A TWO-STAGE PROBABILITY BASED, CONSERVATISM REDUCTION METHODOLOGY FOR TRADITIONAL MINIMAX ROBUST CONTROL SYSTEM DESIGN

A Thesis Defense
Presented to
The Academic Faculty

by

Linyu Zhang

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy in the
School of Aerospace Engineering

Georgia Institute of Technology
May 2016

A TWO-STAGE PROBABILITY-BASED, CONSERVATISM REDUCTION METHODOLOGY FOR TRADITIONAL MINIMAX ROBUST CONTROL SYSTEM DESIGN

Approved by:

Professor Dimitri N. Mavris, Advisor
School of Aerospace Engineering
Georgia Institute of Technology

Professor Graeme J. Kennedy
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Scott. Duncan
School of Aerospace Engineering
Georgia Institute of Technology

Dr. Ameet Deshpande
General Electric Global Research Center

Dr. Olugbenga Anubi
General Electric Global Research Center

Date Approved: April 4, 2016

ACKNOWLEDGEMENTS

First and foremost, I would like to express my sincere gratitude to this committee, especially my advisor Professor Dimitri Mavris, for taking me into the Aerospace System Design Lab (ASDL) in the first place, where I have felt at home. It's impossible to fully express such gratitude in words, as I couldn't complete the long Ph.D. journey without his unwavering mentoring and guidance, as well as his continuous support of my study and research. His instruction has been instrumental in my development as a researcher.

Besides my advisor, I would like to thank the rest of my thesis committee members for their encouragements, insightful comments, and hard questions. Special thanks to Dr. Scott Duncan for his careful guidance and patience through the whole Smart Campus project and his time-consuming efforts to evaluate and critique my Ph.D research. I also want to thank Professor Graeme J. Kennedy for sharing valuable time to review my work and for using his expertise in optimization to provide me technical feedback.

Also many thanks to Dr. Ameet Deshpande and Dr. Olugbenga Anubi from General Electric Global Research Center, with whom I got acquainted while I was on internship at their site. Their expertise accumulated from industry and insights of control system designs and optimization helped me to evaluate my research among all state-of-art solutions, as well the applicability of my method in modern industry. Talking to them is not only beneficial, but always a great pleasure.

While I do have the pleasure of working on multiple projects and with many people, I want to deliver thanks to my fellow lab mates in ASDL: Juhyun Kim, Yanal Issac and Arun Ramamuthy for the stimulating discussions and for all the fun we have had in the last few years.

Last but not the least, no words in the world can describe my feelings to my parents Shaomei Ru and Chao Zhang, for giving birth to me, raising me, encouraging me to chase my own dream and for supporting me financially and spiritually throughout my whole life. Though they are not around me location-wise, I know deeply in my heart that they are always standing with me and giving me encouragements until forever. Thank you.

Linyu Zhang

Atlanta, GA

May, 2016

Table of Contents

LIST OF TABLES

Page

LIST OF FIGURES

Page

LIST OF SYMBOLS AND ABBREVIATIONS

x	State variable
y	Measurement
u	Controller output
d	Disturbance
w	Sensor noise
r	Reference signal
e	Error signal
K_p	Proportional gain
K_i	Integral gain
K_d	Derivative gain
A	Control system state-space equation parameter
A_n	Normal control system state-space equation parameter
A_s	Steady-state control system state-space equation parameter
A_v	Variant/uncertain control system state-space equation parameter
A_k	Closed loop control system state-space equation parameter
A_v	Uncertain control system state-space equation parameter
ΔA	Variant from normal control system state-space equation parameter
Δa	One element/uncertainty channel in ΔA
B	Control system dynamic equation parameter
B_s	Steady-state control system state-space equation parameter
C	Control system state-space equation parameter
R	LQR objective function weight

Q	LQR objective function weight
P	Solution from Lyapunov/Riccati equation
V	Disturbance covariance matrix
W	Sensor noise covariance matrix
L	LQG controller gain
$J(x)$	Objective function
J_{lqr}/K_{lqr}	Linear Quadratic performance with Linear Quadratic Regulator
$J_{\max}\|K$	Worst-case performance with given controller K
$J_{minimax}/K_{minimax}$	Minimax performance with minimax controller
J_{ave}/K_{ave}	Optimal average performance with optimal average performance controller
θ	Adaptive parameter
H_∞	H-infinity norm
H_2	H-2 norm
t	Time
T_1	Supply air temperature
T_2	Room temperature
T_w	DX evaporator wall temperature
Z_1	Air side heat transfer area of evaporator
α_1	Air side convective heat transfer coefficients
ρ	Room air density
f	Supply air volume flow rate
V	Room air volume
V_{h1}	Air side volume of evaporator
Q_{load}	Room heat load

ABM	Agent Based Modeling
AFCS	Adaptive Flight Control System
AHU	Air Handling Unit
AR	Auto Regression
CG	Center of Gravity
DX	Direct Expansion
NEMCD	Norm Extended Minimax Control Design
GA	Genetic Algorithm
HVAC	Heating, Ventilation, and Air Conditioning
LQ	Linear Quadratic
LQE	Linear Quadratic Estimator
LQG	Linear Quadratic Gaussian
LQR	Linear Quadratic Regulator
MC	Monte Carlo
MIAC	Model Identification Adaptive Control
MIMO	Multiple Input Multiple Output
MRAC	Model Reference Adaptive Control
MISO	Multiple Input Single Output
PCOAPC	POF Constraint Optimal Average Performance Control
PI	Proportional Integral
PID	Proportional, Integral Derivative
POO	Probability of Occurrence
POF	Probability of Failure
SIMO	Single Input Multiple Output
SISO	Single Input Single Output

TMCD	Traditional Minimax Control Design
VAV	Variable Air Volume

SUMMARY

Control system design serves as one of the most important areas in modern engineering. There are many controller design methods developed to fulfill all types of design objectives. Optimal Control or equivalently Linear Quadratic Gaussian (LQG) Control is one of the most commonly used.

On the other side, there are several types of uncertainties generally inherent in a control system such as plant disturbance, sensor noise, and parameter uncertainty. While the first two are mitigated by LQG control, parameter uncertainty is not, and it will degrade a system's performance if the controller designed for the normal plant is still used. In such a circumstance, designers turn to Adaptive/Robust control designs within various performance domains and robustness requirements. When the traditional quadratic performance from LQG is still used and the robustness requirement is to optimize the worst-case scenario, the corresponding robust controller design method is Minimax controller design (MCD), which assumes norm-bounded parameter uncertainty, augments the worst-case performance with scale factors, and solves the performance optimization equation under the stability constraint in the form of coupled Lyapunov equations.

However, there are several gaps observed which lead to conservatism in the design.

1. The norm bounded assumption enforces dependence between uncertainties, which is not necessarily true;
2. There is an augmented term composed of scale factors added into the performance equation, which deviates the performance calculation from the real value;

3. There are no probability considerations associated with the uncertainty range. The incorporating of probability information into design relaxes the tight constraint and allows a tradeoff between the worst-case and other performances such as the most-likely or average performance.

In this research, a two-stage methodology is developed. In the first stage, the properties of parameter uncertainty when a controller is given is first analyzed, followed by a theorem that proves the worst-case point is always located at corners from the uncertainty space. Due to the fact that the location of the worst-case point also changes from corner to corner when the controller changes, it makes the worst-case performance curve not differentiable everywhere. Next, a line search method that profiles the intersection of two performance responses' topologies at two corners is proposed for the minimax controller design. An algorithm is also provided to survey all candidate corners to ensure that calculated minimax controller is global minimax.

The second stage of proposed methodology re-formulates the controller design problem in a way so that the merit of the optimization problem is to design a controller to optimize the average performance under parameter uncertainty while keep the probability of performance failure (POF) smaller than a given value. The performance constraint used to calculate POF is the aforementioned minimax performance with reduced conservatism from the first stage. A numerical method based on local performance-controller gradient calculations and the given probability distributions is provided to find a line search direction towards the optimal average performance. Instead of sampling method that becomes inefficient due to small POF considerations, a new discretized-summation numerical method is provided to calculate the POF at each step along the line

search so that above line search shifts direction to the constant POF contour after the POF constraint is hit.

Finally, a case study of an HVAC control system design, whose model contains two parameter uncertainties, is physically represented and then mathematically built in Matlab. Then the proposed methodology is fully applied to design a robust controller with reduced conservatism to demonstrate the whole design process.

CHAPTER 1

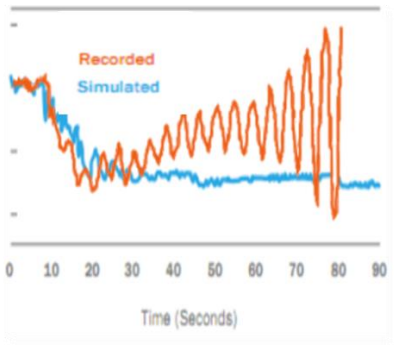
INTRODUCTION

1.1 Motivation

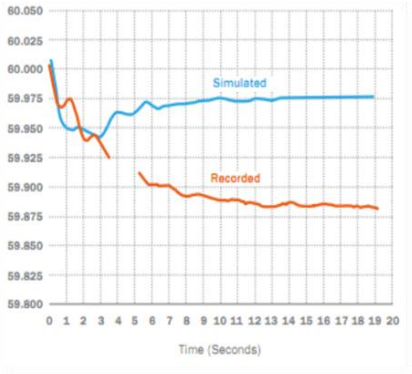
Control system design serves as one of the most important aspects in modern industry. It has wide application and is emphasized in almost all engineering fields. Over the century, many controller design methods have been developed by researchers towards multiple design objectives.

It is also well recognized that a good system design requires an accurate mathematical model in the design process. It is especially true for control system design due to its dynamic nature: the future state is based on the current state. Thus a small, unexpected offset may accumulate with time and gradually destroy the whole control system even the system is stable initially.

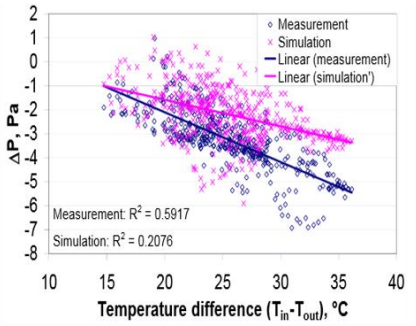
Taking the HVAC control system as an example, it is observed in real life and literature [1] [2] that many system designs suffer from inaccurate models. Figure 1 shows observed discrepancies between measured and simulated control system performance in multiple aspects such as stability (Figure 1. 1 (a)), transient performance (Figure 1. 1 (b)), discrete sampling (Figure 1. 1 (c)) and controller effort (Figure 1. 1 (d)).



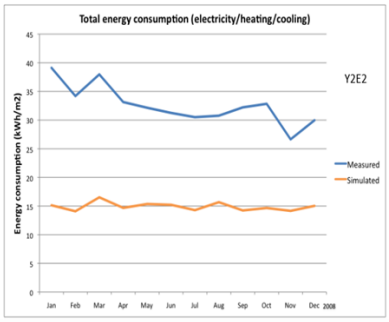
(a) Valve position output



(b) Supply air temperature variation



(c) Pressure control



(d) Building energy consumption

Figure 1. 1 Discrepancy between measured and simulated for HVAC system

When such discrepancy is observed, mainly it is caused by uncertainties in the system. Even if such uncertainty is observed and modeled, a system's performance cannot always be guaranteed due to the uncertain nature.

While further elaboration will be given in the next chapter, the observation here is that almost all types of systems suffer from uncertainties. This necessitates a method to systematically identify, analyze and mitigate the effects of these uncertainties to avoid control system performance degradation. In the next few chapters, such need will be addressed and the state-of-art solutions will be summarized from a literature search; gaps will be observed and new methodologies will be proposed to fill such gaps to improve current methodologies from current limitations.

1.2 Research Statement and Objective

In this research, there will be a systematic procedure performed to: identify all the uncertainties a control system should expect, analyze their properties and survey literature to find the state of art solutions, and perform a gap analysis and propose solutions.

It turns out that disturbance and sensor noise can be generally taken care of by LQG. However, parameter uncertainty cannot, and it can only be addressed by adaptive control and robust control. Further investigation shows that there are two levels of conservatism from the traditional norm-bounded, worst-case based robust control. The usage of parameter uncertainties' probabilistic information can be used to reduce such conservatism.

The research objective of this dissertation is the following:

Built up from the traditional minimax (worst-case based) robust control system design methodology with avoidable conservatism, develop a two-stage probability-based methodology to reduce such conservatism.

1.3 Thesis Organization

The organization of this thesis is as follows. All research questions and corresponding hypotheses, as well the structure of this thesis are listed in Figure 1.2.

A brief introduction of uncertainties in a control system is given in this chapter. Introductory questions are also brought out.

Chapter 2 contains respective physical and mathematical interpretations of control system design, in the sequence of overview, plant, and controller. The optimal control is introduced in Section 2.3.2 and the emphasis of this chapter is Linear-Quadratic Regulators (LQR), which defines the control system's performance evaluation criteria, as well a Lyapunov equation as the mapping from a selected controller to the calculated performance. Parameter uncertainty is then introduced in Section 2.4. It is also

highlighted to distinguish from the other two types of uncertainties (disturbance and sensor noise) via a comparison. Next, it is briefly analyzed and leads to the conclusion that parameter uncertainty will affect the system's physical nature and thus needs special treatment. Assumptions and nomenclatures to be used in future chapters are included in Section 2.5.

Chapter 3 particularly focuses on parameter uncertainty. A deeper analysis of the properties of parameter uncertainty is first provided, followed by a brief introduction of the state of art solutions (adaptive control and robust control) to parameter uncertainty. Next, a minimax control design method, as a particular type of robust control design method, is introduced as the baseline robust control design method for future reference.

Gap analysis can be found in Chapter 4. Two levels of conservatisms are observed from the traditional minimax control design. They are stated in Observation 1 and Observation 2. To reduce such conservatism, a two-stage conservatism reduction robust control design methodology is provided in Chapter 5 and Chapter 6.

Chapter 5 proposes a Norm Extended Minimax Controller Design to extend the norm-bounded uncertainty range with a purpose to reduce the first level of conservatism. Research question 1.1 and 1.2 respectively motivate the analysis about the properties of a performance curve over parameter uncertainty space and a worst-case performance curve over controller design space. The combination of the two leads to and answers research question 1.3 and 1.4, which concern the possible locations of minimax controller design point. Also based on research question 1.1 and 1.2, research question 1.5 extends research question 1.3 to higher dimensions and serves as the theoretical foundation of research question 1.6 and 1.7, which proposes a numerical gradient calculation method nested in a line search algorithm to calculate the global minimax controller.

With the calculated minimax controller and associated minimax performance as the evaluation criteria for POF estimation, Chapter 6 proposes a POF Constrained Optimal Average Performance Controller Design, essentially a numerical line search method to

design a controller that enables a tradeoff between POF and average performance to reduce the second level of conservatism. Research questions 2.1 and 2.2 provide methods for calculating average performance and a searching direction towards the optimal average performance design point. Research questions 2.3 and 2.4 propose a method to calculate POF and the search direction along the constraint POF contour. Research question 2.5 describes the indicative condition when the line search reaches the desired design point. A comprehensive algorithm is provided to fulfill the proposed methodology.

A comprehensive HVAC example is provided in Chapter 7, where a physical model is built and uncertain parameters are identified. The proposed method is then applied and a controller is designed to reduce the conservatisms.

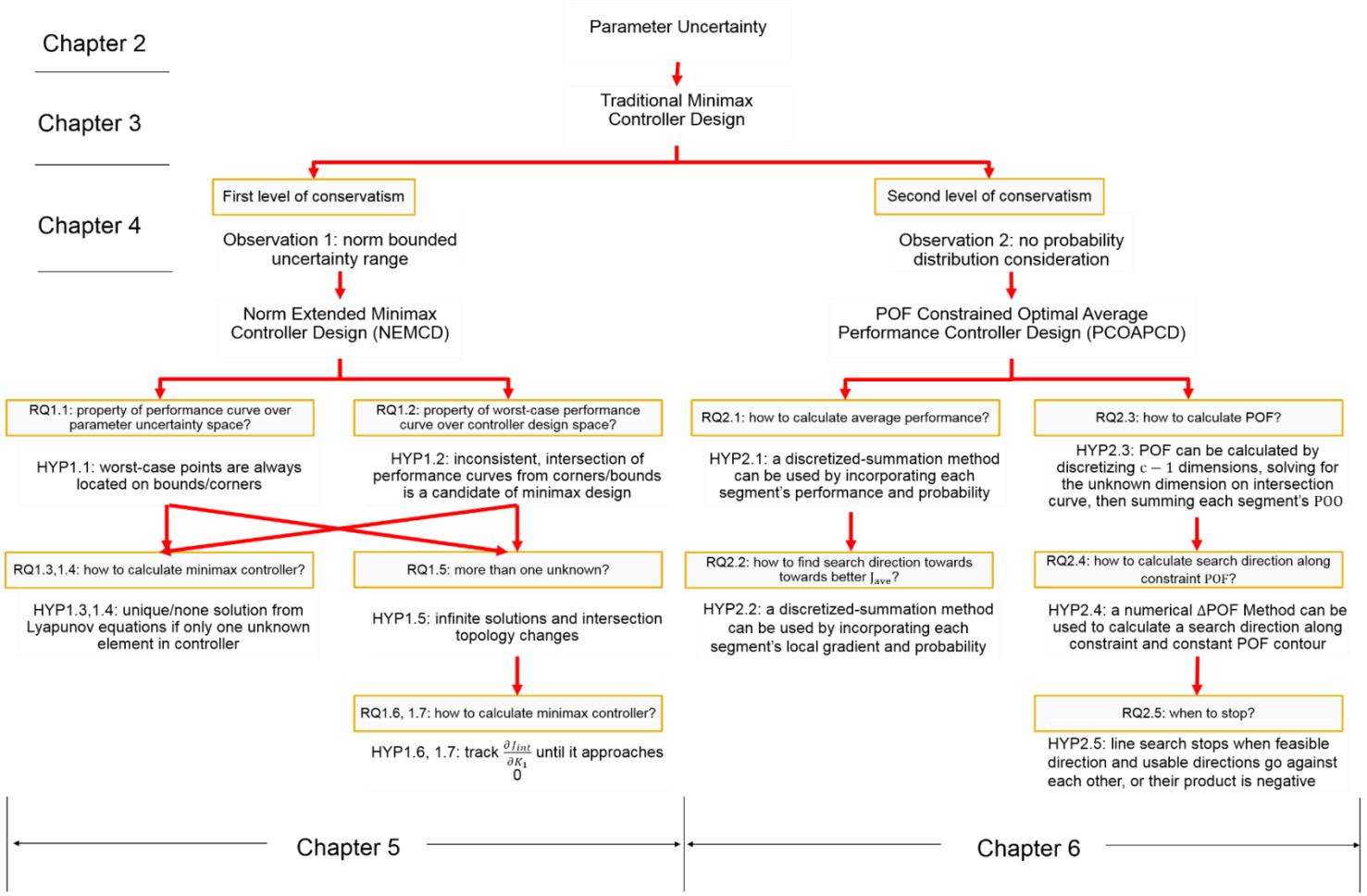


Figure 1. 2 Two-stage Conservatism Reduction Methodology

CHAPTER 2

BACKGROUND

In this chapter, the background of the control system will be elaborated, by the sequence of system overview, plant, and controller. Built on this, it proceeds to the introduction of uncertainties in the system. The focus is then put on parameter uncertainty.

2.1 Basic Control System

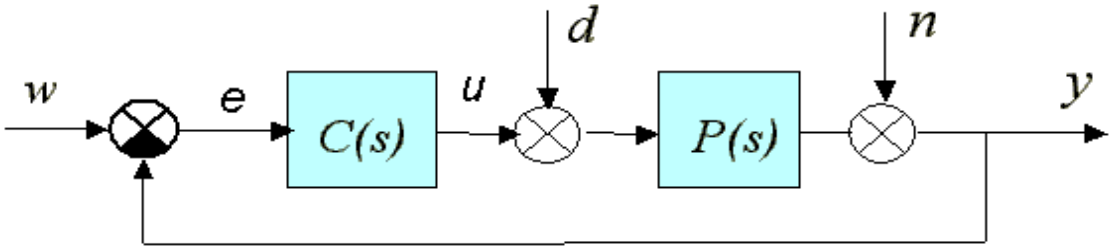


Figure 2.1 Basic feedback control system

A basic control system with feedback structure is exhibited in Figure 2.1. While open-loop, feedforward structure exists as an alternative to a feedback structure, most control systems utilize the feedback structure. Compared to a feedforward structure, a feedback structure has multiple advantages: it achieves stability more easily, a better signal track performance, less dependence of model's accuracy level, etc. [3].

Besides feedforward/feedback, with the variations of plant's physical features, the control system can be categorized in many ways, e.g. linear/non-linear, MIMO/MISO/SIMO/SISO, continuous/discrete. With variations from control objective, feedback, and controller type, the control system has subareas like optimal control, preview control, adaptive control, robust control, etc. There are also many evaluation criteria associated with control systems such as system stability, time domain and frequency domain response performance, and control effort (energy consumption).

In Figure 2.1, the two boxes (***Plant*** and ***Controller***) are linked by multiple signals including controller output u, measurement y, and error signals e. The *Plant* models the physics of the monitored target. There are one or more state variables embedded in the plant that needs to be controlled. The *Controller* models the physical actuator. These two components are the most important in a control system and will be further introduced in the following sections.

2.2 Plant

The function of the plant is to take in output from the controller and outside disturbance and calculate the state variables' values at the next time point. The state variables are measured from the sensor and then delivered to the controller. In real life, usually the control system suffers from a outside disturbance which acts on the plant and noise which acts on the sensor.

The plant's mathematical representation equation is usually derived from the physical law of the monitored target such as the law of energy conservation, Newton's second law of motion. Generally, what happens in a linear plant can be described in the next set of linear equations,

$$\dot{x} = Ax + Bu + Cd \tag{2. 1}$$

$$y = Dx + w \tag{2. 2}$$

where u is the output signal from the controller. x is the state variables that needs control. d is the disturbance. y is the measured signal from sensor. n is the sensor noise. The coefficients A, B, C, D are derived physically to match the units and dimensions.

2.3 Controller

The purpose of the controller is to take in measured signal and then deliver necessary control effort to make the plant function as desired. Take the feedback control loop for example, the input into the controller is the feedback signal, i.e., the error signal between

the measured state variable and the reference signal. The output of the controller is then delivered to the actuator such as a motor, fan and valve, and the actuator then delivers appropriate control effort into the plant. The dynamics of the actuator is usually integrated into the controller's transfer equation.

While there are multiple controller types available, the controllers can be represented by the following equation,

$$u = F_1(s)F_2(y, r) \tag{2. 3}$$

where $F_1$is Laplace equation in s domain and F_2 is a function of sensor measurement and reference signal.

So far, how the controller works is identified which enables it to proceed to controller types and controller design methods. While there are many types of controller and associated controller design methods, ***PID controller*** and ***Optimal Controller*** are selected as the typical controller and the controller design method reviewed in this research since they are widely used.

2.3.1. Proportional-Integral-Derivative (PID)/Proportional-Integral (PI) Control

The PID controller dates backs to the 1900s and first got application in ship steering control [4]. With the appearance of electronic and programmable logic controllers, the PID controller got wide applications in all fields such as industrial manufacture, aircraft, and HVAC.

The mathematical description of PID controller is given in Equation (2.4) and Equation (2.5). The first one is expressed in time domain while the second one is transformed into Laplace domain. The three coefficients K_p, K_i, K_d represent proportional, integral, and derivative gain.

$$u = K_p e + K_i \int_0^t e\, d\tau + K_d \frac{d}{dt} e \tag{2. 4}$$

$$\frac{u(s)}{e(s)} = K_p + \frac{K_i}{s} + K_d s \tag{2. 5}$$

Generally, the PID controller has been considered the best controller historically. By tuning the three parameters in the PID controller algorithm, the controller can provide control action designed for specific process requirements [5]. It is also well recognized that in some applications not all the three parameters are needed to achieve certain system performance requirement.

