix}^H \Pi_\Delta \begin{bmatrix} u_\Delta \\ \Delta(u_\Delta) \end{bmatrix} d\omega \geq 0 \right\}, \\
&\Leftrightarrow\ \left\{ \tilde{\Delta} : \int_{-\infty}^{+\infty} \begin{bmatrix} u_\Delta \\ \frac{1}{\tau}\tilde{\Delta}(u_\Delta) \end{bmatrix}^H \Pi_\Delta \begin{bmatrix} u_\Delta \\ \frac{1}{\tau}\tilde{\Delta}(u_\Delta) \end{bmatrix} d\omega \geq 0 \right\}, \\
&\Leftrightarrow\ \left\{ \tilde{\Delta} : \int_{-\infty}^{+\infty} \begin{bmatrix} u_\Delta \\ \tilde{\Delta}(u_\Delta) \end{bmatrix}^H \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{\tau} \end{bmatrix} \Pi_\Delta \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{\tau} \end{bmatrix} \begin{bmatrix} u_\Delta \\ \tilde{\Delta}(u_\Delta) \end{bmatrix} d\omega \geq 0 \right\}, \\
&\Leftrightarrow\ \tilde{\mathbf{\Delta}} \subset \mathrm{IQC}\left(\begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{\tau} \end{bmatrix} \Pi_\Delta \begin{bmatrix} 1 & 0 \\ 0 & \frac{1}{\tau} \end{bmatrix} \right).
\end{aligned} \tag{B.7}$$

□