2.3.2 Optimal Control

Different from PID controller, optimal control specifically focuses on minimizing a control cost, which is a function of state and control variables. Essentially, the method is to find such a control law by solving a set of differential equations. A ***Linear-Quadratic-Gaussian*** (LQG) control design method is widely used.

With a control system structure defined in Equation (2.6) and Equation (2.7), LQG aims to minimize control cost of a control system suffering from additive white Gaussian system disturbance d and additive white Gaussian measurement noise n by implementing a quadratic cost function J (as function of K), as shown in Equation (2.9).

$$\dot{x} = Ax + Bu + Cd \tag{2.6}$$

$$y = Dx + n \tag{2.7}$$

$$u = -Kx \tag{2.8}$$

$$\min.\ J(K) = E(x^T F x + \int_0^T x^T Q(t) x + u^T R(t) u dt) \tag{2.9}$$

where Q and R are weight coefficients. The controller gain K serves as the design variable in this research.

One necessary step is to "filter" the real plant state variable x and increase measurement's fidelity from "polluted" sensor measurement. In this manner, the LQG is composed by two components: a ***linear-quadratic estimator*** (LQE), i.e., a Kalman filter and a ***linear-quadratic regulator*** (LQR). The two serve as dual to each other, plus that LQE is not the primary research target in this research, no further elaboration will be given on this topic.

Sometimes the initial/final state variable is removed from Equation (2.9). Also it is assumed that there is no disturbance d and sensor noise n, Q and R are constant, Equation (2.9) reduces to Equation (2.10)

$$minimize\, J(K) = \int_0^T (x^T Q x + u^T R u) dt \qquad (2.10)$$

The linear-quadratic performance J has the following mathematical meaning in terms of performance evaluation criteria. The three are equivalent provided that the linear system is asymptotically stable:

1. the L_2 norm of the impulse-response function;
2. the mean-squared stochastic response;
3. the H_2 norm of transfer function.

$J(K)$ is calculated via an Algebraic Riccati or Lyapunov equation, shown in Equation (2.11), depending on whether the designers want to achieve optimality or not,

$$(A + BK_*)^T P_* + P_*(A + BK_*) + Q + K_*^T R K_* = 0 \qquad (2.11)$$

where P_* is a symmetric matrix with same dimension of A. Above equation also serves as the stability constraint for the system. $J(K)$ is then calculated as follows,

$$J = x_0^T P x_0 \qquad (2.12)$$

where x_0 is initial state variable.

The Algebraic Riccati /Lyapunov equation can be analytically solved from Kronecker matrix algebra (Appendix I), which basically rewrites Equation (2.11) into a linear form a$x = b$ and solves for x.

2.3.3 Important property of LQR

After briefly introduced in above section, it is still necessary to revisit some properties of LQR which will be frequently used in future chapters.

First, some well-known and strictly proved conclusions are visited: convexity and global optimality. While there are multiple ways to reach Riccati equation which essentially solves the LQR, such as dynamic programming, a Lagrange multiplier method

is used in this thesis. Following equations are cited directly from [6], no further elaborations are provided for new symbols.

Property 1: gradient and convexity

Let the Lagrangian be defined as

$$L(K, P_r, Q_r, \lambda_0) = \mathrm{tr}[\lambda_0 R_r Q_r + (A_r Q_r + Q_r A_r^T + D_1 D_1^T) P_r] \quad (2.\ 13)$$

Take the first order and second order derivatives,

$$\frac{\partial L}{\partial K}|_{(K_*, P_*, Q_*)} = 2R_2 K_* Q_* + 2B^T P_* Q_* \quad (2.\ 14)$$

$$\frac{\partial^2 L}{\partial K^2}|_{(K_*, P_*, Q_*)} = 2R_2 Q_* \quad (2.\ 15)$$

From definition, R_2 is positive defined; to ensure system is stable, Q_* satisfies the following Lyapunov equation and is also positive defined.

$$(A + BK_*)Q_* + Q_*(A + BK_*)^T + D_1 D_1^T = 0 \quad (2.\ 16)$$

Thus, the RHS of Equation (2.15), or Hessian Matrix, is also positive defined. Thus the performance J's response WRT controller K is strictly convex.

Property 2: optimality

The definition of LQR is that there exists K_{lqr} who uniquely gives the optimal performance J. It can be directly solved from Equation (2.14) so that

$$K_{lqr} = -R_2^{-1} B^T P \quad (2.\ 17)$$

where P satisfies the following Lyapunov equation

$$(A + BK_*)^T P_* + P_*(A + BK_*) + R_1 + K_*^T R_2 K_* = 0 \quad (2.\ 18)$$

2.4 Uncertainty Identification

The quality of a traditional control system is rooted in the validity of the mathematical models used in its design, the fidelity of the information it receives, and the health of its actuation devices. That is why in many cases, system's performance is degraded when the validity of the mathematical model fails to meet the real one. Only when the model is

a good match to reality, the sensors and actuators are functioning as expected, uncertainty in the system is low, the system behaves as designed and predicted.

However, control systems do not always perform as their models would predict — due to inaccurate parameters estimation, online system faults and as a result, system performance degrades and mission effectiveness is reduced. When the control system is viewed from a state-space equation point of view, there are three types of uncertainties in the system: disturbance d, sensor noise n and parameter uncertainty. When the control system is represented in Equation (2.1), the parameter uncertainty stands for the parameter A, B, C changes through the simulation time. Such system is identified as ***Time-varying Control System***.

Compared to the disturbance and noise, the parameter uncertainty is not considered in either PID or optimal controller design methodology. When viewed from a system level, such uncertainty will change the physical nature of the control system.

***Argument 1*:** Parameter uncertainty will change the physical nature of the control system.

Proof of Concept: If a Laplace transformation of the control system in Equation (2.1) is performed, the two input/output equations are shown below.

$$\frac{x}{u} = \frac{B}{s-A} \tag{2. 19}$$

$$\frac{x}{d} = \frac{C}{s-A} \tag{2. 20}$$

Classic control theory states that a system's performance, which could be either stability, transient response, disturbance rejection or energy usage, depends on open/closed loop transfer equation. When the disturbance or sensor noise presents in the system as d, n change with time, the RHS of Equation (2.19) and Equation (2.20) still keep unchanged. On the contrary, when parameter uncertainty appears as A, B, C change, the RHS of Equation (2.19) and Equation (2.20) change, either in the numerator or denominator. In this manner, same control signal u will yield different state variable x.

Therefore, parameter uncertainty changes the physical nature of the system and thus the performance of the system with a pre-tuned controller.

The LQG optimal control is actually suitable for a time-varying control system design under the premise that the values of time-varying parameters are known. The equations used to solve such time-varying control system are the same with Equation (2.11)-(2.12); the only difference is that now the controller gain is not constant through simulation, but is updated with the same Equation (2.17), but new values of parameters A_t, B_t, C_t at every time point t. No new equations will be given in this thesis to avoid redundancy.

The premise of above statement is that the parameter uncertainty should be either deterministic or measurable (different from control system state's observability). If the parameters are un-measurable, then the time-varying LQG control is not applicable. Truth is, such idealized condition is rare in real life and leaves the time-varying LQG control not a viable choice. So this method is not considered in this thesis and other solutions will be evaluated in the next chapter to regulate system's performance under parameter uncertainty.

2.5 Uncertainty definitions and nomenclatures

Before any further analysis is given, several terms and nomenclatures are defined to fluent future chapters. Any models or examples to be analyzed in this thesis are mapped and represented in the form of state space (Equation (2.21) to Equation (2.22)). Note that to make the problem easier, neither observer equation/sensor noise nor plant disturbance is adopted here. It is easy to extend the conclusions derived in future chapters so that above ignored terms are also considered, as the commonality between L_2 and H_∞ has been proved in all textbooks [6]. For the moment, all uncertainty channels are assumed to be independent from each other. Controller matrix B is assumed to be constant. The system is assumed to be controllable for any possible uncertain parameter.

$$\dot{x} = Ax + Bu \tag{2.21}$$

$$u = -Kx \tag{2. 22}$$

$$J(K) = \int_0^T (x^T Q x + u^T R u) dt \tag{2. 23}$$

Here the normal plant matrix is denoted as $A_n \in \mathbb{R}^{n*n}$. Controller input matrix is defined as $B \in \mathbb{R}^{n*m}$. The controller given by default LQR method is denoted as $K_{lqr} \in \mathbb{R}^{m*n}$. The corresponding normal optimal performance is denoted as $J_{lqr} \in \mathbb{R}^1$ and is a scalar. Note that unless specified otherwise, when a performance is mentioned in future chapters, it refers to the quadratic performance J from Equation (2.23). In this context, a better or equivalently smaller performance means a smaller value of J; a worse or equivalently larger performance means a larger value of J; the optimal performance means the smallest value of J over all controller design space.

The closed loop system is described as

$$A_k = A + BK \in \mathbb{R}^{n*n} \tag{2. 24}$$

The real plant that deviates from the normal when the parameter uncertainty is observed is denoted as $A_v \in \mathbb{R}^{n*n}$

$$A_v = A_n + \sum_{i=1}^{c} \Delta A_i \tag{2. 25}$$

where the parameter uncertainty matrix $\Delta A_i \in \mathbb{R}^{n*n}$ has the following structure: for each $\Delta A_i, \forall i = 1 \dots c$, there is only one non-zero real number $\Delta a_i \in [a_i^l, a_i^u] \in \mathbb{R}^1$ in the matrix and the cell that contains such non-zero real number could be anywhere.

$$\Delta A_i = \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & \ddots & \vdots \\ 0 & \cdots & 0 \end{bmatrix} \begin{matrix} \\ \Delta a_i \\ \\ \end{matrix} \in \mathbb{R}^{n*n} \tag{2. 26}$$

To distinguish it from the definition of matrix's dimension, each parameter uncertainty Δa_i and the corresponding range $[a_i^l, a_i^u]$ define a "$channel$". c is used to denote the total number of ΔA_i, or equivalently the number of channels in matrix A. The combination of all channels forms the uncertainty space U, e.g., if a $2 * 2$ plant $A_n = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$ has

two uncertain elements A_{11} and A_{12}, then it has two uncertainty channels ΔA_{11} and ΔA_{12}. The uncertainty space is defined as

$$U \equiv \{\Delta A_{11}, \Delta A_{12} | \Delta A_{11} \in [a_{11}^l, a_{11}^u], \Delta_{12} \in [a_{12}^l, a_{12}^u]\} \quad (2.27)$$

In future chapters, for clarification purposes, sometimes the subscript of uncertainty channels is replaced by (i, j) in the matrix when the position of uncertainty channel ΔA_{ij} is known. Also note that due to realization, the total number of uncertain elements in A not necessarily equals to the total number of uncertain parameters in the physics equations, though the two have the same physical meaning and can be easily transformed through realization. Since it is not the interest of this research and to avoid confusion, when the parameter uncertainty is mentioned in Chapters 3, 4, 5 and 6, no physics equations will be referred; instead, it only refers to the uncertain elements in A.

The mathematical meaning of above structure is that all parameter uncertainties can be expressed as the variations offset from the normal values of one or multiple elements in A. Equation (2.25) does nothing but simply decomposes all parameter uncertainties into individual "channel". Such channels could have different magnitude and sign; could be correlated or not. The term "uncertainty structure" will be used purposely in future chapters to describe the positions and properties of these uncertainty parameters.

At the same time, the controller design space can be represented as follows. The only requirement associated with it is that the selected controller ensures the plant is stable.

$$S \equiv \{K_1, K_2 \dots K_n | A + BK < 0\} \quad (2.28)$$

From a design space exploration point of view, when the term "point in the uncertainty space U" or "point in the controller design space S" is mentioned in future chapters, the word "point" means a specified system space state matrix A_v whose uncertain elements are defined in Equation (2.25), or one specified controller in the controller design space S defined in Equation (2.28).

With above definitions, the system to be dealt with in this thesis can be written in the following manner,

$$\dot{x} = (A_n + \sum_{i=1}^{c} \Delta A_i)x + Bu \quad (2.\ 29)$$

$$u = -Kx \quad (2.\ 30)$$

$$J(K, \Delta A_1, \dots \Delta A_c) = \int_0^T (x^T Q x + u^T R u) dt \quad (2.\ 31)$$

For reference, a system who suffers from parameter uncertainty is labeled as $nDcC$, where n denotes a n by n matrix A representing the system dynamic characteristic and there are c uncertainty channels. The following $1D1C$, $2D1C$ and $2D2C$ examples will be used frequently in future chapters to provide visual examples for hypotheses proofs. Their normal plants as well uncertainty channels are summarized below so that there will be no need to re-describe them when used.

$1D1C$ example:

$$\dot{x} = A_v x + u = (1 + \Delta a)x + u$$

$$\Delta a \in [-0.5, 0.5], Q = R = 1, x_0 = 1$$

$$K_{lqr} = -1.414$$

For the following examples, all $Q = I^{2*2}$, $R = 1$, $x_0 = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$.

$2D1C$ example1:

$$\dot{x} = A_v x + u = \begin{bmatrix} 1 + \Delta a & 1 \\ 0 & 1 \end{bmatrix} x + \begin{bmatrix} 1 \\ 1 \end{bmatrix} u$$

$$\Delta a \in [-0.5, 0.5]$$

$$K_{lqr} = [5.0273 \quad -0.4142]$$

$2D1C$ example2:

$$\dot{x} = A_v x + u = \begin{bmatrix} 1 & 1 + \Delta a \\ 0 & 1 \end{bmatrix} x + \begin{bmatrix} 1 \\ 1 \end{bmatrix} u$$

$$\Delta a \in [-0.1, 0.1]$$

$$K_{lqr} = [5.0273 \quad -0.4142]$$

$2D2C$ example1:

$$\dot{x} = A_v x + u = \begin{bmatrix} 1 + \Delta a_1 & 1 + \Delta a_2 \\ 0 & 1 \end{bmatrix} x + \begin{bmatrix} 1 \\ 1 \end{bmatrix} u$$

$$\Delta a_1 \in (-1, 1.5], \Delta a_2 \in [0, 0.6]$$

$$K_{lqr} = [5.0273 \quad -0.4142]$$

$2D2C$ example2:

$$\dot{x} = A_v x + u = \begin{bmatrix} 1 & 1 \\ 0 + \Delta a_1 & 1 + \Delta a_2 \end{bmatrix} x + \begin{bmatrix} 0 \\ 2 \end{bmatrix} u$$

$$\Delta a_1 \in [-0.5, 0.5], \Delta a_2 \in [-0.5, 0.5]$$

$$K_{lqr} = [3.5201 \quad -0.6180]$$

For a plant with a matrix A whose size is larger than 2 by 2, the corresponding performance over controller design space requires at least 3 dimensions view and is not viable in any plot and thus not considered as illustrative example here. However, there will be a high dimension HVAC control system design used as a case study in Chapter 7.

CHAPTER 3

SOLUTIONS TO PARAMETER UNCERTAINTY

The takeaway from Chapter 2 is that the parameter uncertainty might degrade system performance if not treated carefully. In this chapter, system performance's properties such as curvature and monotonicity will be first examined, followed by two types of solutions to parameter uncertainty: adaptive control and robust control, surveyed from a literature search. In the end, a particular type of robust control design method, the robust minimax controller design method will be studied and treated as the reference in future research.

3.1 Analysis of properties of parameter uncertainty

3.1.1 Mapping between controller and uncertainty

When considered from a closed loop controller tuning point of view, for a system described in Equation (2.29), its system dynamics "solely" depends on the closed loop dynamic matrix, as shown in Equation (3.1),

$$A_{cv} = A_v + BK_{lqr} + \sum_{i=1}^{c} \Delta A_i \tag{3. 1}$$

Argument 2: Under some circumstances (see following), parameter uncertainty and controller tuning are "equivalent": there exists a mapping from uncertainty space U to controller design space S. Alternatively speaking, given the dimension of K and the structure of parameter uncertainties, there exists a corresponding $\Delta K \in \mathbb{R}^{m*n}$ so that the following equation is satisfied for each point from the uncertainty space U,

$$\sum_{i=1}^{l} \Delta A_i = B\Delta K \tag{3. 2}$$

Proof of concept: It's an easy proof since their "impacts" on system dynamic matrix are **linearly** added, or it is impossible to identify whether "impacts" are contributed from controller or parameter uncertainty. In such case, there will be a unique "mapping" that

"links" the controller design space S to the parameter uncertainty space U. The following example is provided to help understand.

Example 3.1: In $2D2C$ example 2, for any uncertainty point within the space ($a_1 \in [-0.5, 0.5], \Delta a_2 \in [-0.5, 0.5]$) there exists one unique $\Delta K_1 \in [-0.25, 0.25]$ and $\Delta K_2 \in [-0.25, 0.25]$. The meaning of above statement is that for any variation contributed from the two uncertainty channels that acts on the normal plant A_n along with normal controller K_{lqr}, there exists a system composed of a normal plant A_n with a new but unique controller K_{new}, as shown in Equation (3.3): the two systems are equivalent with each other.

$$K_{new} = K_{lqr} + [\Delta K_1, \Delta K_2] \tag{3. 3}$$

$$\Delta K_1 \in [-0.25, 0.25], \Delta K_2 \in [-0.25, 0.25]$$

In such case, a lot of efforts can be saved since the properties of parameter uncertainty, such as system's stability, convexity, and optimality is equivalent with that of the controller. E.g., if a particular controller makes the system unstable, then the corresponding point in the parameter uncertainty space that satisfies Equation (3.3) will also make the system unstable; since the performance response WRT the controller is convex with the normal plant, then the performance response WRT the parameter uncertainties when a controller is given is also convex.

However, above circumstance actually has a strong requirement that the corresponding B that maps controllers to parameter uncertainty is non-singular, which is not usually the case. A counter example is provided below.

Example 3.2: For $2D2C$ example1, there is no mapping, or ΔK exists in the controller design space to make Equation (3.19) valid, as $\Delta K = \frac{\sum_{i=1}^{l} \Delta A_i}{B} = \begin{bmatrix} \Delta a_1 & \Delta a_2 \\ 0 & 1 \end{bmatrix} / \begin{bmatrix} 1 \\ 1 \end{bmatrix}$ leads to singularity. Occasionally there exist a B' that differs from B in terms of dimension and rank, but that would easily lead to a uncontrollable system.

When the system dimension increases, such chance further decreases and thus efforts are still needed when such mapping doesn't exist.

3.1.2 System performance WRT to parameter uncertainty

From a closed loop point of view, both controller and parameter uncertainty will affect system's performance. Due to the fact that the controller can be tuned and designed, it will be assumed to be "given" or at least fixed when the property of parameter uncertainty is analyzed, i.e., unless specified otherwise, the controller designed from the traditional LQR method with normal plant will be used as the default controller in future chapters.

Argument 3: How performance responses to parameter uncertainty depends on the position and sign of parameter uncertainty.

Proof of concept: First, re-formulate the uncertainty structure so that each uncertainty channel can be viewed as $1D$ variation,

$$\Delta A_i = \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & \Delta a & \vdots \\ 0 & \cdots & 0 \end{bmatrix} = \begin{bmatrix} 0 & \cdots & 0 \\ \vdots & 1 & \vdots \\ 0 & \cdots & 0 \end{bmatrix} \Delta a \qquad (3.4)$$

where I is the identity matrix and Δa is a scalar.

Next several examples are provided to support above argument.

Example 3.3: In $1D1C$ example with K_{lqr} pre-selected and specified parameter uncertainty, the closed loop system dynamic equation is shown in Equation (3.5),

$$\dot{x} = A_v x + u = (1 + \Delta a)x - 2.414x = (-1.414 + \Delta a)x \qquad (3.5)$$

$$\Delta a \in [-0.5, 0.5]$$

The eigen value is exactly calculated as $\lambda = -1.414 + \Delta a \in [-1.914, -0.914]$. Thus even with the parameter uncertainty, the system is still stable. In terms of performance J, it can be expected that a positive variation will degrade the system performance while a negative variation improves, as shown in Figure 3.1.

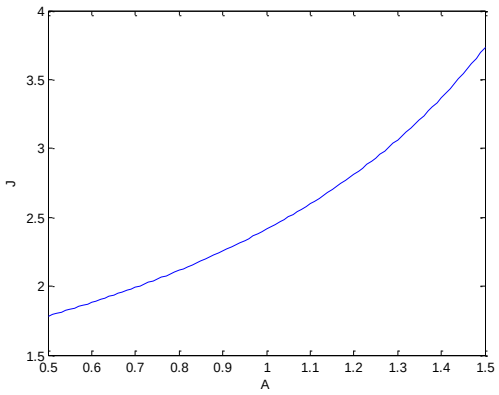


Figure 3.1 Performance over uncertainty space $\boldsymbol{1D1C}$

For system with a higher dimension, the position of parameter uncertainty matters in a way of affecting the dominant eigen value. E.g., in a 2*2 system dynamic matrix $A = \begin{bmatrix} A_{11} & A_{12} \\ A_{21} & A_{22} \end{bmatrix}$, the eigen values are given as

$$\lambda_{1,2} = \frac{A_{11} + A_{22} \pm \sqrt{(A_{11} - A_{22})^2 + 4A_{12}A_{21}}}{2} \qquad (3.6)$$

Given that $(A_{11} + A_{22})$ is negative to ensure stability, if $(A_{11} - A_{22})^2 + 4A_{12}A_{21} \ll A_{11} + A_{22}$, then a positive variation in either element A_{11} or A_{22} will equivalently degrade the system performance, since it pushes the eigenvalue towards the imaginary axis; if $(A_{11} - A_{22})^2 + 4A_{12}A_{21} > 0$, then the system has two poles on the real axis: a positive variation in either position A_{11} or A_{22} will not necessarily degrade system since it might decrease the value of $\sqrt{(A_{11} - A_{22})^2 + 4A_{12}A_{21}}$ and thus push the dominate eigenvalue away from the imaginary axis, as shown in the next example; for a variation in the element A_{12} or A_{21}, how it affects the system performance depends on the value and sign of the other and it is possible that such effect is not monotonic. It is also possible that the system goes unstable due to parameter uncertainty. Thus, there is no general conclusion about the performance WRT parameter uncertainty. Next, a numerical example is given to illustrate above concept.

Example 3.4: For $2D2C$ example1 with K_{lqr}, a sweep method is used to explore the uncertainty space and the result is shown in Figure 3.2. In the Δa_1 channel, the system performance decreases as Δa_1 goes towards negative bound and even becomes unstable when $\Delta a_2 = 0$ and $\Delta a_1 = -1$; the system performance also degrades as Δa_1 goes towards positive bound. On the other side, in Δa_2 channel within the given uncertain range, the system performance improves monotonically in the given uncertainty range as Δa_2 moves towards positive bound.

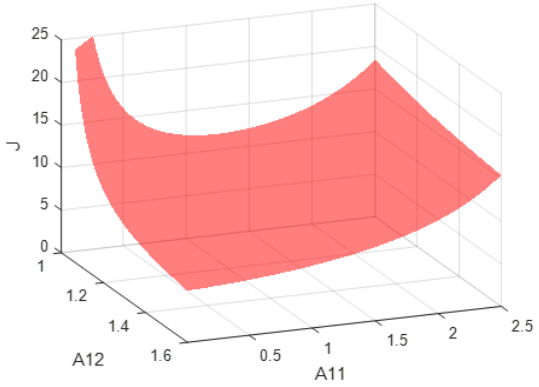


Figure 3.2 Performance over uncertainty space $\boldsymbol{2D2C}$

One other interesting observation is, the non-monotonic trend in Δa_2 channel gradually disappears as Δa_1 increases. This again demonstrates the non-consistent feature of the system performance WRT parameter uncertainty. When there are more than one uncertainty channels, they could get coupled with each other and make the system performance non-consistent through the whole parameter uncertainty space.

Above analysis yields the following facts: parameter uncertainty not necessarily degrades system performance; while it is a fact that parameter uncertainty is usually non-constant but a range of values, there is no one overall answer [7]. What is worse, such non-consistence brings difficulty in the controller design as there is no such controller that could ensure both robustness and optimality. All above analysis contributes to the

challenge to design a controller and such challenges will be further illustrated in future chapters.

In this context, adaptive control and robust control emerge as the state of art solutions to the parameter uncertainty.

3.2 Adaptive Control

As illustrated in Section 2.4, the time-varying LQG controller design method is naturally an adaptive control as the control gain is updated at each time point. The traditional adaptive control method, as a solution to parameter uncertainty, has less strong assumption since it does not require the uncertain parameters to be known.

The core idea of adaptive control is that every time the parameter uncertainty occurs in the plant and manifests itself through sensor measurement by deviating system performance from desired, the controller "dynamically" adjusts itself to "fit" such uncertainties to avoid performance degradation.

Adaptive control has many subareas and techniques. Two frequently used techniques are ***Model Reference Adaptive Controllers*** ($MRAC$) [8] and ***Model Identification Adaptive Controllers*** ($MIAC$) [9]. Besides $MRAC$ and $MIAC$, thanks to the modern computer and the digital programming techniques, there are multiple derivations developed from the concept of adaptive control such as fuzzy logic control [10], neural network control [11], and machine learning control [12].

Only the MRAC will be shown in this section since it is the most representative of the core concept of adaptive control. The MRAC aims to create a closed loop controller with a gain that can be updated to change the response of the system to "match a desired" model. To fulfill this, an error signal is first generated by comparing desired and actual model output. Then such error is penalized by performing a cost function J as a function of error to "enforce" the model to output desired results.

$$e = y_{plan} - y_{model} \tag{3. 7}$$

$$J(\theta) = \frac{1}{2}e^2(\theta) \quad (3.\ 8)$$

where θ is the parameter that will be adapted in the controller. Such parameter adaption procedure is depicted as the time gradient of parameter variation as a function of output error

$$\frac{d\theta}{dt} = -\gamma \frac{dJ}{d\theta} = -\gamma e \frac{de}{d\theta} \quad (3.\ 9)$$

In this manner, the parameter θ is constantly tuned to generated desired results.

3.3 Robust Control

When such "on-line" parameter tuning is not feasible, or the model is corrupted with noise and disturbance, then the capability of adaptive control is limited. Consider LQG control method for example, the accurate estimation of the plant outputs from noise is based on an accurate knowledge of the plant parameters. If there are plant parameter uncertainties as well as disturbance and noise, then it is impossible that one can tell exactly whether the performance variations are contributed from parameter uncertainty, disturbance or noise. In such case, robust control is needed.

The concept of robust control is to synthesize controllers achieving stabilization with guaranteed performance as long as uncertain parameters are within some sets. The most widely used design method is H_2/H_∞ method based on ***Small Gain Theorem*** [13].

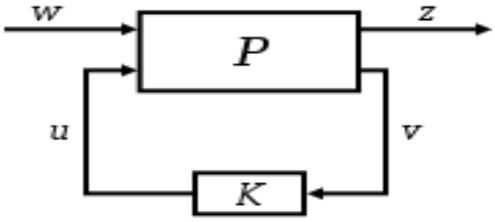


Figure 3.3 $\boldsymbol{H_2/H_\infty}$Method

H_∞ is the space of matrix-valued functions that are analytic and bounded in the open right-half of the complex plane defined by $Re(s) > 0$. H_∞ norm is the maximum

singular value of the function over that space, which is corresponding to the worst-case scenario.

$$H_\infty = \| F_l(P,K) \|_\infty = sup\overline{\sigma}(F_l(P,K)(j\omega)) \quad (3.10)$$

$$F_l(P,K) = P_{11} + P_{12}K(I - P_{22}K)^{-1}P_{21} \quad (3.11)$$

Robust control then minimize H_∞ norm to find the maximum allowable stable margin. H_∞ technique can be used to minimize the closed loop impact of a perturbation. Depending on the problem formulation, the impact will either be measured in terms of stabilization or performance.

But note that simultaneously optimizing robust performance and robust stabilization is difficult. So a lot of research is done WRT play with the balance between stability and performance. H_∞ Loop Shaping method overshadows H_∞ method by describing the desired performance (responsive and noise-suppression) through forcing a weight function into the transfer function [14].

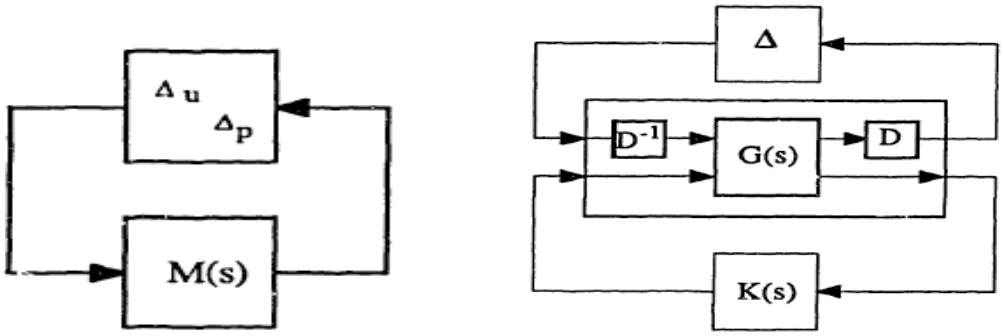


Figure 3.4 $\boldsymbol{U}$ Synthesis and D-K iteration

U synthesis overshadows H_∞ method since there are more conservatism in the latter method at some specific types of problem formulation (the small gain problem has a block diagonal structure) [14]. It uses structured singular value to reduce such conservatism. Instead of minimizing $H_\infty = \| F_l(P,K) \|_\infty$, the structured singular value can be written as

$$H_\infty = \| DF_l(P,K)D^{-1} \|_\infty \quad (3.12)$$

However, the mathematical calculation can be difficult. Usually, it is done through an iterative process by D-K iteration. However, the solution is not guaranteed to converge.

3.4 Remarks about control concepts

Note that although there are discrepancies between adaptive control and robust control, sometimes the two are mingled as "adaptive robust control" with tunable parameter and the robust performance as the control objective. In this thesis, robust control will be the main focus and the term such as "adaptive robust control" will not appear to avoid confusion. It is also worth mentioning that both adaptive control and robust control are controller design methods, rather than controller types, as both have applications on PID controller and optimal LQG controller [15].

3.4.1 Comparison between adaptive control and robust control

It is interesting to make a comparison between adaptive control and robust control, as both aim to design a controller for a system with uncertain parameters, but under different assumptions. Such comparison is shown in Table 1.

Table 1 Comparison between adaptive control and robust control

Category	Adaptive Control	Robust Control
Definition	Designs a controller which must adapt to a controlled system with parameters varying	Designs a controller to function properly so long as uncertain parameters are within some set
Structure	Changes control system structure	Remains control system structure
Feature	Adaptive control algorithm dynamically adjusts to the changing conditions	Robust control policy is static; rather than adapting to variations
Application	Effective when parameter uncertainty is measurable	Effective when parameter uncertainty is unmeasurable

The discrepancy is more evident when viewed from mathematical equations. Referring to the Equation (2.4) and Equation (2.30), adaptive control is to let K_p, K_i and K_d (K for LQG) be changeable, so that the system can still reach desired performance even A, B and C change. On the contrary, robust control is to pre-select the best setting of K_p, K_i and K_d (K for LQG) so that the performance variation due to A, B and C variations is minimized.

If the uncertainty has high and slow variation, then adaptive control is preferred [16]. Generally speaking, adaptive control needs constant tuning, which is effort and cost taking [17]. Even with digital programmable devices available, there is a need for extra components to output the compensation signal. Still, it has wide applications such as building energy systems, Adaptive Flight Control System (AFCS) [18].

If the uncertainty has low variation but a high frequency, then robust control is preferred [17]. Compare to adaptive control, robust control needs no tuning, but its performance is not as good as that of adaptive control. It also has wide applications such as robots design [19].

In this thesis, considering the fact that not all control systems have access to digital programmable devices, robust control will be used as the baseline control method and will be compared with the proposed control system design method later.

3.4.2 Comparison between traditional design and control system design

It would be easier to understand the concept of robust control through a comparison between traditional design method with noise variable (TDMNV) proposed in [20] and robust control, as shown in Table 2.

Table 2 Comparison between traditional design with noise variable and robust control

Category	Traditional design method with noise variables (TDMNV)	Robust control system design
Evaluation criteria	Performance feasibility Viability	Stability and others
Identify design Variables	Design variable range Noise variable distribution	Controller type Parameter uncertainty range
Evaluation methodology	Modeling MC Sampling & POF	Dynamic simulation System estimation Find response extremes
Design space exploration	Random (MC sampling)	Analytical robust control design from response extremes

For TDMNV, the noise variables which are in-deterministic in nature bring uncertainty into the performance response. Thus instead of using performance feasibility as measurement, viability or ***Probability of Failure*** (POF) can be used to accommodate the probabilistic feature. For robust control design, evaluation criteria remain unchanged.

When it comes to the second step, the probability distributions of noise variables from TDMNV are needed. Robust control only needs the ranges of uncertain parameters and the probability information is not necessary. Actually, it increases the conservatism of the calculated result for robust control, which will be highlighted in Chapter 4.

After probability distribution is investigated, designers can easily use the modeling and sampling techniques to calculate POF for one particular design point. For robust

control system design, besides dynamic simulation and system estimation which have been already included in traditional control design, response extreme (worst-case scenario) also needs to be calculated, such as the stability and performance extremes which are represented in the form of maximum norm H_∞.

There is a trick in the last step for TDMNV. If the evaluation criteria are still defined in the form of feasibility, then everything remains the same with design procedure without noise variables. If viability is used rather feasibility, sampling is the most straightforward method since it requires the least information of knowledge of the physical law for viability calculation. Otherwise, analytical and numerical methods can be applied only when a response surface equation can be built between design variables and corresponding viability. However for robust control, as the response extremes have already been located, the remaining work is only to design a controller to minimize the extremes.

3.5 Minimax Controller

Given that the parameter uncertainty could potentially degrade system's performance, the need and purpose of robust control are conceptually introduced in above section. The key concept is that robust controller design brings in robustness by ensuring that the system performance meets some evaluation criteria against all possible parameter uncertainty.

While there are many robust controller design methods developed to regulate system performance in various aspects, such as stability domain, frequency domain and time domain, the robust controller design methodology mentioned in future chapters is restricted to ***Minimax Robust Controller Design*** [21][22][23]. The designed controller from this method which guarantees system's quadratic performance J is consistent with the system performance definition utilized in Equation (2.23). A brief introduction of this method is given below.

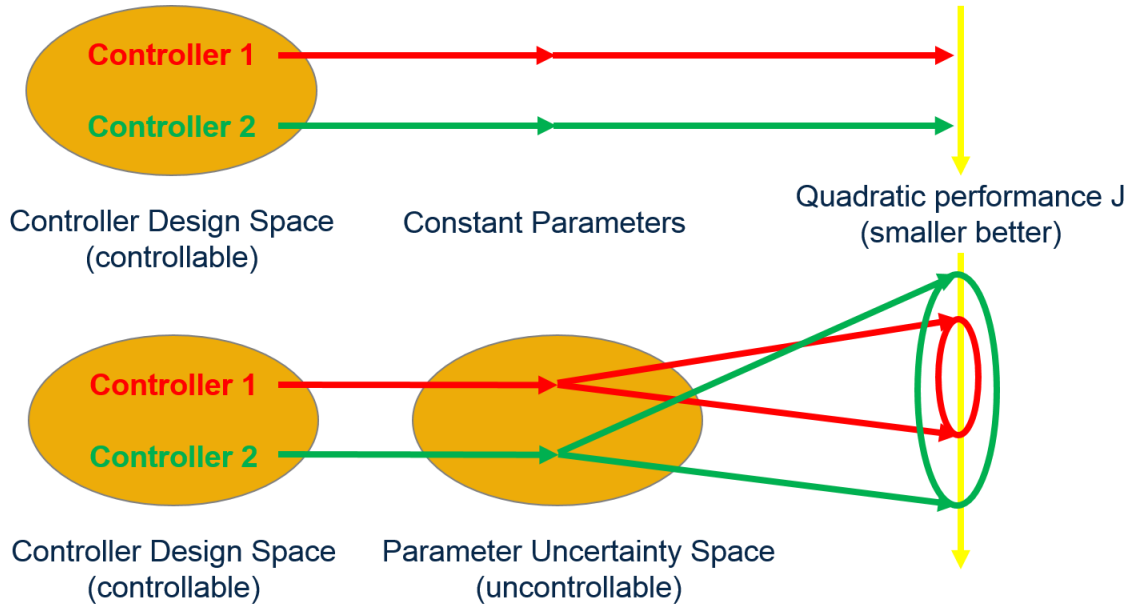


Figure 3.5 Performance perturbation from parameter's uncertain range

As shown in the first graph of Figure 3.5, if there is no parameter uncertainty, when a controller is selected, there will be a unique performance mapped towards quadratic performance J. On the contrary, when there is parameter uncertainty, if the same controller is applied, instead of deterministic value, now the system performances is no longer constant with different values of A. Instead, it has a corresponding performance range, as shown in the second graph of Figure 3.5. Though not deterministic, such performance uncertain range can still be calculated since the range of A is bounded (which is an important assumption in robust control). Now define the worst-case performance over the uncertain range of J as J_{max}, or equivalently the maximun value of quadratic performance J,

$$J_{max}|_K \equiv sup(J(A_v, K)) \tag{3. 13}$$

where the operator sup is short for $supreme$, denotes the upper bound.

With above definitions, the design of minimax controller is just to find the following

$$J_{minimax}|_{K_{minimax}} \equiv inf\left(sup\left(J(A_v, K)\right)\right) \tag{3. 14}$$

For better reference, let the controller K that gives $J_{minimax}$ denoted as $K_{minimax}$.

The "worst-case scenario" point J_{max} from robust control can always be found since the parameter A's uncertain range is bounded. Note that the value of J_{max} is not necessarily limited as for a given controller, there could be a value of Δa_i that leads to instability.

It is also helpful to take a quick look into the challenges of finding such minimax controller. Nominally, the general procedure to find the minimax controller is first to find the "maximum", or the worst-case performance point in the uncertainty space, then tune the controller in the controller design space to "minimize" it. However, as stated in Section 3.1, the impact of parameter uncertainty on system performance is non-consistent, meaning once the controller is tuned, the corresponding worst-case point also changes and thus the system performance. This suggests that the uncertainty space and the controller design space needs to be considered **simultaneously**, or at least **iteratively**. This concept is totally different from traditional LQR method where there is only one controller design space to explore. Thus the complexity increases and not necessarily there exists analytical solutions.

3.5.1 Traditional minimax controller design method (TMCDM)

In fact, the traditional minimax method falls into the category of considering two spaces simultaneously. For the method proposed by this research with the purpose of relaxing the norm bounded assumption, it falls into the category of considering two spaces iteratively, due to the in-consistent property of the location of the worst-case performance point.

The structure of the uncertainty used in TMCDM is that the norm of parameter uncertainty is bounded by a given factor, as described in Equation (3.15) [24],

$$|| \sum_{i=1}^{c} \Delta A_i || < 1 \qquad (3.15)$$

where several scale factors τ_i are utilized to represent each parameter uncertainty channel. A theorem is strictly proved to analytically calculate an augmented and lowest bound

reachable by any controller through a set of modified Riccati equations, as described in Equation (3.16),

$$J(K, \Delta a_i) = sup(x_0^{\mathrm{T}} P x_0 + \sum_{i=1}^{c} \tau_i^{-1} \Delta a_i) \qquad (3.16)$$

Next, a sweep method is performed through all possible combination of scale factors to find the contour of such augmented and lowest bound, as shown in Figure 3.6, which is generated from a $4D2C$ example problem. The smallest value is then selected as the minimax performance and the controller can be calculated accordingly.

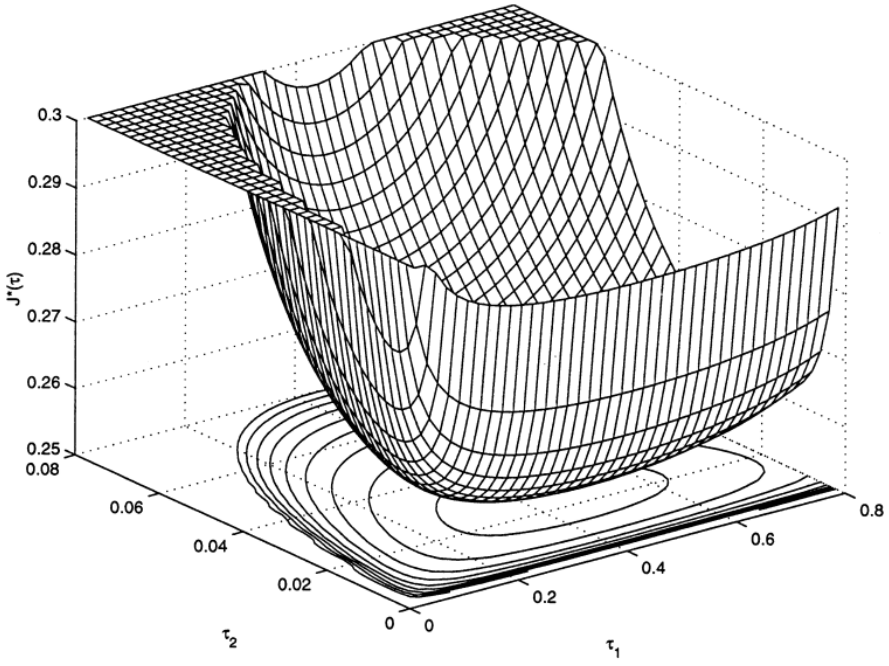


Figure 3.6 Sweep method for the augmented performance

CHAPTER 4

GAP ANALYSIS

4.1 Gap Analysis 1: First level of conservatism of TMCDM

4.1.1 Relax norm constraint

As stated in Section 3.3, a key assumption of the structure of uncertainty in TMCDM is that the norm of uncertainty is within certain specified bound. The consequence associated with this assumption is over conservatism. In real life, it is a very strong assumption since not necessarily uncertainties in multiple channels interact with each other. Most system representations try to decouple modes; in the physical plant, uncertainties could emerge anywhere and be uncorrelated at all. A more reasonable assumption should relax above norm-bound assumption but simply isolate and specify the range for each uncertainty channel to reduce the conservatism.

After the norm-bounded parameter uncertainty is removed, now the uncertainty structure is relaxed to exactly what is defined in Section 2.5. In this circumstance, the uncertainty space is expanded from a circle in two channels structure to a square, a sphere in three channels structure to a cuboid respectively, as shown in Figure 4.1. Such concept is easily extendable to higher dimensional cases.

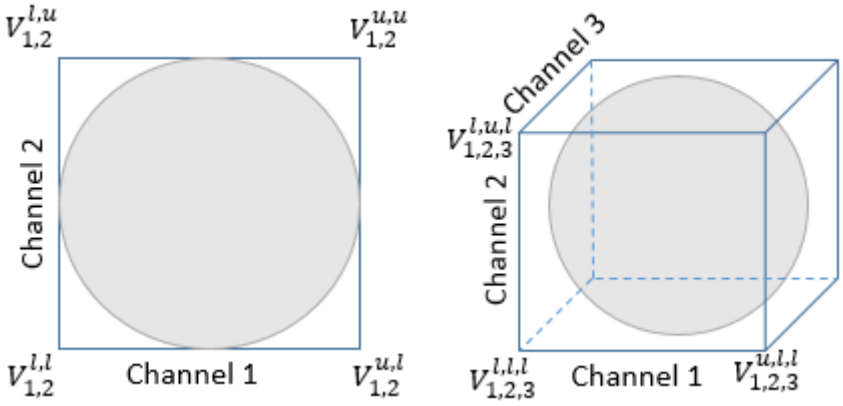


Figure 4.1 Relax norm bounds

The key difference distinguishes above two structures is: taking $2D2C$ case as example, instead of smooth boundary conditions, the extended structure now has multiple "corners" $V_{i,j}^{ul}$ to shape extremes. These corners represent the intersection point of two bounds in two channels i and j ($i \neq j$) over the uncertainty space. The superscript ul specifies whether the intersection points is from upper or lower bound of two channels. For a $2D2C$ case, it has four corners:

$$ul \in ([upper, upper], [upper, lower], [lower, upper], [lower, lower])$$

Without loss of generality, for c uncertainty channels, there will be 2^n corners, should there are no degenerated ones.

4.1.2 Remove scale term

On the other side, in Equation (3.16) the augmented and lowest bound with term $\sum_{i=1}^{c} \tau_i^{-1} \Delta a_i$ brings an extra level of conservatism into consideration, though the scale term's sign could be either positive or negative. In order to reduce the conservatism, such scale term should be removed from the optimization equation and leave the pure quadratic performance $J = x_0^T P x_0$ to be optimized. But very likely the removal of scale term will jeopardize the structure of Equation (3.16) and rule out the derived analytical solution.

The proposed norm extended minimax controller, to be introduced in the next chapter, particularly focuses on reducing such conservatisms, which is summarized in Observation 1.

Observation 1: *In Traditional Minimax Control Design Method, there are two sources of conservatism to be removed: norm-bounded uncertainty range and a scale term from the augmented optimization equation.*

4.2 Gap Analysis 2: Second level of conservatism of TMCDM

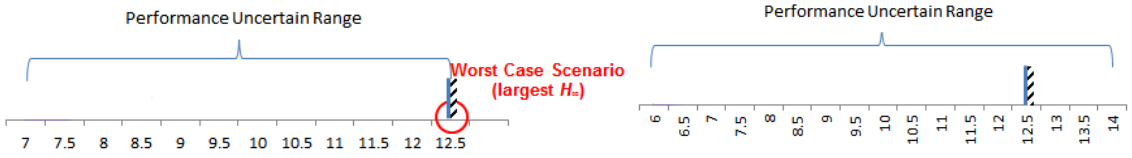
Not limited to above minimax control design method, the drawback of any worst-case based robust control design is that all uncertain values are given an equal likelihood of occurrence. Within these methods, the worst-case point is weighted equally with the most likely points in the uncertainty space. When designs are developed using norm-bounded uncertainties, systems often lack the performance characteristics that could be achieved for the most likely cases. The consequence of such design is another level of overly conservative.

A similar gap is observed by Brett A. Smith. In his NASA technical paper *"Probabilistic Parameter Uncertainty Analysis of Single Input Single Output Control Systems"* [25], he argues:

"The drawback of this approach (Robust Control) is that all uncertain values are given an equal likelihood of occurrence. Realistically most physical random variables have some sort of probabilistic distribution."

-------Brett A. Smith etc.

In this context, a control system with a controller particularly designed for the worst-case scenario is also unnecessary if the chance of a worst-case scenario is rare. In a word, incorporating probability information has the potential to reduce such conservation. A notional example of a controllers comparison depicted in Figure 4.2 and Figure 4. **3** would be helpful to make above concept intuitively clear.



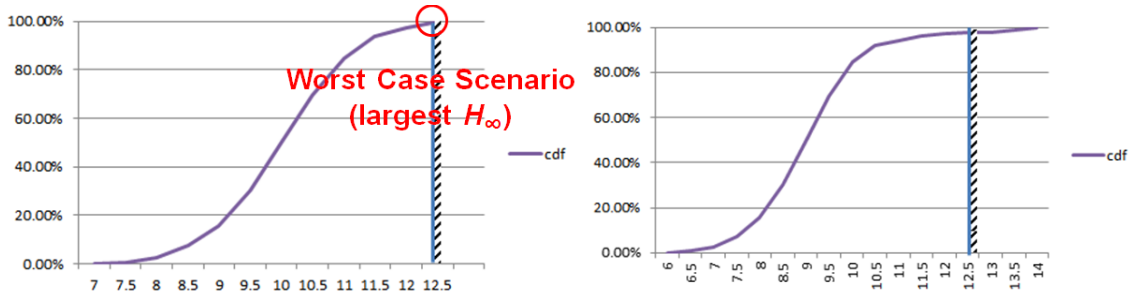
(a) Performance with controller 1 (b) Performance with controller 2

Figure 4.2 Performance range with no parameter probability information

Example 4.1: It is assumed that both controller 1 and 2 are applied in a system under variant values of A and the resulted H_∞ performance ranges are plotted in Figure 4.2(a) and (b). The point highlighted with red circle in Figure 4.2(a) corresponds to the worst-case performance or largest value over the performance uncertain range with controller 1. It can be guaranteed that all possible performance variations will always meet the evaluation criteria of 12.5. It suggests that should 12.5 is used as the evaluation criteria, controller 1 is robust against parameter uncertainty.

Similarly, with the same assumption, it is evident in Figure 4.2(b) that system's performance cannot always meet evaluation criteria with controller 2 applied as there are "outliers" at the right side of evaluation criteria, should the same 12.5 is used. Another way to interpret such concept is that to ensure that the performance meets criteria, the allowable parameter uncertain range of A would be potentially reduced. In this context, it can be claimed that controller 1 is better than controller 2 since it is more "robust".

On the other side, if the probability distribution of uncertain parameter A is known, then in terms of performance evaluation, not only the performance uncertain range can be calculated, but also its probability distribution. Such extra information is exhibited in the form of ***cumulative distribution function*** (CDF) curve and is shown in Figure 4. **3**.



(a) Performance distribution with controller 1

(b) Performance distribution with controller 2

Figure 4. 3 Performance distribution with parameter probability information

In this way, the probability that the calculated performance doesn't meet the evaluation criteria or equivalently ***probability of failure*** (POF) can be calculated

accordingly. In Figure 4. **3**(a) where controller 1 is used, with the assumption that uncertain parameter A has certain type of probability distribution, it can be expected that CDF curve crosses the performance evaluation criteria threshold right at 100% on the vertical axis. In Figure 4. **3**(b) where controller 2 is applied, the CDF curve and evaluation criteria threshold intersect each other at 97% on the vertical axis. So it can be claimed that with controller 2 applied, $POF = 3\% = 0.03$.

With the probability information available, it is also observed from Figure 4. **3** that the average performance equals to 9.9 with controller 1 and it equals to 9 with controller 2. So the averaged performance of controller 2 is better than that of controller 1. As long as the performance failure won't cause catastrophe, such as the quadratic performance, maximum overshoot value, it is reasonable to soften the "hard" evaluation criteria and instead, evaluate the performance in the form of POF. If it happens that POF is small, e.g., 3% in the previous example, a tradeoff between POF and average performance becomes reasonable and it is fair to argue that controller 2 is better than controller 1 since it provides a better average performance.

Similar examples can be easily found in many system designs, considering the fact that except for stability, the majority of evaluation criteria are not "hard" requirement. For example, one of the evaluation criteria for HVAC design is the steady-state error. Though it might cause slight uncomfortableness to the occupants due to the offset in the steady-state when parameter uncertainty is presented, a small tracking error of room temperature, e.g., 2 degree won't cause catastrophe or system to break down. Thus, a robust controller that tightly regulates the steady-state error is over conservative and unnecessary if the resulted POF is small.

Above analysis is summarized into the second observation:

Observation 2: *In robust control method, all uncertain values are given an equal likelihood of occurrence, so it lacks the performance characteristics that could be*

achieved for the most likely cases. The consequence of such design without probability consideration is overly conservative.

4.3 Proposed methodology

In the next two chapters, multiple steps will be presented to establish a comprehensive two-stage methodology to reduce such conservatisms, composed of a Norm Extended Minimax Controller Design (*NEMCD*) and a POF Constrained Optimal Average Controller Design (*PCOACD*), as shown in Figure 4.4.

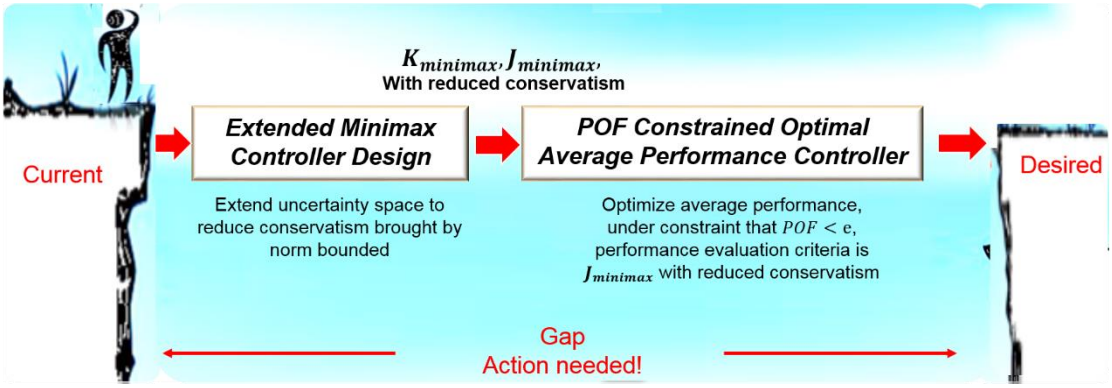


Figure 4.4 Proposed methodology

The purpose of NEMCD is to reduce the conservatism brought from the norm-bounded uncertainty assumption and the deliverable from this stage is a minimax controller and associated minimax performance with reduced conservatism, which will be feed into the second stage whose aim is to design a controller to optimize average performance under the constraint that POF is smaller than or equal to a given number.

In this context, the research statement of this thesis is described as follows:

Research statement: ***Based on traditional overly conservative minimax robust controller design, develop a two-stage conservatism reduction methodology, composed of Norm Extended Minimax Controller Design (NEMCD) and POF Constrained Optimal Average Performance Controller Design (PCOAPCD).***

The two-stage method will be presented in the sequence of problem's properties analysis, solutions proposed based on properties analysis and the exhibition of proposed solutions via numerical examples. When the properties are analyzed, concerns include existence, uniqueness, optimality, convexity, and stability; solution techniques will be investigated in the preference of analytical, numerical and sampling method. Complexity, computation time when dimensions increases and achievability will also be mentioned for the proposed solution. For each hypothesis, proof, and proposed method, there will be corresponding numerical examples, selected from $1D1C$ to $2D2C$ examples described in Section 2.5, in an order of increasing complexity. Finally, a comprehensive example of HVAC system will be provided to test the proposed methodology.

CHAPTER 5

NORM EXTENDED MINIMAX CONTROLLER DESIGN METHOD (NEMCDM)

In this chapter, a Norm Extended Minimax Controller Design (NEMCD) method, as the first stage of the proposed methodology will be introduced to reduce the first level of conservatism brought from the traditional minimax controller design method by extending the norm-bounded parameter uncertainty range, as stated in Observation 1.

5.1 Design objective and problem formulation

The general solution procedure of NEMCD is shown in Figure 5. 1. Notionally, after a controller is selected, e.g., $K = 1$, the performance curve over the parameter uncertainty space can be plotted, as highlighted in red in the upper plot. The worst-case design point with the worst-case performance, or equivalently the largest point on the curve can be found accordingly and is highlighted in a red dot. In above case where $K = 1$, the worst-case point is located on the leftmost bound. The controller design point and corresponding worst-case performance can be easily mapped into the lower plot whose horizontal axis is the controller design space.

Above procedure can be repeated in a continuous fashion until all points from the controller design space are visited, and thus the worst-case performance curve over controller design space can be plotted in a continuous fashion as well, as shown in the lower plot in Figure 5. 1. Then the best worst-case point, or equivalently the smallest point on the curve can also be found. Such controller and performance are exactly the minimax controller and minimax performance to be found. Such concept can be easily extended to higher dimensions, i.e., if the parameter uncertainty space is a $2D$ space, then the corresponding worst-case point is the largest point on the performance response surface over parameter uncertainty space.

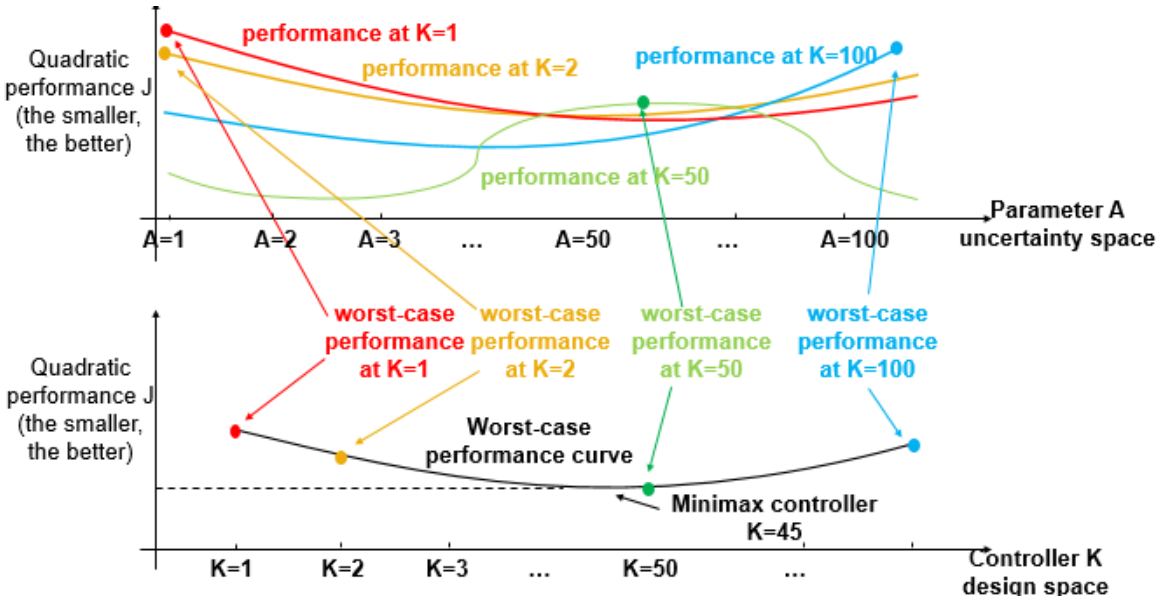


Figure 5. 1 General solution procedure for NEMCD

With above analysis, there are several questions to be answered:

1. What are the properties of such performance curve over parameter uncertainty space, so that the worst-case point can be located?
2. What are the properties of such worst-case performance curve over controller design space, so that the best worst-case performance point can be located?
3. How to calculate such best worst-case point?

5.2 Properties of worst-case point over parameter uncertainty space

RQ 1.1: what are the worst-case point's properties in terms of location, existence, uniqueness, etc.?

To answer RQ 1.1, Theorem 1, Theorem 2 and Theorem 3 are provided to introduce Hypothesis 1.1, which is the key element for NEMCD. Figure 5. 2 illustrates the proof process.

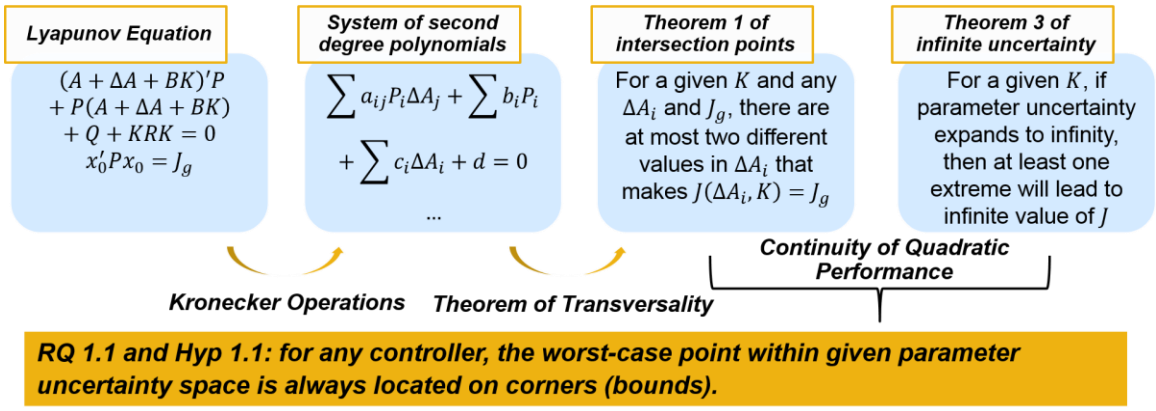


Figure 5. 2 Proof of Hypothesis 1.1

Theorem 1: For a given controller K, for any particular uncertainty channel ΔA_i (other uncertainty channels are fixed with pre-selected values), for any given value J_g, there are at most two different values in this uncertainty channel that make $J(\Delta A_i, K) = J_g$. Geometrically speaking, for a given controller K and above specified conditions, if any horizontal line $J = J_g$ is drawn, there are at most two intersection points with the performance curve over parameter uncertainty space.

Proof of concept: proving above theorem is equivalent with examining the number of real solutions of the following set of equations (Equation (5.1) and (5.2)), in which there is only one unknown element in ΔA.

$$(A+\Delta A+BK)^TP + P(A+\Delta A+BK) + Q + K^TRK = 0 \qquad (5.1)$$

$$x_0^TPx_0 = J_g \qquad (5.2)$$

Equations (5.1) is the standard Lyapunov equation, serving as the stability constraint. Equation (5.2) regulates the calculated performance equals to the desired value J_g.

Apply Kronecker operations (see Appendix I) on Equations (5.1), it converts above matrix equation into a system of second degree polynomial equations. Note that Kronecker operations convert a n by n matrix to a systems of equations with n^2 rows. Due to symmetry, it reduces to $n(n+1)/2$ unknowns and equations. The last unknown is introduced from the only unknown in the uncertainty channel, i.e., ΔA_{11}. Equation (5.2)

serves as the last balance equation. In this way, the total $\frac{n(n+1)}{2}+1$ unknowns match exactly the total $\frac{n(n+1)}{2}+1$ balance equations and thus Equation (5.1)-(5.2) can be solved directly.

Example 5.1: using $2D1C$ example 1 for reference and let the element in the i^{th} row and j^{th} column in matrix P denoted as P_{ij}. Note that due to symmetry $P_{12}=P_{21}$. Controller $K=[K_1,K_2]$ and J_g are given. There are four unknowns (P_{11}, P_{12}, P_{22} and ΔA_{11}) and four balance equations, and thus it is solvable.

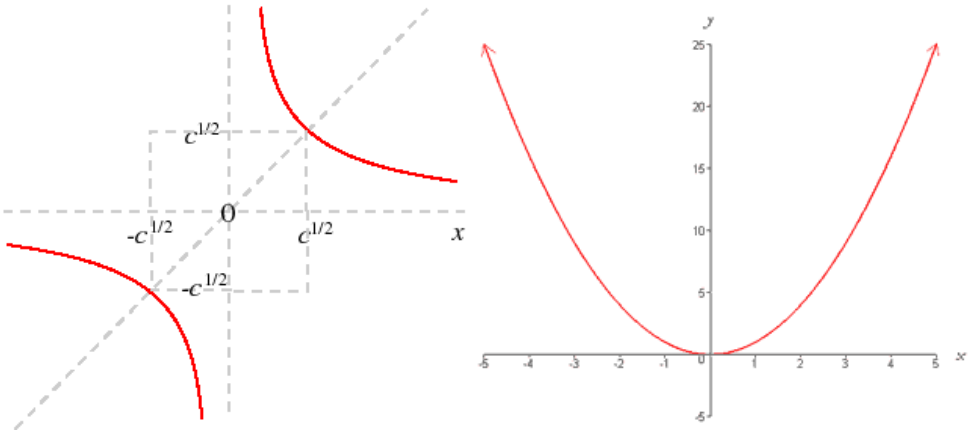
$$(2-2K_1+2\Delta A_{11})P_{11}-2K_1P_{12}=-K_1^2-1 \quad (5.3)$$

$$(1-K_2+\Delta A_{11})P_{11}+(2-K_1-K_2)P_{12}-K_1P_{22}=-K_1K_2 \quad (5.4)$$

$$(2-2K_2)P_{12}+(2-2K_2)P_{22}=-K_2^2-1 \quad (5.5)$$

$$P_{11}=J_g \quad (5.6)$$

Definition: the concept of ***topology manifold*** is introduced here to help the proof. It locally resembles the real *n*-dimensional space. E.g., a $2D$ space is a curve; a $3D$ space is a surface. Let the term ***unique manifold*** denotes that for any given $n-1$ dimensions, there is at most one value from the unknown dimension that makes the point be located on the topology space. It is easy to know that any closed manifold is not unique manifold. A notional example of a unique $2D$ space is plotted in Figure 5. 3(a). In Figure 5. 3(b), though the space is open, it is not unique since for a given value of $y>0$, there are two value of x make the point (x,y) rides on the curve.



(a) $y = \frac{c}{x}$ (b) $y = x^2$

Figure 5. 3 Unique manifold vs non-unique manifold

Theorem of Transversality: in differential topology, the intersection of any two n dimensional topology spaces, if exists, is a $n-1$ dimensional topology space [26].

Proof of concept: see reference [26].

Theorem 2(extension of Theorem of Transversality): the intersection of any two unique n dimensional topology spaces, if exists, is a unique $n-1$ dimensional topology space.

Proof of concept: assuming that the $n-1$ dimensional intersection topology space is non-unique. Then due to the fact that the whole intersection space must be located on both of the original topology spaces, it leads to the conclusion that the two n dimensional spaces are non-unique as well, which is against the initial definition.

(End of proof of Theorem 2)

Next Theorem 1 is to be proved. Turn back to the system of equations (5.3)-(5.6), except for the last linear equation, all others are second degree polynomials. There are no square terms such as Δa_i^2 or P_{ij}^2, but only interaction and linear terms. It can be extended to higher dimensions and above statement still holds true. That means for any given values of P_{ij}, all terms become linear and there is only one solved value of Δa_i. Thus these topology spaces but last one are open and unique. When the system of equations are

solved via the concept of topology space intersection, it reduces one dimension with the elimination of one second degree polynomial equation. Without loss of generality that there is no degenerated case, such system of equations will finally reduce to a two dimensional unique space, such as the one shown in Figure 5. 3(a) along with a two dimensional linear equation from last equation and the two have at most two intersection points and leads to Theorem 1.

(End of proof of Theorem 1)

Theorem 3: for any controller with gain limited, if the bounded parameter uncertainty expands to infinity, then at least one extreme will lead to infinity value of J.

Proof of concept: to calculate the eigen value of $A + \Delta A + BK \in R^{n*n}$ where $\Delta a \to \pm\infty$, the n degree polynomial contains at least one term $(\lambda_i - f(\Delta a))$ or $(\lambda_i^2 + g(\Delta a)\lambda_i + h(\Delta a))$. Here $f(\Delta a)$, $g(\Delta a)$ and $h(\Delta a)$ are functions of Δa and Δa doesn't appear on the denominator. Thus at least one of the eigen values will go to $+\infty$.

Hypothesis 1.1: with the uncertainty structure defined in Section 2.5, for any controller, the worst-case point within a given parameter uncertainty space is always located on corners (or bounds if $1D$).

Proof of concept: it is assumed that one point $(A + \Delta A_i)$ within the uncertainty space other than corner or boundary points yields the worst-case performance, or a global maximum J_{max}, then for sure it is a local maximum. That suggests that there exists two points $A + \Delta A_i^l$ and $A + \Delta A_i^r$ on the left and right hand side of the point $A + \Delta A_i$ respectively and the two give the same and smaller value of $J(A + \Delta A_i^l, K) = J(A + \Delta A_i^r, K) = J' < J(A + \Delta A_i, K) = J_{max}$. Thus the three points form a "mountain", as shown in Figure 5. 4.

Combine Theorem 3 and the fact that given stable, performance curve over parameter uncertainty space is continuous [27], there must be another point $\Delta A_i'$ between $[-\infty, A + \Delta A_i^l]$ or $[A + \Delta A_i^r, +\infty]$ makes $J(A + \Delta A_i', K) = J'$. Thus for a value of J', there are three

points over $[-\infty, +\infty]$ in the uncertainty channel that yield the same calculated performance value J', which is against Theorem 1. Thus the assumption that there is a local maximum is not valid; equivalently, the initial assumption that worst-case point doesn't ride on corners or bounds is not valid either.

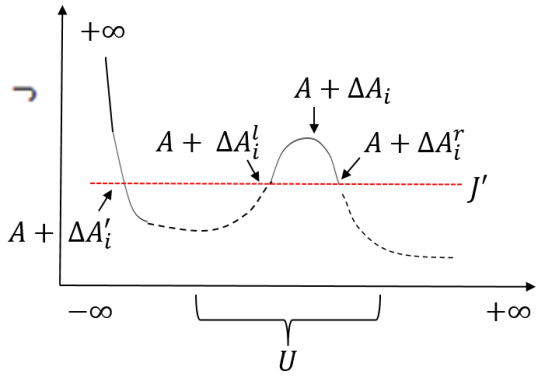


Figure 5. 4 Conceptual proof of Hypothesis 1.1

(End of proof of Hypothesis 1.1)

There might exist other proof, such as taking second order derivative $\frac{\partial^2 J}{\partial \Delta A^2}$, or equivalently ***Hessian matrix***, but above term ends up to a three-dimensional matrix if ΔA has more than one row or column and the concept of positive definite is no longer applicable here [28]. Additional effort is needed and will be addressed in future research.

In this context, the potential spots of the worst-case to be examined are located at the boxed corners or bounds. E.g., for $2D2C$ example1 in Figure 3.1, there are four corners to be examined.

$[\Delta a_1 = -1, \Delta a_2 = 0], [\Delta a_1 = -1, \Delta a_2 = 0], [\Delta a_1 = -1, \Delta a_2 = 0], [\Delta a_1 = -1, \Delta a_2 = 0]$

For c channels, there are 2^c potential corners. Note that due to the fact that neither the range of uncertainty nor the behaviors is symmetry in terms of normal plant, both upper and lower bounds in any uncertainty channel need to be examined and there is no implication from one to the other, i.e., when controller is tuned, the corner that gives the

worst-case point might change from one to others and thus every corner needs to be considered to make sure that there are no left overs.

5.3 Properties of best worst-case point over controller design space

5.3.1 Inconsistence of worst-case point

Hypothesis 1.1 hugely reduces the potential locations of the worst-case point to corners/bounds and brings out the next research question.

RQ 1.2: Whether such worst-case point is always located on one corner/bound consistently?

If RQ 1.2 holds true, then the uncertainty space and controller design space can be decoupled and thus the minimax controller design problem reduces to the taditional LQR design problem as long as the normal plant is replaced with the corresponding plant on that corner/bound where the worst-case point is located.

Hypothesis 1.2: The location of the worst-case point is not consistently on one corner/bound when K changes.

Instead of mathematical proof, an intuitive example is given below to illustrate Hypothesis 1.2.

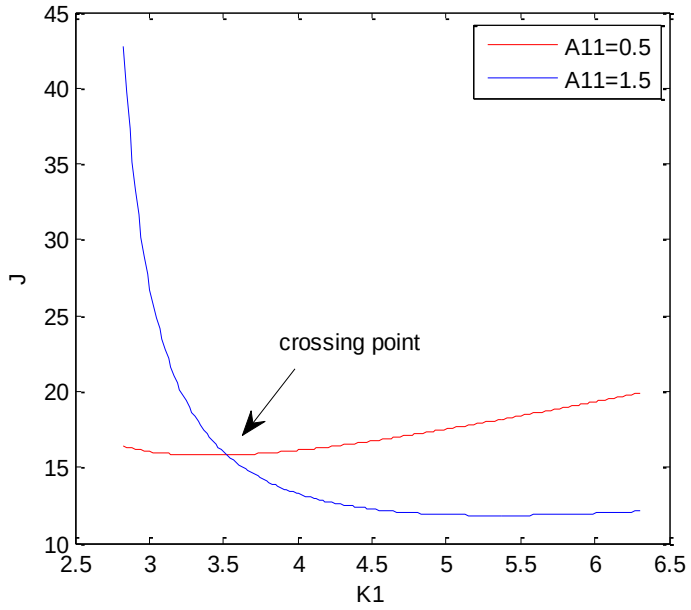


Figure 5.5 Performance curves at two bounds WRT K

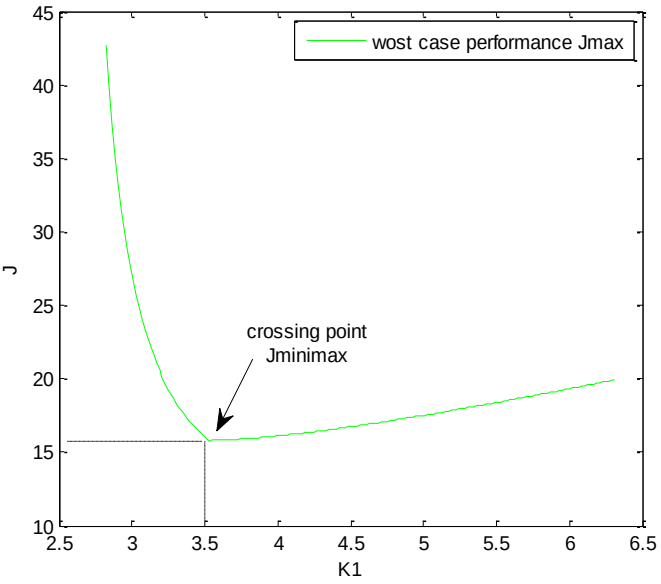


Figure 5.6 Worst-case performance WRT K

Example 5.2: In Figure 5.5, $2D1C$ example 1 is shown where $K = [K_1, K_1] \in \mathbb{R}^{1*2}$, K_2 is fixed at $K_2 = -0.1$ and K_1 sweeps from 2.8 to 6.3. The red and blue curves respectively represent the performance curves at bound $\Delta a = -0.5$ and $\Delta a = 0.5$. At the

bound where $\Delta a = -0.5$ and $A_v = \begin{bmatrix} 0.5 & 1 \\ 0 & 1 \end{bmatrix}$, the value of J increases, or equivalently the performance degrades as K_1 increases while the other bound where $\Delta a = 0.5$ behaves differently. The two curves intersect each other at a point where $K_1 \cong 3.5$ (pointed to by the arrow), so that the following equation is true at the intersection point,

$$J(A + \Delta A_1|_{\Delta a=0.5}, K_1, K_2 = -0.1) = J(A + \Delta A_1|_{\Delta a=-0.5}, K_1, K_2 = -0.1) \quad (5.7)$$

Above analysis also yields that given $K_2 = -0.1$, for $K_1 < 3.5$, the worst-case point or the "maximum" value of J to be captured in the minimax design is located at the corner where $\Delta a = 0.5$; while $K_1 > 3.5$, the worst-case point switches to the corner $\Delta a = -0.5$. Such fact of inconsistent location of worst-case point brings in the biggest challenge in NEMCDM. This feature is labeled as the ***inconsistency*** of the worst-case point and naturally distinguishes this method from TMCDM.

For a given controller K and a specified uncertainty space, assuming that there are no instable points from the uncertainty space, there is a curve of the worst-case performance J_{max}, as highlighted in light green in Figure 5.6 which shows nothing but the combination of the worse performance between two corners' performances from Figure 5.5. The case that both K_1 and K_2 are allowed to change is a more complicated and will be discussed in the next research question.

Note that such inconsistent worst-case point makes J_{max} naturally different from the quadratic performance J, which is continuous in both controller design and parameter uncertainty space.

5.3.2 Potential location of best worst-case point

Another interesting and helpful fact from above $2D1C$ example is that, given $K_2 = -0.1$, clearly $K_1 = 3.5$ gives a minimum point on the J_{max} curve with $J_{minimax} = 15.5$ and it is exactly the minimax performance to be found within K_1 design space exclusively.

Hypothesis 1.2 (continued): one potential solution of $K_{minimax}$ is the intersection point of two corners/bounds' performance curves over the controller design space.

How to calculate such intersection point and the properties such as existence and uniqueness of such type of intersection point will be addressed in the next few sections.

Note that the assumption that the intersection point is exactly where the worst-case point located is only valid when the local slopes of the two performance curves at the intersection point are of different signs. Otherwise, the local "trends" of two curves are the same and the intersection point is not a valid candidate of the worst-case point. Such concern will be added into check list when it comes to the local NEMCDM in Section 5.5.

5.4 Calculations and properties of performance intersection point/curve

5.4.1 Calculations of intersection point

RQ 1.3: For any selected pair of corners/bounds $A + \Delta A_1$ and $A + \Delta A_2$, if there is(are) intersection point(s), how to calculate it(them)?

Hypothesis 1.3: Depends on the knowledge of controller, e.g., should the controller $K \in \mathbb{R}^{m*n}$ has $m * n - 1$ values given, then the corresponding intersection point (the only remaining unknown value in the controller) can be calculated by solving a set of Lyapunov equations,

$$(A + \Delta A_1 + BK)^T P_1 + P_1(A_1 + \Delta A_1 + BK) + Q + K^T R K = 0 \qquad (5.\ 8)$$

$$(A + \Delta A_2 + BK)^T P_2 + P_2(A_1 + \Delta A_2 + BK) + Q + K^T R K = 0 \qquad (5.\ 9)$$

$$x_0^T P_1 x_0 = x_0^T P_2 x_0 \qquad (5.\ 10)$$

Proof of concept: The first two equations are just modified from the original Lyapunov Equation (5.1) at two corners when part of K is given. The last equation denotes that at the intersection point, performances at two corners equal to each other. Note that different from Equation (5.1) and (5.2), there is an extra Lyapunov equation in above system of equations and the extra unknown comes from the unknown value in the

controller instead of the parameter uncertainty channel. That leaves total $n(n+1)+1$ unknowns and balance equations. Thus the system of equations is still solvable.

Note that except for the last linear equation, all other rows in the system of equations have quadratic terms. There is one unknown K_i in K and it appears in the system of equations via the term $K^T RK$, in the form of K_i^2. It contributes to another difference from Equation (5.1) where there are no square but only interaction terms. An example of such system of equations will be provided later.

5.4.2 Existence and uniqueness of intersection point

***RQ 1.4*:** If there is only one unknown in the controller, will there always be an intersection point for any pair of corners? What are the necessary conditions and implications for those cases?

Hypothesis 1.4: There could be multiple cases and the number of intersection points ranges from 0, 1. It depends on the range of uncertainty, sign of first order derivative, initial condition, etc.

Proof of concept: Similar to the analysis in Section 5.2, after further simplification and elimination, one will end up with one quadratic equation with one unknown variable. Then all that matters is the ***discriminant*** $\sqrt{b^2-4ac}$ (a, b and c here are just placeholders, not to be confused with previously used notations) [29] which also determines the number of solutions. Next few cases list all possibilities of the discriminant.

Case1: $b^2-4ac<0$, or the system of equations ends up with a pair of complex conjugate solutions and thus the complex controller makes no physical sense. In such case, there is no intersection point at all. Alternatively, if it happens that the solved intersection point(s) fall outside of the uncertain range, then there will be no intersection point either, even it does exist.

Example 5.3: A simple $1D1C$ example is shown to support above analysis. Substitute all known parameters into Equation (5.8) to (5.10) and the system of equations reduces to the following,

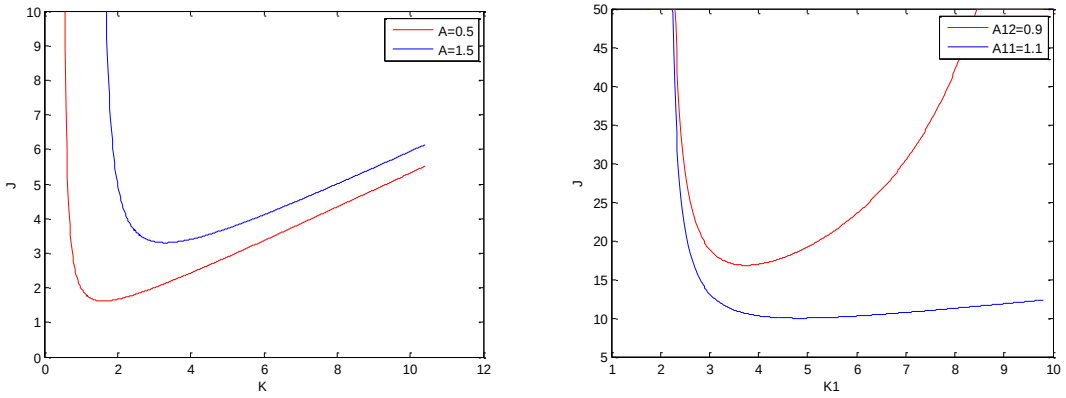
$$2(1.5 + K)P_1 + K^2 + 1 = 0 \quad (5.\ 11)$$

$$2(0.5 + K)P_2 + K^2 + 1 = 0 \quad (5.\ 12)$$

$$P_1 = P_2 \quad (5.\ 13)$$

It further reduces to $K^2 + 1 = 0$ and the only solution to above equation is $P_1 = P_2 = 0, K = \pm i$, which makes no physical sense. Thus there is no such intersection point exists. It can also be viewed graphically. For each bound, a sweep is performed and the results are shown in Figure 5.7 (a). It can be seen that when the value of K decreases towards the LHS bound, system goes unstable quickly and there is no intersection point on LHS of $K_{lqr} = 1.414$. On the RHS of K_{lqr}, the two performance curves increase with K at almost the same rate towards infinity, thus there will be no intersection point either towards positive infinity.

Actually, an extension from above example is that for any $1D1C$ case with variant A, there will be no intersection point and for any value of the controller since R is always non-negative defined.



(a) $1D1C$ example (b) $2D1C$ example

Figure 5.7 No intersection point of two performance curves at two bounds

Example 5.4: The $2D1C$ example 1 is examined: now there are two elements in the controller to be taken care of. Set K_2 constant at $K_2 = -0.1$. K_1 sweeps from 2 to positive infinity. It can be seen from Figure 5.7 (b) that as K_1 approaches 2, J increases sharply and the system becomes unstable. On the other direction, the value of J at the bound $A_{12} = 0.9$ increases much quicker than that of bound $A_{12} = 1.1$. Visually inspect, it is intuitive to draw the conclusion that there is no intersection point from the two curves either. Mathematically, in this example above system of equations ends up with $K_1 = 0.108 \pm 0.036i$, a complex conjugate pair of solutions which makes no physical sense and implies that there is no intersection point.

When there is no intersection point, one implication is that the performance at one corner is always better than the other. So if it is the case in the minimax controller search process, the corner with better performance can be ignored when searching for the worst-case point, though there is still need to examine both corners to confirm which corner is the better one to be ignored.

Case2: $b^2 - 4ac = 0$. In such circumstance, there is only one solution exists and it suggests that there is only one intersection point for the two curves. In fact, such case seldom occurs. It requires a strong premise that the curvatures of two are exactly the same.

Case3**:** $b^2 - 4ac > 0$. Such circumstance doesn't exist due to its special structure. The main reason is that, there is only one unknown with square term. For other quadratic terms, they are all interactions between different unknowns.

Example 5.5: Similar to Example 5.1, the $2D1C$ example 1 and the corresponding system of equations from Kronecker operation is used. Also it is assumed that the unknown comes from one specified element K_1 from the controller, K_2 is held constant at $K_2 = -0.1$.

$$(2 - 2K_1)P_{1_{11}} - 2P_{1_{12}}K_1 = -K_1^2 - 1 \qquad (5.14)$$

$$P_{1_{11}} + (2.1 - K_1)P_{1_{12}} - K_1 P_{1_{22}} = -0.1K_1 \quad (5.15)$$

$$2P_{1_{12}} + 2.2P_{1_{22}} = -1.01 \quad (5.16)$$

$$(2 - 2K_1)P_{2_{11}} - 2P_{2_{12}}K_1 = -K_1^2 - 1 \quad (5.17)$$

$$1.2P_{2_{11}} + (2.1 - K_1)P_{2_{12}} - K_1 P_{2_{22}} = -0.1K_1 \quad (5.18)$$

$$2.4P_{2_{12}} + 2.2P_{2_{22}} = -1.01 \quad (5.19)$$

$$P_{1_{11}} = P_{2_{11}} \quad (5.20)$$

As mentioned above, the term K_1^2 only exists in the first and forth equation. All other quadratic terms are the product of K_1 and elements in P_1 and P_2. In such case, after all unknowns in P are substituted with expressions of K_1, it remains a third degree polynomial with only K_1 to solve. Theoretically saying there is a real solution, along with a pair of conjugate solutions. If there are two intersection points, or two real solutions from above system of equations, then one of the two solutions must come from the pair of conjugate solutions and it is real. That suggests that the other one in the conjugate pair is also real and thus there are three real solutions in total. However, when the point is fixed in the uncertainty space and there is a controller given, there is uniquely calculated performance, thus the performance curve is open WRT controller. From Theorem 1, there are at most two intersection points. Then there is a contradiction and implies that initial assumption that there are two real solutions from above system of equations is wrong.

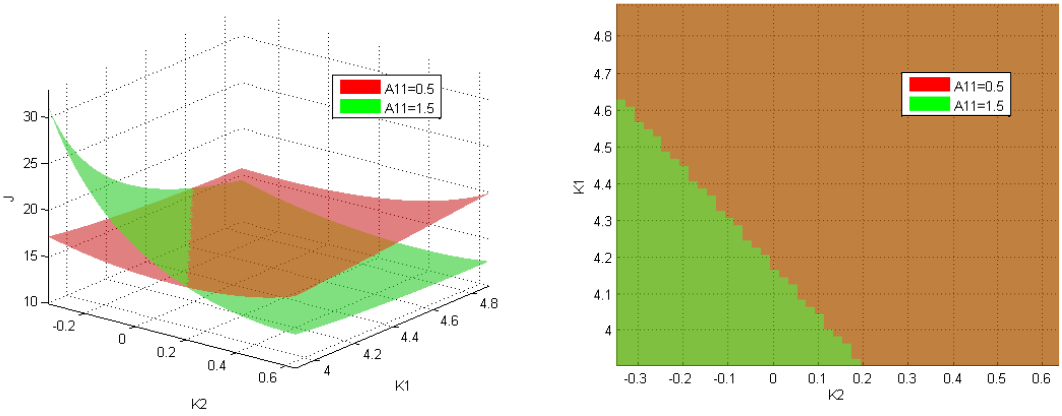
5.4.3 Extensions to high dimension cases

All above analysis is based on the assumption that there is only one unknown in the controller, which is seldom the case. When the given values in the controller, e.g., $K_2 = -0.1$ in $2D1C$ example changes, for sure above condition will be changed. Next research question will expand the controller design space to high dimensions and make preparations for NEMCDM to be introduced in next section.

RQ 1.5: What if there are multiple unknowns in the controller in Equation (5.1)-(5.2)?

Hypothesis 1.5: There will be infinite solutions and the intersection topology will change.

Proof of concept: When there are multiple unknowns in K, then there will be extra unknowns in the system of equations. Then numerically the system of equations can only be reduced to a single polynomial equation with multiple (more than one) unknowns. In such circumstance, for sure there will be infinite solutions. Graphically, other than an intersection point, there will be a different topology, i.e., an intersection curve, depends on the numbers of unknowns.



(a) 3D view (b) Bird view

Figure 5.8 Intersection curve of two performance surfaces at two bounds

Example 5.6: Again, the $2D1C$ example 1 is used as an illustrative example: instead of intersection point, now there is an intersection curve of the two response surfaces of two bounds where $A_{11} = 0.5$ and $A_{11} = 1.5$, whose $2D$ projection into controller design space is just the single polynomial equation simplified from Equation (5.8)-Equation (5.10) with two variables (K_1 and K_2) to solve. Note that the zigzag shape of the intersection curve is due to the granularity used in the sweep method to generate Figure 5.8 (a): the response surface is not continuously but discretely sampled; though in fact it should be smooth and the intersection curve should be also smooth everywhere.

5.5 Local Minimax Controller Design

In this section, the intersection of higher manifolds will be first examined and profiled, followed by a ***Triangle Based Gradient Method*** to find a local minimax controller for a pre-selected pair of corners, on the premise that the two performance topologies at two corners intersect.

5.5.1 Existence of worst-case point

The challenge brought from the higher manifolds intersection is that the controller design points who are located on the intersection curve are no longer unique, i.e., any controller design point that rides on the intersection curve in Figure 5.8 makes the performances at two corners equal to each other and thus satisfies Equation (5.1)-(5.2). Next, stick to the $2D1C$ example 1, a closer look is taken to yield properties of such intersection curve.

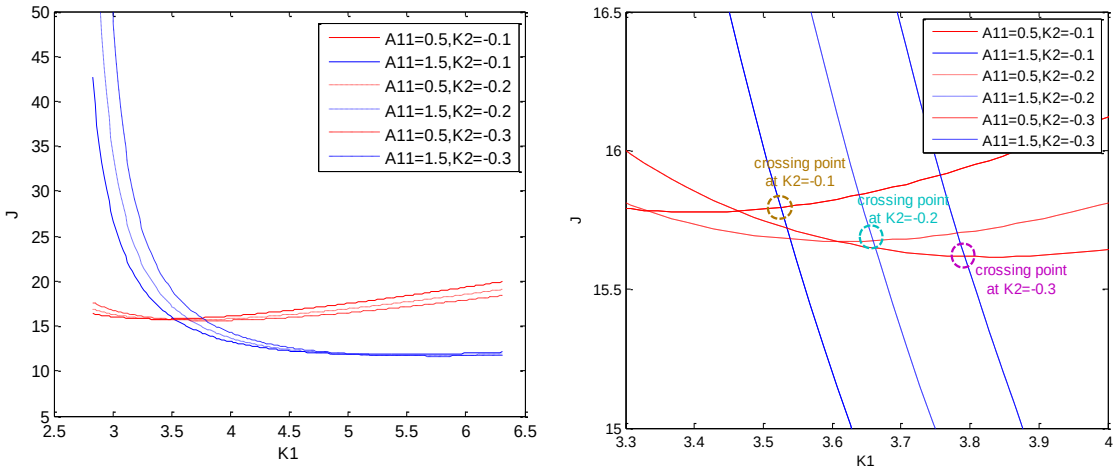


Figure 5.9 Intersection point shifts with $\boldsymbol{K_2}$

Figure 5.9 (a) extends from Figure 5.5 in the sense that K_2 shifts from the default value $K_2 = -0.1$, e.g., now $K_2 = -0.2$. Now the performance curves of two bounds in terms of K_1 changes and there will be a new intersection point so that following equation still holds true,

$$J(A + \Delta A_1, K_1, K_2 = -0.2) = J(A + \Delta A_2, K_1, K_2 = -0.2) \qquad (5.\ 21)$$

Similarly, when K_2 further shifts to -0.3, there will be another new intersection point and following equation still holds true:

$$J(A + \Delta A_1, K_1, K_2 = -0.3) = J(A + \Delta A_2, K_1, K_2 = -0.3) \qquad (5.\ 22)$$

A finer view from Figure 5.9 (b) yields that when such deviation moves in a continuous fashion, the intersection points shape a curve and it is exactly the intersection curve of the two surfaces in Figure 5.8 and whose projection on the controller design space $[K_1, K_2]$ is mathematically described in RQ 1.5. Now let $S_{int} \subseteq S$ denotes the set within the controller design space, so that for all points $[K_1, K_2 \dots K_n]$ in S_{int}, the following equation is true:

$$[K_1, K_2 \dots K_n] \in S_{int} | J(A + \Delta A_1, K_1, K_2 \dots K_n) = J(A + \Delta A_2, K_1, K_2 \dots K_n) \in J_{int} \qquad (5.\ 23)$$

In above example where $n = 2$, the topology of S_{int} is a curve. When the dimension n further increases, e.g., $n = 3$, the topology becomes an surface. The corresponding set of such performance is denoted as J_{int}.

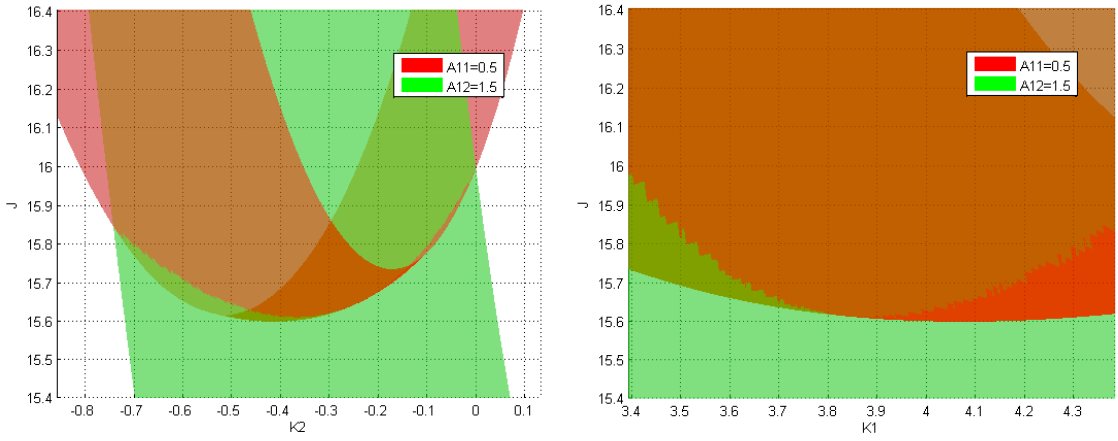


Figure 5.10 Side view of intersection curve of two response surfaces at two bounds

Though constrained by Equation (5.23), but there are still infinite combinations of $[K_1, K_2 \dots K_n]$ meets such constraint and each combination gives a unique J_{int}. There exists a minimum value of J_{int}, notionally $J_{minimax}$, with definition $J_{minimax} \equiv \inf(J_{int})$

and is shown in Figure 5.10 (a) and (b) who are the side views (K_2J and K_1J) of Figure 5.8. Though disturbed by zig-zag, there is still a clear evidence of the existence of $J_{minimax}$ on the intersection curve, which is the exact the design objective of NEMCDM. That brings the most challenging part, which is formally stated in the next research question.

5.5.2 Properties of worst-case point

RQ 1.6: How to calculate K_{minmax} and the corresponding J_{minmax}?

This research question is equivalent to the following question, which is described geometrically in terms of Figure 5.8: i.e., in $2D1C$ example 1, the two performance surfaces of two corners intersection each other and profile a curve, how to find the lowest point on that curve?

Hypothesis 1.6: A similar concept from Equation (2.14) can be borrowed so that when K_{minmax} is selected, $\frac{\partial J_{int}}{\partial K}|_{K_{minimax}} = 0$.

Hypothesis 1.6 can be directly proved as long as it can be proved that such local minimum is also a global minimum.

Proof of concept: Similar to the proof of Hypothesis 1.1, by assuming that there is another global minimum $J'_{minimax}$other than the local minimum J_{minmax}, it implies that the following equation holds true: $J'_{minimax} < J_{minmax}$. Then for the two adjacent points around $J_{minimax}$ and one other point between $J'_{minimax}$ and J_{minmax}, the three will give the same performances, which is against Theorem 1. Thus the initial assumption is not valid, which means that for a K that makes $\frac{\partial J_{int}}{\partial K} = 0$, it is a global minimax controller for this pair of corners.

The key concept can be graphically viewed from Figure 5.10 where J_{int} is mapped against all $K_1, K_2 \dots K_n$ where $[K_1, K_2 \dots K_n] \in S_{int}$. Suppose that there is a small

increment in all K_i, if the corresponding increment of J_{int} equals 0, then a local minimum is found. With above analysis, RQ 1.6 reduces to the following.

5.5.3 Calculations of worst-case point

RQ 1.7: How to calculate the gradient $\frac{\partial J_{int}}{\partial K_i}$ and $[K_1, K_2 \dots K_n] \in S_{int}$ so that $\frac{\partial J_{int}}{\partial K_i} = 0$?

Note that J_{int} is totally different from traditional quadratic performance J in Equation (2.31). So the gradient in Section 2.3.3 cannot be used. Next a new method is proposed to answer this research question.

Hypothesis 1.7: A ***Triangle based gradient method*** can be used to calculate $\frac{\partial J_{int}}{\partial K_i}$.

Before any further analysis, some notations are provided. Let $J^{A+\Delta A_1}|_{K_1,K_2}$ denotes the performance when the parameter uncertainty is known as $A_v = A + \Delta A_1$, the controller as $K = [K_1, K_2] \in S_{int}$. Essentially, it is the same with previously used expression $J(A + \Delta A_1, K_1, K_2)$, only with newly added superscript and subscript for linearization notations. Also let ΔK_n denotes a small increment in the value of K_n.

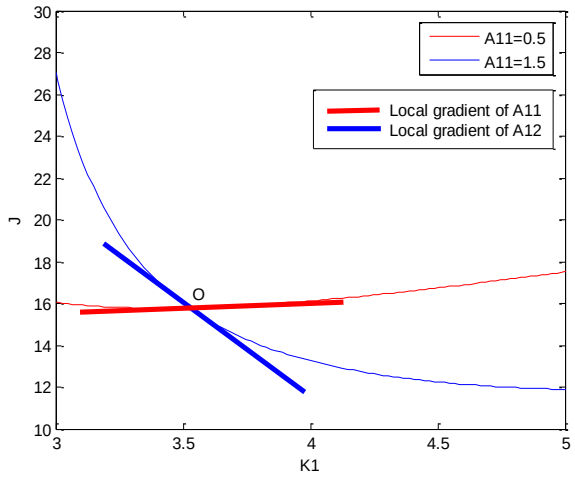


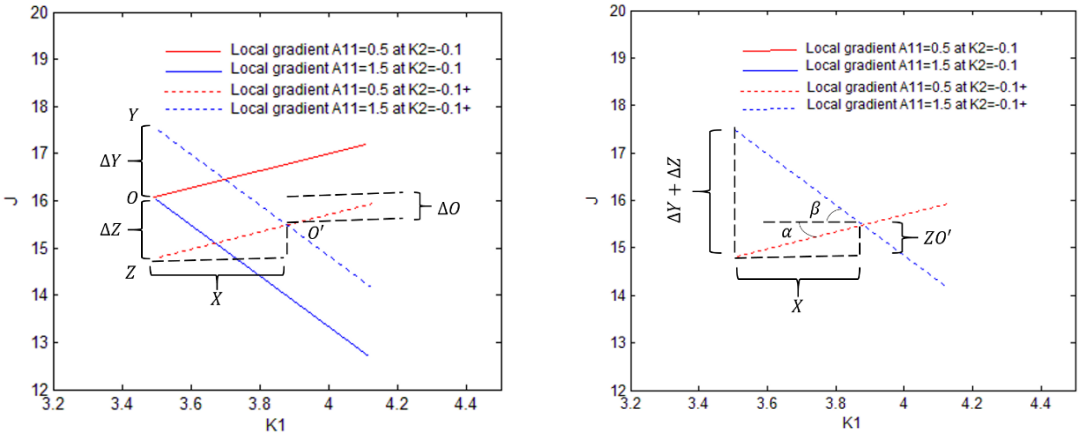
Figure 5.11 Local linearization of two performance curves of two bounds

Again, use the $2D1C$ example 1 for reference. Same with Figure 5.5, in Figure 5.11 there are two light solid curves in above plot, corresponding to the performances curves

at two bounds $J^{A+\Delta A_1}$ and $J^{A+\Delta A_2}$ as a function of K_1 exclusively ($K_2 = -0.1$, constant). Their intersection point is denoted as O. The bold solid lines denote the local gradients of the two curves after linearization and can be expressed as follows,

$$J^{A+\Delta A_1}|_{K_1+\Delta K_1} = J^{A+\Delta A_1}|_{K_1,K_2} + \frac{\partial J^{A+\Delta A_1}}{\partial K_1}|_{K_1,K_2}\Delta K_1 \quad (5.\ 24)$$

$$J^{A+\Delta A_2}|_{K_1+\Delta K_1} = J^{A+\Delta A_2}|_{K_1,K_2} + \frac{\partial J^{A+\Delta A_2}}{\partial K_1}|_{K_1,K_2}\Delta K_1 \quad (5.\ 25)$$



(a) (b)

Figure 5.12 Triangle based gradient method

$$Y = \frac{\partial J^{A+\Delta A_1}}{\partial K_1}|_{K_1}\Delta K_1 \quad (5.\ 26)$$

$$Z = \frac{\partial J^{A+\Delta A_2}}{\partial K_1}|_{K_1}\Delta K_1 \quad (5.\ 27)$$

When there is a small increment ΔK_2, there will be two brand new curves with local gradients, highlighted as dashed lines in Figure 5.12 (a) and the new intersection point moves to O'. In this context, let the two points Y and Z denote the value of J at $[K_1, K_2 + \Delta K_2]$. Their respective vertical distances (in J direction) from the initial intersection point O, or equivalently the distances between solid and dashed line are $\Delta Y = Y - O$ and $\Delta Z = O - Z$,

$$\Delta Y = Y - O = \frac{\partial J^{A+\Delta A_1}}{\partial K_2}|_{K_1,K_2}\Delta K_2 \quad (5.\ 28)$$

$$\Delta Z = O - Z = -\frac{\partial J^{A+\Delta A_2}}{\partial K_2}|_{K_1,K_2}\Delta K_2 \quad (5.\ 29)$$

Also, note that the vertical distance between O' and O, or $\Delta O = O' - O$ is exactly J_{int}'s change due to small increment in K_2. Thus the term $\frac{\partial J_{int}}{\partial K_i}$ can be written as follows,

$$\frac{\partial J_{int}}{\partial K_i} = \frac{O'-O}{\Delta K_2} \quad (5.\ 30)$$

From Figure 5.12 (b) it can be further seen that the two dashed lines and vertical line form a triangle. The slopes of the two lines are the performance J in terms of K_1 at the point O'. Though K_1 at new intersection point O' also change from O, due to small increment assumption, the slopes of new curves are assumed to be unchanged,

$$\tan(\alpha) = \frac{\partial J^{A+\Delta A_1}}{\partial K_1}|_{K_1,K_2} \cong \frac{\partial J^{A+\Delta A_1}}{\partial K_1}|_{K_1+\Delta K_1 K_2} \quad (5.\ 31)$$

$$\tan(\beta) = -\frac{\partial J^{A+\Delta A_2}}{\partial K_1}|_{K_1,K_2} \cong -\frac{\partial J^{A+\Delta A_2}}{\partial K_1}|_{K_1+\Delta K_1 K_2} \quad (5.\ 32)$$

Using ΔK_1 to denote the increment in K_1 due to change from K_2, the distance between Y and Z can also be calculated trigonometrically:

$$Y - Z = [\tan(\alpha) + \tan(\beta)]\Delta K_1 = (\frac{\partial J^{A+\Delta A_1}}{\partial K_1}|_{K_1,K_2} - \frac{\partial J^{A+\Delta A_2}}{\partial K_1}|_{K_1,K_2})\ \Delta K_1 \quad (5.\ 33)$$

Combine Equation (5.28), (5.29) and (5.33),

$$Y - Z = \Delta Y + \Delta Z = \left(\frac{\partial J^{A+\Delta A_1}}{\partial K_2}|_{K_1,K_2} - \frac{\partial J^{A+\Delta A_2}}{\partial K_2}|_{K_1,K_2}\right)\Delta K_2 = (\frac{\partial J^{A+\Delta A_1}}{\partial K_1}|_{K_1,K_2} - \frac{\partial J^{A+\Delta A_2}}{\partial K_1}|_{K_1,K_2})\ \Delta K_1 \quad (5.\ 34)$$

Now ΔK_1 can be written as function of ΔK_2,

$$\Delta K_1 = \frac{a-b}{c-d}\Delta K_2 \quad (5.\ 35)$$

where a, b, c, d have the following expressions,

$$a = \frac{\partial J^{A+\Delta A_1}}{\partial K_2}|_{K_1,K_2} \quad (5.\ 36)$$

$$b = \frac{\partial J^{A+\Delta A_2}}{\partial K_2}|_{K_1,K_2} \quad (5.\ 37)$$

$$c = \frac{\partial J^{A+\Delta A_1}}{\partial K_1}|_{K_1,K_2} \quad (5.\ 38)$$

$$d = \frac{\partial J^{A+\Delta A_2}}{\partial K_1}|_{K_1,K_2} \quad (5.\ 39)$$

In this manner, the final target can be written as

$$\Delta O = O' - O = \tan(\alpha)\,\Delta K_1 - \Delta Z \quad (5.\ 40)$$

Substitute Equation (5.35) into (5.40),

$$\frac{\partial J_{int}}{\partial K_i} = \frac{\Delta O}{\Delta K_2} = \frac{O'-O}{\Delta K_2} = \frac{\tan(\alpha)\Delta K_1 - \Delta Z}{\Delta K_2} = \frac{(c\frac{a+b}{c+d} - b)\Delta K_2}{\Delta K_2} = c\frac{a-b}{c-d} - b \quad (5.\ 41)$$

Above equation can be used as the stopping criteria for the local minimax controller line search. When $\frac{\partial J_{int}}{\partial K_2}$ approaches zero within a small enough distance, K_2 can be solved and K_1 can be calculated accordingly via Equation (5.8)-(5.9).

Note that similar to any other gradient search method, $\frac{\partial J_{int}}{\partial K_2}$ has the same mathematical meaning of that in Equation (2.31). Thus it is totally reasonable to use the absolute value of $\frac{\partial J_{int}}{\partial K_2}$ to speed up the search process: i.e., the search step length is set to be relatively large when the value of $|\frac{\partial J_{int}}{\partial K_2}|$ is large.

For a controller with more than two unknowns, above steps for each pair of K_i and K_1 can be repeated iteratively to get a $m * n - 1$ dimensional search direction with elements $[\frac{\partial J_{int}}{\partial K_2}, \frac{\partial J_{int}}{\partial K_3} \ldots \frac{\partial J_{int}}{\partial K_{m*n}}]$. For each search step, with all others updated from the line search, K_1 is the only unknown value to be solved from Equation (5.8)-(5.9) in the controller design space.

Now there is a way to calculate $\frac{\partial J_{int}}{\partial K_i}$ to profile the intersection manifold, the next research question concerns about the validity of the assumption that the minimax controller is always located on the lowest point on the intersection manifold.

RQ 1.8: Is it always the case that the minimax controller is located on lowest point on the intersection manifold?

Hypothesis 1.8: No, e.g., if the two response surfaces intersect each other with the same "trend", then even there exists a lowest point on the intersection curve, it is not the worst-case point. Actually, only when the two manifolds have two local gradients of different directions (a positive and a negative) at the intersection point, then it is a minimax controller design point. A notional counterexample is provided in Figure 5. 13. The existence of such "counter example" will be further examined in future research.

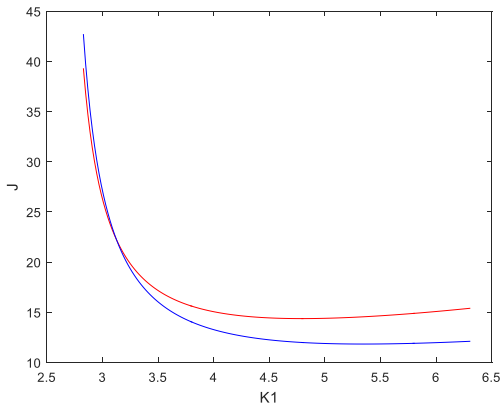


Figure 5. 13 A counter example of the worst-case point is always located on intersection manifold

Under such circumstance, the minimax controller actually goes to the default LQR of one corner/bound who yields a worse performance. A quick validity check can be made: first calculate K_{lqr} and corresponding performances at two corners. If for the two calculated K_{lqr}, performances of one corner/bound are always worse than the other, then the minimax controller of this pair of corners/bounds equals to the default LQR at this corner/bound.

5.5.4 Algorithm to local minimax controller

Given any pair of corners/bounds, combining above analysis, the line search algorithm to find a local minimax controller is summarized below. The merit of the algorithm is to numerically profile the intersection curve and keep tracking the value of $\frac{\partial J_{int}}{\partial K_i}$. When the

value approaches 0, then a local/global minimum design point on the intersection curve of the selected pair of corners is reached.

Algorithm 1	**Line search to a local minimax controller**
Step 1	Calculate K_{lqr} and corresponding performances at both corners
Step 2	With two K_{lqr}, if performances of one corner are always better than the other corner's, then this corner can be ignored. Algorithm stops as the worst-case point is located at the corner with worse performance exclusively.
Step 3	Otherwise, treat one element from the controller, e.g., K_1 as unknown and all others $[K_2 \dots K_{m*n}]$as known, use the method in Section 5.2 to calculate K_1 so that Equation (5.8)-(5.10) hold true.
Step 4	Calculate $[\frac{\partial J_{int}}{\partial K_2}, \frac{\partial J_{int}}{\partial K_3} \dots \frac{\partial J_{int}}{\partial K_{m*n}}]$ by using Triangle based gradient method in Section 5.4 and use this as a search direction to update $[K_2 \dots K_{m*n}]$.
Step 5	Repeat step 3 and step 4 until the norm of gradient $[\frac{\partial J_{int}}{\partial K_2}, \frac{\partial J_{int}}{\partial K_3} \dots \frac{\partial J_{int}}{\partial K_{m*n}}]$ is within certain small value.

Example 5.7: Utilize above line search method on the $2D1C$ example 1 and the result is shown in Table 3 and Figure 5. 14. At the first step, the initial K_2 is selected as $K_2 = -0.1$. K_1 and $\frac{\partial J_{int}}{\partial K_2}$ are then solved as $K_1 = 3.525$, $\frac{\partial J_{int}}{\partial K_2} = 1.56$ by using the Triangle based gradient method. Since the optimization direction is to minimize, the search direction is the opposite of local gradient: $\Delta K_2 = -\frac{\partial J_{int}}{\partial K_2} = -1.56$. A scale factor of 0.2 is selected to regulate the step length. Table 3 lists the value of K_1, K_2 and $\frac{\partial J_{int}}{\partial K_2}$ at each line search step.

Table 3 Line search steps for Algorithm 1

Step No	$\boldsymbol{K_1}$	$\boldsymbol{K_2}$	$\frac{\partial J_{int}}{\partial K_2}$	**Step No**	$\boldsymbol{K_1}$	$\boldsymbol{K_2}$	$\frac{\partial J_{int}}{\partial K_2}$
1	3.525	-0.1	1.5625	**6**	3.883	-0.3680	-1.75e-4

	0				1		
2	3.9423	-0.4125	-0.1775	**7**	3.8831	-0.3680	-2.9e-5
3	3.8950	-0.3770	-0.0373	**8**	3.8831	-0.3680	-5e-6
4	3.8851	-0.3695	-6.464e-3	**9**	3.8831	-0.3680	-1e-6
5	3.8834	-0.3682	-1.068e-3	**10**	3.8831	-0.3680	-1e-7

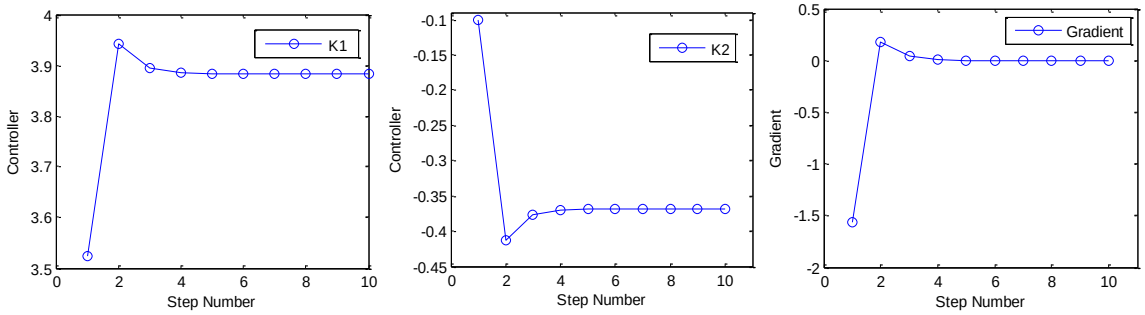


Figure 5. 14 $\boldsymbol{2D}$ controller line search

It can be seen that at step 2, there is a slight overshoot. This is mainly due to the search step length is relatively large, though it can be easily proved that the line search algorithm is globally stable. The line search is then pushed back and after 6 steps and the line search converges at $[K_1 = 3.8831, K_2 = -0.3680]$, gradient $\frac{\partial J_{int}}{\partial K_2}$ gradually approaches 0.

The side view of the intersection curve from Figure 5.10 can be used to visually validate the results from above method. It can be seen that the calculated minimax controller $[K_1 = 3.8831, K_2 = -0.3680]$ is the same with the visually observed one who gives $J_{minimax}$ from Figure 5.10. The minor error could be contributed from granularity in Figure 5.10 and the linearization in the Triangle based gradient method.

5.6 Global Minimax Controller Design

So far, for any pair of corners/bounds, there is a method provided in Section 5.5 to profile the intersection curve and thus find a controller that yields a global best worst-case performance. The calculated controller can be treated as the global minimax controller, but only for this pair of corners/bounds. Thus, it is still a "local" minimax controller when viewed from the whole uncertainty space point of view. The next research question and hypotheses try to extend Algorithm 1 to cover all corners/bounds in the uncertainty space.

RQ 1.9: How to expand above design to find the global minimax controller?

Hypothesis 1.9: Due to the discontinuity between corners/bounds, each pair of them needs to be checked. There is even a chance that one corner has no intersection point with any other corners and yields the worst-case performance exclusively. In such case, the traditional LQR method can be applied at this corner to find the minimax controller.

Next, a comprehensive procedure of finding such global minimax controller over the whole uncertainty space is summarized below.

Algorithm 2	**Method towards a global minimax controller**
Step 1	Calculate K_{lqr} for all corners and corresponding performances at all corners.
Step 2	If with all K_{lqr}, performances of one corner are always worse than the other, then the worst-case point is located at the corner exclusively. Minimax controller is exactly K_{lqr} at this corner. Algorithm stops.
Step 3	Select a pair of corners, calculate whether there is intersection point, or if there is real solution from Equation (5.8)-(5.10). If yes, name one corner as an initial corner and the other corner as the most recent corner. Calculate a local minimax controller by using Triangle based gradient method from Section 5.4.
Step 4	If there is no intersection point, then pick any controller, e.g., the initial controller and check which corner is worse and discard the one with better performance. Apply traditional LQR method on the

	selected corner, treat it as the local minimax controller for this pair of corners.
Step 5	With calculated local minimax controller, calculate performances at other corners. If no larger ones, then it is global minimax controller. Otherwise, reselect new pairs of corners, e.g., a pair composed of the worst performance violating corners.
Step 6	Repeat step3 to step5 until there are no corners yield worse performance.

A comprehensive example will be provided in Chapter 7.

5.7 Complexity analysis

Note that though the system of equations (5.8) to (5.10) is analytically solvable, it is also important to check the computational cost. To solve the traditional Lyapunov equation, numerical method usually gives a better computational efficiency. Haddard also proposed a two-stage iterative method to solve pairs of Riccati equations [30]. On the other side, analytical solution requires a large number of eliminations and becomes inefficient when the matrix dimension increases. Thus, not necessarily that the analytically method is favored. Unfortunately, numerical method to solve such pair of Lyapunov equation is not the research interest of this thesis.

For NEMCDM, a lot of computational time will be spent on solving (coupled) Lyapunov equations. How to solve is not limited to Kronecker operations method only, though a lot of proofs used in this research is performed in this manner. In fact, if standard linear Lyapunov equation is solved using Kronecker operations as a system of $n*(n+1)/2$ equations and $n*(n+1)/2$ variables, the cost is $O(n^6)$. Other methods, such as Bartels and Stewart's algorithm, Schur or upper Hessenberg form can also be used and they are more efficient with $O(n^3)$ [31]. However, as NEMCDM requires the inverse of a matrix and when there is an unknown variable in the matrix to be inversed, aforementioned algorithms are no longer applicable.

In terms of the line search, a relatively large search step can be used to speed up the process, since even an overshoot can be corrected due to the continuous and convex nature of the performance response manifold WRT controller. A linearization method, to be introduced in Chapter 6 can also be used to speed up the search process though it is not necessary at all.

It is interesting to compare the two methods (TMCDM and NEMCDM) in terms of computational time. Note that it makes no sense to compare the calculated minimax controller or performance since the two uncertainty structures/spaces are totally different from each other.

For the traditional method, though it "combines" the parameter uncertainty space and the controller design space, it still needs to sweep the parameter uncertainty space numerically. When the dimension increases, the workload of sweeping exponentially increases since the parameter uncertainty space is continuous. Though the parameter uncertainty space of extended method is even larger than that of the traditional method, thanks to Hypothesis 1.1, the potential locations of worst-case points are actually hugly reduced. Thus given that the two methods take the same complexity for one-time calculation of coupled Lyapunov equation, NEMCDM has less overall complexity.

On the other side, imagine an extreme case of the extended method that there are infinite uncertainty channels, then NEMCDM converges to TMCDM.

5.8 Extensions

This methodology is also applicable to the cases where there is discrete parameter uncertainty space, e.g., in the real plant, some physical entities have discrete settings, leaving the matrix A with discrete values. These values in different uncertainty channels could be even correlated and dooms the definition of "bound" used in NEMCDM, i.e., the corner of two discrete uncertainty channels happens to be unreachable due to strong correlation, then how do the designers move the new corner to the next available point?

In such circumstance, one extra step is needed to formulate the problem into a structure where NEMCDM can be applied by screening out bound/corner points. A ***Pareto Frontier*** method [32] can be used. As shown in Figure 5.15, this method particularly profiles a state of allocation of resources in which it is impossible to make any one individual better off without making at least one individual worse off. Due to the easy concept and page length, no further analysis will be given here.

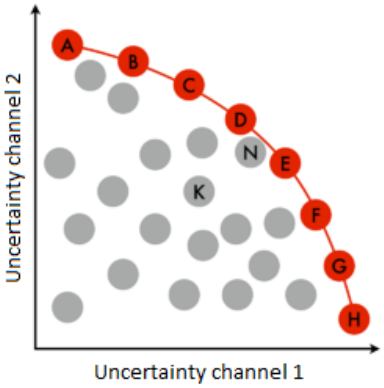


Figure 5.15 Pareto frontier of corners selection [33]

CHAPTER 6

POF CONSTRAINED OPTIMAL AVERAGE PERFORMANCE CONTROLLER (PCOAPC)

In Chapter 5, the first level of conservatism is reduced by relaxing the norm constraint. In this chapter, the second stage will be introduced to further reduce the conservatism by incorporating uncertain parameters' probability information into consideration. The proposed controller design method is labeled as POF Constrained Optimal Average Performance Controller (PCOAPC) method.

6.1 Design objective and problem formulation

With Observation 2 in mind, this chapter starts from the following overall research question.

***RQ 2*:** How to incorporate probability information into performance consideration?

Hypothesis 2: Use average performance instead of the normal performance; introduce the concept of POF as the extra evaluation criteria. The introduction of probability information into the design relaxes the tight constraint and allows a tradeoff between the worst-case (POF) and other performances such as the average performance.

Their definitions, mathematical expressions, properties, and the procedures of finding a desired controller will be given in the next few sections.

6.1.1 Average performance over parameter uncertainty

The formal definition and mathematical expression of the average performance are given below,

$$J_{ave} = E(J) = \int_{a_1^l}^{a_1^u} \dots \int_{a_n^l}^{a_n^u} Pr(\Delta a_1 \dots \Delta a_n) J d\Delta a_1 \dots d\Delta a_n \qquad (6.\ 1)$$

where $Pr(\Delta a_1 \dots \Delta a_n)$ denotes the probability of a particular point $\Delta a_1 \dots \Delta a_n$ in the uncertainty space U and is assumed to be given. Such information could be retrieved

from physical observations, sampled results etc. How the probability information is derived is not of interest in this research.

Above definition is not to be confused with H_∞, who is the expected value of quadratic performance over random noise or random initial value, as referred in many LQR or LQG design methods, where a strong assumption of a normal distribution with a mean equals to 0 is needed. The expected value in Equation (6.1) is the integration over parameter uncertainty space U, whose bound should be specified. The probability distribution could be any and there is no associated requirement.

Next, a new controller K_{ave} is given, with a aim to optimize the average performance J_{ave}, ,

$$E\big(J(A_i, K_{ave})\big) \equiv \min(J_{ave})$$

Note that K_{ave} always stabilizes the system no matter what value the uncertain parameter has. Should there is a value $\Delta a_i \in \left[\Delta a_i^l, \Delta a_i^u\right]$ that makes the system unstable with K_{ave}, $J(A_i, K_{ave})$ approaches $+\infty$ and drives J_{ave} towards $+\infty$ as well, as long as the associated probability at Δa_i is not 0, which is against the definition of K_{ave}.

Example 6.1: The $1D1C$ example problem is used as an example for the existence of such controller. To make it easier, uniform distribution is assumed within the uncertainty range $\Delta a_1 \in [-0.5, 0.5]$. A mathematical expression of average performance is given below so that further analysis such as derivatives could be performed analytically.

With uniform distribution assumed, $\Pr(\Delta a) = \frac{1}{\Delta a^u - \Delta a^l} = 1$. Thus Equation (6.1) is now as follows:

$$J_{ave} = \int_{0.5}^{1.5} \frac{1+K^2}{A+K}\, dA = -(K^2+1)[\log(K+0.5) - \log(K+1.5)] \quad (6.2)$$

With the integration gone, it is easy to take first order derivative of above equation WRT K and solve for K_{ave} by setting the derivative equals to 0, so that at least a local minimum can be found. Since it's a $1D$ problem, an $1D$ non-linear scalar equation is finally reached.

$$\frac{\partial J_{ave}}{\partial K} = -(K^2+1)\left(\frac{1}{K+0.5} - \frac{1}{K+1.5}\right) - 2K[\log(K+0.5) - \log(K+1.5)] = 0 \quad (6.3)$$

Though there is no analytical solution, above equation can still be solved numerically, e.g., Newton's method [34]. It turns out that with $K = -2.61$ uniquely, Equation (6.3) holds true. Second order derivative also yields a positive value which suggest a convexity and a global minimum. Also note that $K_{ave} \neq K_{lqr}$, which is calculated as $K_{lqr} = -2.414$. This simple example shows the fact: when the probability distribution is given, there exists a controller K_{ave} who yields an optimized average performance and is not necessarily the same with K_{lqr}.

However, the bad news is that when the probability distribution gets more complicated than a uniform distribution, there seldom exists an analytical solution even the distribution can be expressed analytically. Not to mention if the probability distributions are discrete. Thus, numerical methods is left alone to find K_{ave}. Also note that the bottom line is, by using sweep method one can exhaust the whole design space and use sampling method to count for the randomness of probability distribution and thus always find a desired controller. But as mentioned in the beginning of this thesis, computational time is also an important consideration. With this being said, there will be analysis in next two sections focusing on the inefficiency of sampling method and instead how to numerically calculate the average performance and thus find a desired controller.

6.1.2 POF calculation via probability information

It is also clear that though K_{ave} takes advantage of probability information, the robustness is sacrificed as it takes no consideration of the worst-case point. A good way to retain the robustness is to fulfill the robustness requirement in a relaxed manner of constraining the POF smaller than a given number, as described in Chapter 4 Gap analysis 2 and is shown in Equation (6.4):

$$POF|_{J>Jc} < e \qquad (6.4)$$

It is also important to select the proper performance criteria J_c. A value of J_c that is too large would bring conservatism while a small one would lead to infeasibility. The minimax performance derived from the last chapter is the perfect one by guaranteeing feasibility and zero possible conservatism. Unless specified otherwise, it will be used as the default evaluation criteria in this and following chapters.

6.1.3 Problem definition

With above definitions, the problem to be solved in this chapter, labeled as ***POF* Constrained Optimal Average Performance Controller (PCOAPC)** is formally formulated as below,

To find a controller K_{pco} that optimizes the average performance $J_{ave}|_K$ while meets the POF constraint:

$$\begin{aligned} min \qquad & J_{ave}|_K \\ & POF|_{J>Jc} < e \end{aligned}$$

Also, note that though the average performance is now the prime optimization objective, it doesn't suggest that the robustness is out of the consideration. Actually the linkage between the average performance and the robustness is: if stability alone is used as the evaluation criteria and the definition of robustness, very likely there will be infinite controllers fulfill such requirement; only when the minimax robustness is used as design objective, usually there exists a unique minimax controller; when the performance evaluation criteria reduces to above POF constraint solely, again there tends to be infinite controllers meet such criteria; only with optimizing the average performance as an extra optimization objective, there usually exists a unique controller that gives optimized average performance while meets POF constraint and this is the one to be found in this chapter. Here the $2D1C$ example is used to illustrate above concept.

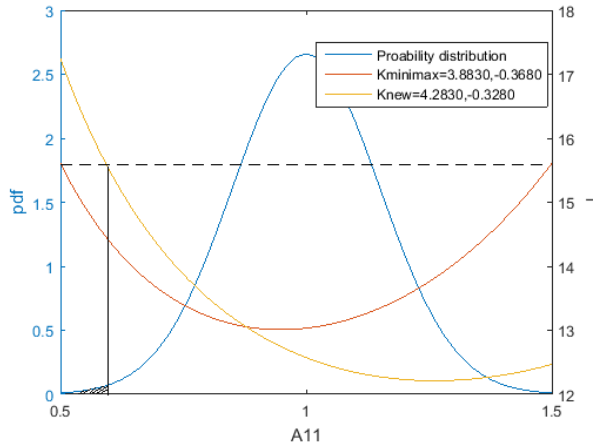


Figure 6. 1 ***POF*** notation

Example 6.2: From Section 5.4, it is calculated that the minimax controller is $K_{minimax} = [3.8830, -0.3680]$ and $J_{minimax} = 15.608$. The performance over parameter uncertainty space curve is plotted in red in Figure 6. 1. It can be visually validated that the two calculated performances at two bounds equal each other at 15.608. When $J_{minimax}$ is used as the evaluation criteria, $POF = 0$ with $K_{minimax}$ since it is a tight constraint.

When K_{new} deviates from $K_{minimax}$, e.g., $K_{new} = K_{minimax} + [0.4,0.4] = [4.2880, -0.3280]$, a new performance curve can be generated and is highlighted in yellow. Now the POF will be greater than 0 with contributions from the shadowed region near A_{11}. Such POF can also be calculated as long as the probability distribution is given, e.g., a truncated normal distribution highlighted in blue whose vertical axis is PDF. Then POF is equivalent with the PDF integrated area, or the CDF of the shadowed region. Finding a controller that makes POF meet the constraint is equivalent with finding a controller that makes the PDF integrated area of the shadowed region exactly equals to or smaller than the given constant.

6.1.4 Solution procedures

Since analytical solution barely exists, there is a need for numerical solutions. Broadly speaking, the problem depicted in above section is a traditional non-linear constraint optimization problem. Many methods are provided to solve this type of problem, such as Lagrange multiplier, penalty method [35], if both target equation and constraint can be analytically expressed. When there are no such analytical expressions available, numerical methods such as line search are preferred: a free line search can be first performed towards better performance until the constraint is hit; then search along the constraint until the feasible and useful directions go against each other.

Figure 6. 2 shows how to translate above general solution procedures into the context of this problem: keep searching towards a direction that improves the average performance (black line search trace), stop when the POF constraint is reached. Then keep searching along the direction that makes POF ride exactly on the constraint while improves the average performance simultaneously (purple search trace). Stop when the product of two search directions is negative.

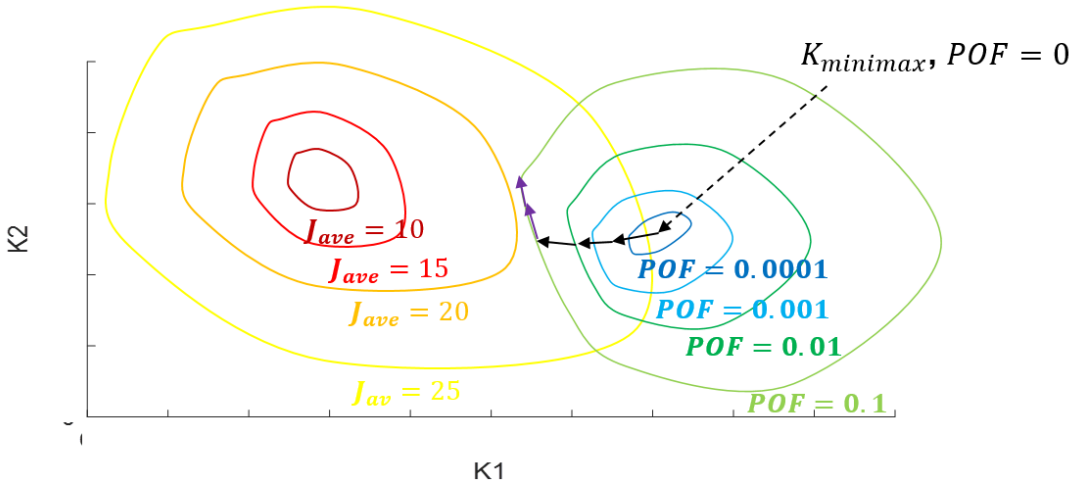


Figure 6. 2 General solution procedure for PCOAPCD line search

With above formulated problem and preliminary solution procedure analysis, there are five questions to be answered:

1. How to calculate average performance?
2. How to find a search direction towards better performance?
3. How to calculate POF?
4. How to find the search direction along the POF constraint?
5. When to stop the line search?

These five search questions will be sequentially answered in the following sections. The last section of this chapter will address other considerations such as the global optimality and the steepest search direction.

6.2 Numerical method for average performance calculations

This section concerns about the average performance related calculations.

RQ 2.1: How to numerically calculate the average performance?

Hypothesis 2.1: A discretized-summation method can be used by incorporating each segment's performance and discretized probability.

Proof of concept: The parameter uncertainty space can always be discretized, or averagely divided into d small segments. Each segment's discretized probability, or equivalently ***probability of occurrence*** (***POO***) can also be calculated from the given probability distribution. The following equation always holds true according to the definition of POO.

$$\sum_{i=1}^{d} POO(\Delta A_1, \dots \Delta A_c) = 1 \tag{6. 5}$$

The average performance can be roughly expressed as the sum of the product of each section's performance and POO,

$$J_{ave} = \sum_{i=1}^{d} J(K, \Delta A_1, \dots \Delta A_c)\, POO(\Delta A_1, \dots \Delta A_c) \tag{6. 6}$$

where POO is calculated as below,

$$POO(\Delta A_1, \dots \Delta A_c) = \frac{Pr(\Delta A_1, \dots \Delta A_c)}{d} \tag{6. 7}$$

It is obvious that the more segments the parameter uncertainty space are divided into, the more accurate the calculated average performance is. How many segments there should have changes from case to case and is not of the interest of this research.

RQ 2.2: How to find a search direction towards a better performance?

Hypothesis 2.2: Discretize the whole uncertainty space and sum the product of each segment's local gradient and POO.

Proof of concept: Given above discretized-summation method and borrow the concept of the steepest descent direction, a similar derivative can be performed about the local gradient of average performance WRT controller. Since POO is independent of controller, there is only a need to take derivative of the performance J in terms of controller.

$$\frac{\partial J_{ave}}{\partial K} = \sum_{i=1}^{d} \frac{\partial J(K, \Delta A_1, \dots \Delta A_c)}{\partial K} POO(\Delta A_1, \dots \Delta A_c) \quad (6.\ 8)$$

Thus, a line search method with a search direction calculated as below can be used,

$$search\ direction = -\frac{\partial J_{ave}}{\partial K} \quad (6.\ 9)$$

The mathematical meaning of above discretized-summation gradient equation is straightforward: when the corresponding POO of one discretized segment is large, a search direction that makes the performance of this segment better is more desirable compared to other search directions, even though this search direction might not benefit, even penalize other discretized segments. Thus the overall search direction that improves the average performance most should lean towards the larger contributors, or weight from the discretized segment that has the larger POO. It's also helpful to two extremes as reference: if the uncertain parameter has uniform distribution, then there shouldn't be any weight on each section's steepest descend direction as each section's importance is equivalent with other. Alternatively, if one segment dominates POO and leave others trivial, then the controller search direction should be dominated by the steepest descend direction from that segment. Thus the search direction at each iteration can be expressed as the sum of the product of each segment's POO and local gradient.

Next research question brings out common concerns which apply to any traditional line search method.

RQ 2.2.1: How to ensure the line search won't end up to a local minimum instead of a global minimum?

Hypothesis 2.2.1: No matter what the probability distribution is, the average performance manifold WRT to the controller is always convex. Thus, a local optimal average performance controller is also a global optimal average performance controller.

Proof of concept: Take a second order derivative of Equation (6.6),

$$\frac{\partial^2 J_{ave}}{\partial K^2} = \sum_{i=1}^{d} \frac{\partial^2 (K,\Delta A_1,\dots \Delta A_c)}{\partial K^2} POO(\Delta A_1, \dots \Delta A_c) \qquad (6.\ 10)$$

From above equation, according to the properties of quadratic performance J, $\frac{\partial^2 (K,\Delta A_1,\dots \Delta A_c)}{\partial K^2} > 0$ for any value of A and K, the other term $POO(\Delta A_1, \dots \Delta A_c) \geq 0$ as well. Thus the sum of their product $\frac{\partial^2 J_{ave}}{\partial K^2} > 0$ anywhere. Thus the average performance in terms of controller K is strictly convex globally. Thus the line search will always lead to a global minimum.

Example 6.3: The $2D1C$ example is used to illustrate above concept. The uncertain range $A_{11} \in [0.5,1.5]$ is averagely divided into 21 segments (shown in Table 4) and the following discrete probability distribution is assumed.

Table 4 Uncertainty discretization

A_{11}	**0.5**	**0.55**	**0.6**	**0.65**	**0.7**	**0.75**	**0.8**	**0.85**	**0.9**	**0.95**	**1**
POO	0.0676	0.0656	0.0636	0.0616	0.0596	0.0576	0.0556	0.0536	0.0516	0.0496	0.0476
A_{11}	**1.05**	**1.1**	**1.15**	**1.2**	**1.25**	**1.3**	**1.35**	**1.4**	**1.45**	**1.5**	**sum**
POO	0.0476	0.0496	0.0516	0.0536	0.0556	0.0576	0.0596	0.0616	0.0636	0.0656	1

With a scale fact of 0.2, the line search method uses less than 100 iterations to find the optimal average performance controller. When it stops at $[K_1 = 5.1651, K_2 = -0.4738]$, the gradient norm reduces to $1e-6$. Align the line search trace with the average

performance contours mapped by sweep method (no sampling necessary since POO is given) in Figure 6. 3, the line search method gives a quite satisfying result.

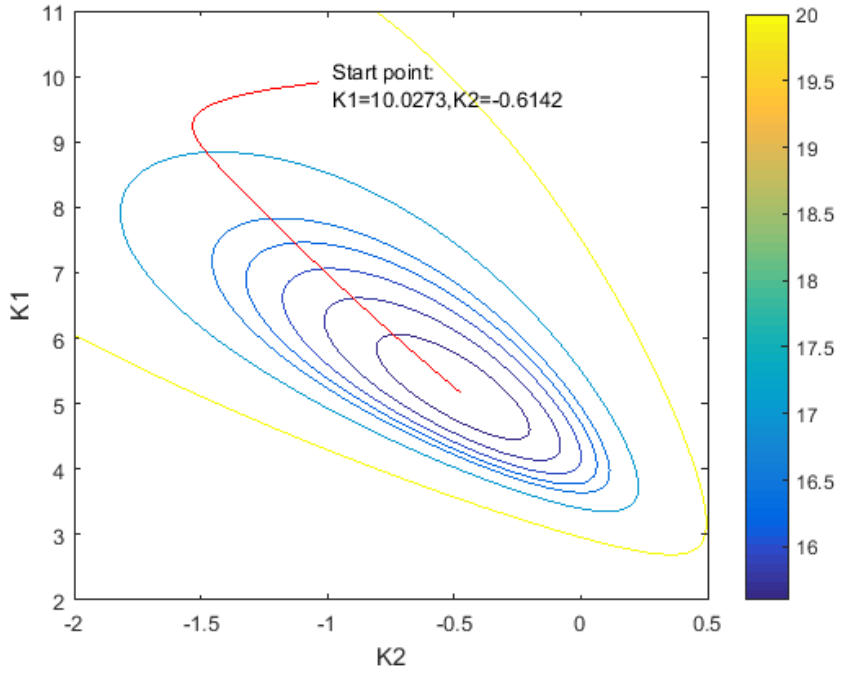


Figure 6. 3 Optimal average performance controller search

6.3 POF calculations and considerations

This section concerns about the POF related calculations.

RQ 2.3: How to calculate POF?

Hypothesis 2.3.1: A sampling method can be used to calculate POF. It is easy to program, but is extremely inefficient.

6.3.1 Inefficiency in sampling method

When there is no analytical method available, one commonly used method in this context is the sampling method, due to the fact that it is easy to program as long as probability distribution is known [36]. However, it becomes extremely inefficient in small POF considerations.

Proof of concept: When the POF is incurred from a small range within a large sampling space, which is very common, there is a need to sample intensively within this small range to achieve certain level of estimation accuracy. But it is impossible that any of the sampling method knows in advance where the small range is located.

Another way to understand the concept is via conceptual examples: it is assumed that $\pm 10\%$ error is allowed for the small POF estimation, e.g., for a target $POF = 0.01$, the allowable sampling error is $-0.001 < \Delta POF < 0.001$, which requires a relatively high fidelity of "absolute" POF estimation, which has to be compensated via a large sampling size.

Compared to the case where the same $\pm 10\%$ error is allowed for a large POF estimation, e.g., target $POF = 0.5$, then allowable sampling error is $-0.05 < \Delta POF < 0.05$, which is much larger than the ΔPOF in above case. The required fidelity of "absolute" POF estimation hugely reduces.

Example 6.4: An intuitive example of capturing a $POF = 0.01$ is provided. Using the sampling method to capture the POF around $K_{minimax}$ within a target range whose $POO = 0.01$, the ideal case is that there is exactly 1 out of 100, or 10 out of 1000 … total sampling points fall within this target range and thus incur performance failures.

However, when uniform sampling method is used, not necessarily there will be desired number of points fall within the target range, i.e., there will be a high chance that more than 1 or 0 out of 100 sampling points fall within the range and thus the calculated POF will be offset from the desired value. Such concern will be alleviated when the number of sampling points increases. The following table, whose calculation is based on Binomial distribution [37], shows the chance that there is $0.01 * (1 \pm$ sampling error) of total sampling points fall into target range, e.g., with 10000 sampling points of uniform distribution from 0-100, the chance that there are 95-105 points fall into range 0-1 is 0.4195.

It turns out that to reach a 90% confidence of accurate (sampling error< 0.05) POF calculation, at least 100000 sampling points are needed, which introduces huge inefficiency in the process. On the contrary, if there is a large POF to be captured, e.g., $POF = 0.5$, there only needs 1000 sampling points to reach a 90% confidence.

Table 5 List of sampling method accuracy requirement

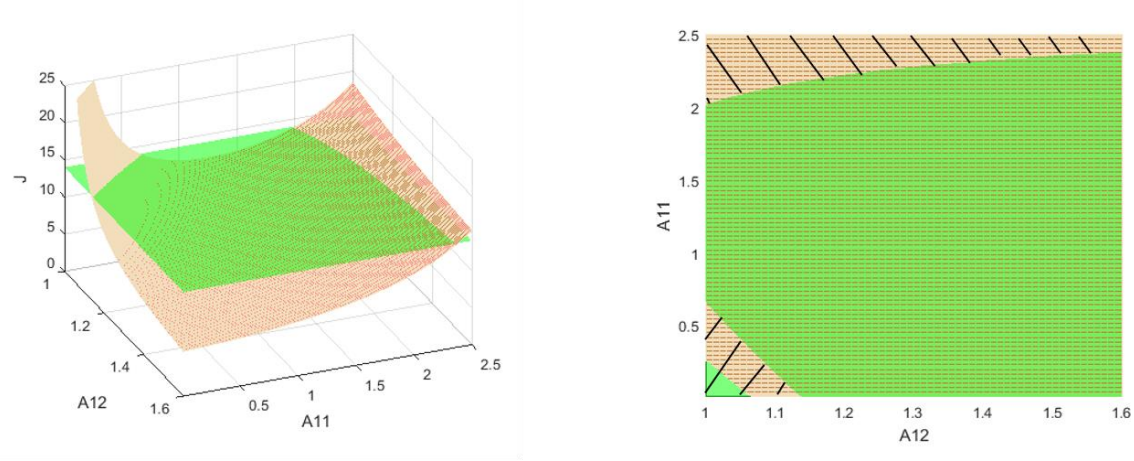
Sampling error=0.1	$\boldsymbol{POF = 0.01}$	$\boldsymbol{POF = 0.5}$	**Sampling error=0.05**	$\boldsymbol{POF = 0.01}$	$\boldsymbol{POF = 0.5}$
100	0.3697	0.7287	**100**	0.3697	0.3827
1000	0.3657	0.9986	**1000**	0.3657	0.8933
10000	0.7089	1	**10000**	0.4195	1
100000	0.9986	1	**100000**	0.8915	1
			1000000	1	1

6.3.2 Probability truncate method

Now the sampling method is ruled out, leaving only numerical methods to be explored. Next, a ***probability truncate method*** is proposed.

Hypothesis 2.3.2: A probability truncate method can be used to profile the intersection manifold and thus discretize the performance violation region and sum up each segment's POO to get the POF.

Proof of concept: The core idea of this probability truncate method is that due to the continuity of performance response in both uncertainty and controller design space, when a controller is selected, the geometry topology of the set of the parameter uncertainty that makes performance fails to meet requirement can be quantitatively profiled via Equation (5.1) and (5.2). Then the POO of this region can be calculated either through integration or discretized-summation method, as long as the probability distribution is given. While a $2D1C$ example is already shown in Example 6.2 and Figure 6. 1, another $2D2C$ example is shown below to further illustrate above concept.



(a) $3D$ view (b) Bird view

Figure 6. 4 Probability truncate method

Example 6.5: In the $2D2C$ example shown in Figure 6. 4, a controller is randomly selected and the corresponding performance response surface is plotted in red. The performance evaluation criteria is further selected as $J_c = 14$ and is plotted as the green horizontal plane. The performance response surface intersects the performance constraint plane. Thus the POF is greater than 0.

Within the specified parameter uncertainty space, the two intersection curves of the response surface and the constraint plane, along with the $2D$ parameter uncertainty bounds, shape two shadowed regions in Figure 6. 4 (b). Any point falls into the two shadowed regions will make the performance fail to meet the constraint and contribute to POF. As long as the intersection curve can be profiled, then the POO of the shadowed regions can be calculated via discretized-summation method again. Note that such region is not limited to one. Now RQ 2.3 reduces to the following research question.

RQ 2.3.1: How to profile the intersection curve(s) when a controller is given?

Hypothesis 2.3.1: First, discretize $c - 1$ channels in the parameter uncertainty space averagely into small segments. For each segment, besides the $c - 1$ channels, the only unknown value from the remaining uncertainty channel that makes the point locate on the intersection curve can be calculated by solving Equation (5.1)-(5.2).

Proof of concept: Though the dimension increases, the $c-1$ uncertainty channels can still be discretized. For each segment in the parameter uncertainty space, the values of ΔA_i from $c-1$ channels are known. The only unknown uncertainty channel can be solved via Equation (5.1) and (5.2), as described in Section 5.1. No further explanations are needed here.

The following method significantly reduces the numerical complexity of profiling the intersection curves.

Linearization of system of second degree polynomials

This method linearizes the system of second degree polynomials from Kronecker operations method and profiles along the intersection curve via accumulating the local gradient. The merit can be easily elaborated by the following simple example:

Example 6.6: To profile the following polynomial,

$$x^2 + y^2 = 1 \tag{6. 11}$$

It is assumed that the start point $[x', y']$ is given on the curve, the next point to be profiled $[x' + \Delta x, y' + \Delta y]$ can be calculated from the next equation,

$$(x' + \Delta x)^2 + (y' + \Delta y)^2 = 1 \tag{6. 12}$$

where one of Δx and Δy is given as a small increment. The other one is to be calculated.

With the assumption that both Δx and Δy are small, the terms $(\Delta x)^2$ and $(\Delta y)^2$ can be ignored. Next, subtract Equation (6.11) from Equation (6.12), Equation (6.12) reduces to the linear Equation (6.13), where x' and y' are from previously profiled point and riding exactly on the curve.

$$\frac{\Delta x}{\Delta y} = -\frac{2y\prime}{2x\prime} = -\frac{y\prime}{x\prime} \tag{6. 13}$$

There exists extension such as remembering and incorporating results from previous steps to avoid singularity.

Apply above method on a system of equations from Kronecker operations method, e.g., Equation (5.3)-(5.6). It is assumed that an initial step is performed and the start value of

A_{11} and A_{12} is solved and stored. For the adjacent segment $A_{12} + \Delta A_{12}$, where ΔA_{12} is given and small, there is an exactly calculated new value $A_{11} + \Delta A_{11}$ that makes the point $[A_{11} + \Delta A_{11}, A_{12} + \Delta A_{12}]$ still ride on the intersection curve exactly. Then ΔA_{11} can be solved from following equation.

$$J(K, A_{11} + \Delta A_{11}, A_{12} + \Delta A_{12}) = J_c \quad (6.14)$$

The linearization method can be applied after Equation (6.14) is written in the form of a system of second degree polynomials after Kronecker operations, shown in the following example.

Example 6.7: Use the $2D2C$ example, where $K = K_{lqr} = [5.0273, -0.4142]$.

$$(2A_{11} - 10.0546) * P_{11} - 10.0546P_{12} = -26.2741 \quad (6.15)$$

$$(A_{12} + 0.4142) * P_{11} + (A_{11} - 05.0273 + 1.4142) * P_{12} - 5.0273 * P_{22} = 2.0824 \quad (6.16)$$

$$(2A_{12} + 0.8284) * P_{12} + 2.8284 * P_{22} = -1.1716 \quad (6.17)$$

$$P_{11} = 14 \quad (6.18)$$

Here P_{21} is ignored due to symmetry. Incorporate Δ terms into above system of equations so that the new solution point with a small increment still rides on the intersection curve.

$$(2A_{11} + 2\Delta A_{11} - 10.0546) * (P_{11} + \Delta P_{11}) - 10.0546(P_{12} + \Delta P_{12}) = -26.2741 \quad (6.19)$$

$$(A_{12} + \Delta A_{12} + 0.4142) * (P_{11} + \Delta P_{11}) + (A_{11} + \Delta A_{11} - 05.0273 + 1.4142) * (P_{12} + \Delta P_{12}) - 5.0273 * (P_{22} + \Delta P_{22}) = 2.0824 \quad (6.20)$$

$$(2A_{12} + 2\Delta A_{12} + 0.8284) * (P_{12} + \Delta P_{12}) + 2.8284 * (P_{22} + \Delta P_{22}) = -1.1716 \quad (6.21)$$

$$P_{11} + \Delta P_{11} = 14 \quad (6.22)$$

Also, it is assumed that all Δ terms are small so that high order terms such as $\Delta A_{11}\Delta P_{11}$ are neglected. Expand above system of equations (6.19)-(6.22) and subtract

from original system of equations (6.15)-(6.18), leaving the following system of equations

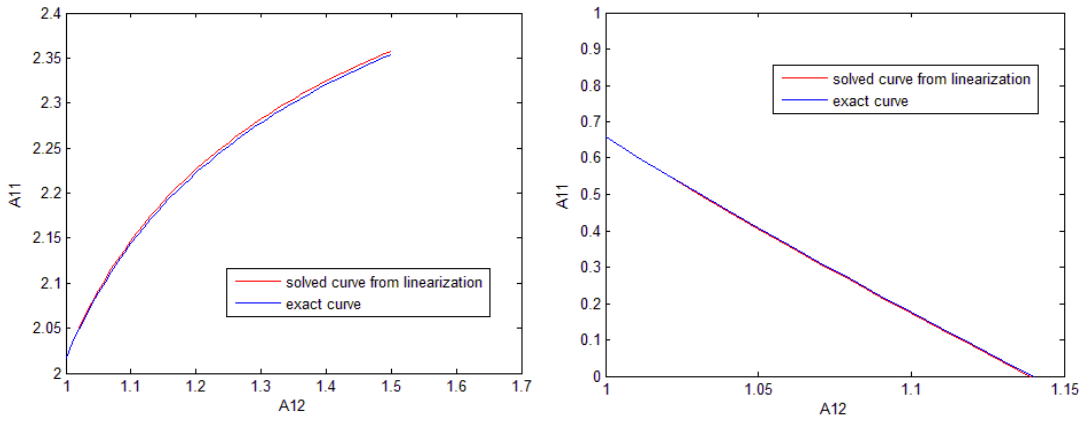
$$2P'_{11}\Delta A_{11} - 10.0546\Delta P_{12} = 0 \quad (6.\ 23)$$

$$P'_{12}\Delta A_{11} + P'_{11}\Delta A_{12} + (A'_{11} - 5.0273 + 1.4142)\Delta P_{12} - 5.0273\Delta P_{22} = 0 \quad (6.\ 24)$$

$$2P'_{12}\Delta A_{12} + (2A'_{12} + 0.8284)\Delta P_{12} + 2.8284\Delta P_{22} = 0 \quad (6.\ 25)$$

In above equations, all terms are linear and there are no crossing terms between ΔA_{11}, ΔA_{12} and ΔP (ΔP_{11} equals to 0 and is ignored). All terms with $'$ on superscript denotes the solved value from the previous step. The value of ΔA_{12} is given as a small number, e.g., $\Delta A_{12} = 0.01$, leaving three unknowns to be solved: ΔA_{11}, ΔP_{12} and ΔP_{22}. Note that due to the special structure of initial condition, only P_{11} is left from Equation (6.22); otherwise, there should be a forth linear equation composed of initial condition and ΔP terms only and this equation regulates that the calculated quadratic performance equals to J_c. In this context, above system of equations can be solved directly from the traditional linear algebraic, e.g., Gaussian elimination method.

Again, the merit of this method is to profile such "slope" $[\Delta A_{11}, \Delta A_{12}]$ instead of $[A_{11}, A_{12}]$ and it can be repeated step by step to profile the whole curve, as long as an initial point is given, e.g., from Figure 6. 4 (b) and upper half curve, $A_{11} = 2.0168$ and $A_{12} = 1$. Figure 6.5 (a) plots two profiled curves: the blue curve is solved from the exact quadratic calculations for every segment of A_{12} with increment $\Delta A_{12} = 0.01$; the red curve is profiled from linearization method with the same increment. It can be seen that the two curve have the same trend and the discrepancy is trivial. Given that with the start value $A_{12} = 1$, there are two values of A_{11} which make the point $[A_{11}, A_{12}]$ rides on intersection curve, another plot is shown below in Figure 6.5 (b) with a new start point $A_{11} = 0.6580$.



(a) Upper curve (b) Lower curve

Figure 6.5 Profiled curve from linearization method

Due to the fact that a small increment is assumed between adjacent segments, the step length has to be small and the number of segments increases. It is advised that the profiling process is "reset" after several steps by using the traditional Kronecker operations method and solving the systems of second degree polynomials. How many steps between each reset changes from case to case, i.e., a good idea is to constantly monitor the accumulated slope change of $[\Delta A_{11}, \Delta A_{12}]$. If the change between each step is small, meaning the curve is leaning toward linear, then there can be more steps allowed until the accumulated curvature goes beyond certain threshold, e.g., 1 degree.

Also, note that this method is also applicable in Chapter 5 where a line search method is used for the controller search, but is not advised there since for the case in Chapter 5, a large step length could be used, making the linearization inaccurate. For application in this section, the increment in A_{12} uncertainty channel is selected to be small and constant, thus the linearization has certain guaranteed level of accuracy. For the application to be mentioned in Section 6.4.1, the controller search path is relatively short due to small POF considerations and thus the step size would also be small and thus won't degrade linearization's accuracy.

(End of linearization of system of second degree polynomials)

Thus after the intersection curve(s) are profiled, POF can be calculated accordingly by integrating POO of the shadow region in Figure 6. 4. An example is given below to illustrate such concept.

Example 6.8: First, a two dimensional joint probability distribution is created in the following way: shrink the uncertainty space from above example to $A_{11} \in [0.5,1.5]$ and $A_{12} \in [0.5,1.5]$ so that a small POF can be captured, let

$$\Pr(A_{11}, A_{12}) = -3 * [(A_{11} - 1)^2 + (A_{12} - 1)^2)] + 1.5 \qquad (6.\ 26)$$

The PDF surface plot is shown in Figure 6.6. To validate, perform a double integral over A_{11} and A_{12} to cover the whole uncertainty space and the resulted integration equals to 1.

$$\int_{0.5}^{1.5}\int_{0.5}^{1.5}(-3 * [(A_{11} - 1)^2 + (A_{12} - 1)^2)] + 1.5)dA_{11}dA_{12} = 1 \quad (6.\ 27)$$

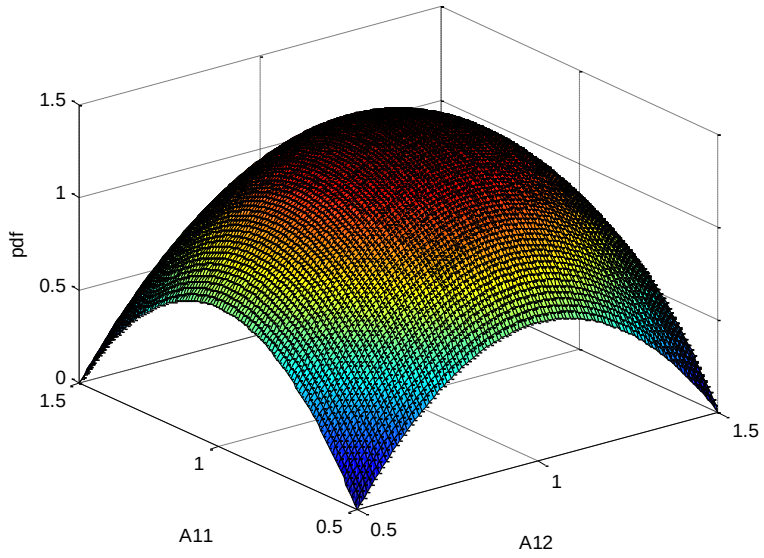


Figure 6.6 $\boldsymbol{2D}$ joint probability distribution

Using the joint probability distribution created above, the calculated POF is shown below,

$$POF_{discretize} = 0.0478 \cong POF_{linearized} = 0.0471 \qquad (6.\ 28)$$

Note that for $2D2C$ example, the $2D$ region can be discretized and summed easily by discretizing one uncertainty channel, e.g., channel A_{12} in above case; for high dimensional cases, it could be costly to calculate POF, e.g., for a $3D3C$ case with uncertainty channels in $[A_{11}, A_{12}, A_{13}]$, there is a need to discretize and sweep the $[A_{11}, A_{12}]$ uncertainty space and calculate A_{13} on the intersection point for each segment in $[A_{11}, A_{12}]$. The next step is to integrate the calculated volume of each $2D$ segment over the $3D$ uncertainty space and thus leads to POF. However, it would be still more efficient compared to the sampling method with small POF to be captured.

Since there is a need to discretize the uncertainty space and integrate, an initial point selection is very important and a lot of information can be revealed after the first-time solution of Kronecker equations. A good selection of start point helps to avoid sweeping the whole uncertainty space. E.g., in above $2D2C$ example, if one starts from the bound $A_{12} = 1$ and sweep in A_{12} channel towards the other bound $A_{12} = 1.5$, it would be way more efficient, as it stops right around point $A_{12} = 1.14$, compared to the case of selecting the start point from the other bound $A_{12} = 1.5$ and sweep backwards the whole uncertainty range.

RQ 2.3.2: How to effectively select an initial point?

Hypothesis 2.3.2: Always starts from the pair of corners/bounds that gives $J_{minimax}$.

Proof of concept: As long as the controller deviates from $K_{minimax}$, it is always one from above mentioned pair of corners/bounds that yields performance violations first, since the constraint is tight initially.

RQ 2.3.3: How to select which dimensions to discretize and sweep through?

Hypothesis 2.3.3: There is no particular preference. Discretize the uncertainty channel with a relatively smaller uncertainty range would be favorable since there will be fewer segments with the same discretization granularity.

Thus the overall process of calculating POF is summarized below:

Algorithm 3	***POF* calculation**
Step 1	With a given controller, first select $c-1$ uncertainty channels from the parameter uncertainty space and discretize them.
Step 2	Sweep these channels. For each discrete segment, solve Equation (6.14) by using Kronecker method to calculate the value of unknown uncertainty channel. Check its validity such as whether the calculated value falls into uncertainty space.
Step 3	Calculate corresponding POO for each segment according to given probability distribution, sum up each segment's POO to get POF.

6.4 Controller search direction considerations

6.4.1 Constant *POF* contour search method

As briefly mentioned in Section 6.1, the line search could be further executed for better average performance by searching along the constraint POF contour after the POF constraint is hit for the first time. It brings out the next research question.

RQ 2.4: How to find the search direction along the constraint and constant POF contour?

Hypothesis 2.4: A ***delta POF based method*** can be used to find a search direction along the constraint and constant POF contour.

The merit of this method is to calculate the increment of POF from the increment in the controller. Recall that in the $2D2C$ case, for each segment in Figure 6. 4 (b), the value of A_{12}, K_1 and K_2 are given and A_{11} is calculated accordingly. When there is a given small increment ΔK_2 in the value of K_2 and another small increment ΔK_1 in the value of K_1, the increment of calculated value of A_{11}, denoted as ΔA_{11} is a function of ΔK_1 and ΔK_2 for each segment where A_{12} is known. The new value of $A_{11}+\Delta A_{11}$ profiles a new intersection curve, as shown in the dashed red curve in Figure 6.7.

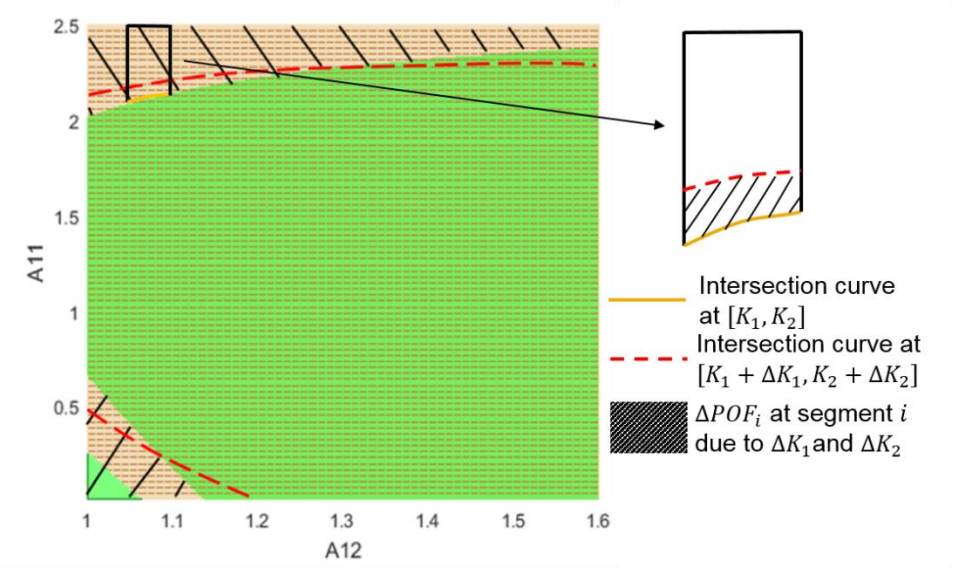


Figure 6.7 New intersection curve with increment of $\boldsymbol{K_2}$

The multiplications of each segment's POO and the calculated ΔA_{11} summed up to the increment of POF (ΔPOF). Let it be assumed that ΔK_2 is small and given, then the value of ΔK_1 that makes $\Delta POF = 0$ can be calculated accordingly.

From mathematical point view, the task of above method is that for each segment, calculate ΔA_{11_i} as a function of ΔK_1 when ΔK_2 is given as a small increment, so that the following equations hold true

$$J\left(A_{11_i}, A_{12_i}, K_1, K_2\right) = J\left(A_{11_i} + \Delta A_{11_i}, A_{12_i}, K_1 + \Delta K_1, K_2 + \Delta K_2\right) = J_g \quad \forall i = 1 \dots d \tag{6. 29}$$

Then the ΔPOF due to ΔK_1 and ΔK_2 is calculated via the following equation, where $\Delta POF_i = \Delta A_{11_i} \Delta A_{12} \Pr\left(A_{11_i}, A_{12_i}\right)$ is the calculated ΔPOF for each segment. Regulating the following sum of ΔPOF_i equals to 0 enables ΔK_1 be solved as a function of ΔK_2, and the normalized combination of ΔK_1 and ΔK_2, which essentially forms a search direction, is a valid one that makes the line search along the constant POF contour.

$$\Delta POF = \sum_{i=1}^{d} \Delta A_{11_i} \Delta A_{12} \Pr\left(A_{11_i}, A_{12_i}\right) = 0 \tag{6. 30}$$

Note that the linearization method can still be applied here, the only difference is that there is an extra unknown ΔK_1 in the equations, along with an extra linear equation $\Delta POF = 0$. So the total unknowns and equations in the system are still balanced.

Example 6.9: For each segment with A_{12} constant and given, Equation (6.29) reduce to the following after linearization.

$$2P'_{11}\Delta A_{11} - 10.0546\Delta P_{12} + 2K_1\Delta K_1 = 0 \quad (6.31)$$

$$P'_{12}\Delta A_{11} + (A'_{11} - 5.0273 + 1.4142)\Delta P_{12} - 5.0273\Delta P_{22} + K_2\Delta K_1 + K_1\Delta K_2 = 0 \quad (6.32)$$

$$(2A'_{12} + 0.8284)\Delta P_{12} + 2.8284\Delta P_{22} + 2K_2\Delta K_2 = 0 \quad (6.33)$$

Note that due to the value of controller changes, thus there will be linearized terms containing ΔK_1 and ΔK_2 from the term $Q + K^T RK$. Similar to the previous case, in above linear equations, the value of ΔK_2 is given as a small number, i.e., $\Delta K_2 = 0.01$, leaving four unknowns ΔA_{11}, ΔK_1, ΔP_{12} and ΔP_{22} to be solved from three equations. In such circumstance, there are infinity solutions since the equations are underdetermined, but ΔA_{11} can be expressed as a linear function of ΔK_1. Note that due to special structure of initial condition, only P_{11} is left from last equation; otherwise, there should be a forth linear equation composed of the initial condition and ΔP only to regulate the calculated quadratic performance equals to the given value.

Repeat above calculation for each discretized segment of A_{12_i} with the same value of ΔK_2, the sum of ΔA_{11_i} is still a linear function of ΔK_1 solely and thus a unique value of ΔK_1 can be solved from equation (6.30). The direction of $[\Delta K_1, \Delta K_2]$ is the one that enables the line search go along the constraint and constant POF contour.

Still use the $2D2C$ example, but some simplifications are made so that the plot shown below can be visually validated: a uniform distribution, or $Pr(A_{11}, A_{12}) = 1$ is assumed over the whole uncertainty range, so that the PDF integrated area, or simply the area of region shaped by the intersection curve and two axes is equivalent with POF; the

uncertainty range is further shrinked to $A_{11} \in [0,0.5]$ and $A_{12} \in [1,1.1]$ so that only left lower part of shadowed uncertainty region in Figure 6. 4 (b) will lead to performance violation. With the same controller, calculated $POF = 0.1796$. Though it is against initial assumption of small value, but this example only serves with an illustrative purpose, thus it is reasonable that POF has a relative large value.

Now let $\Delta K_2 = 0.1$ and $\Delta A_{12} = 0.01$. Perform one step of above search direction calculation, the calculated $\Delta K_1 = 0.02728$. Figure 6.8 plots the intersection curves of two response surfaces with the constraint plane together. It can be observed that from a bird view, the two intersection curves form two quasi-triangle regions and they overlap each other in the middle. Visually inspect and it reveals that the areas of two regions are the same with each other. Numerical result shows that $POF(K_1, K_2) = 0.1796 \cong 0.1801 = POF(K_1 + 0.02728, K_2 + 0.01)$.

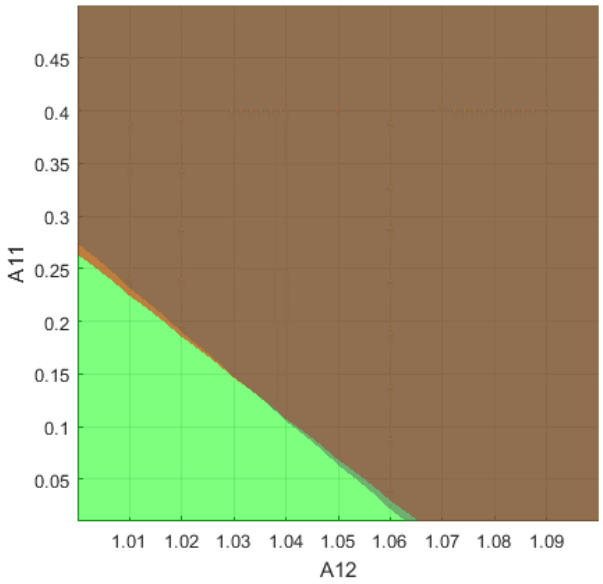


Figure 6.8 New intersection curve with an increment of $\boldsymbol{K_2}$ (calculated result)

One interesting property to check is that, after linearization, the geometrical interpretation of such search direction along the constant POF contour is the local slope,

which should yields two search directions who are opposite with each other along the local slope.

Hypothesis 2.4.1: If the sign of ΔK_2 is changed but the absolute value is kept constant, the new calculated value of ΔK_1 should also be the same, but only with a different sign.

Proof of concept: The LHS of Equation (6.31)-(6.33) are linear combinations of constants and variables, and the RHS is 0 for all rows. That means as long as one of the variables in the solution changes its sign, all other variables in the new solution keep the same absolute value, but with different signs. The search direction from the newly calculated results is equivalent with the previously calculated search direction shifts 180 degrees.

Also, note that above conclusion is only valid after linearization. With original quadratic equations, changes of sign won't lead to above conclusion.

6.4.2 Stopping criteria

RQ 2.5: When to stop line search along the constraint and constant POF contour?

Hypothesis 2.5: The line search stops when the product of the average performance's steepest descent direction and the constant POF search direction is negative. Equivalently, the projection of one direction on the other is negative.

Proof of concept: The concept is easy to understand via the useful and feasible directions. No numerical example but a notational example is shown in Figure 6.9. Utilization of such concept through a comprehensive case study can be found in the next chapter.

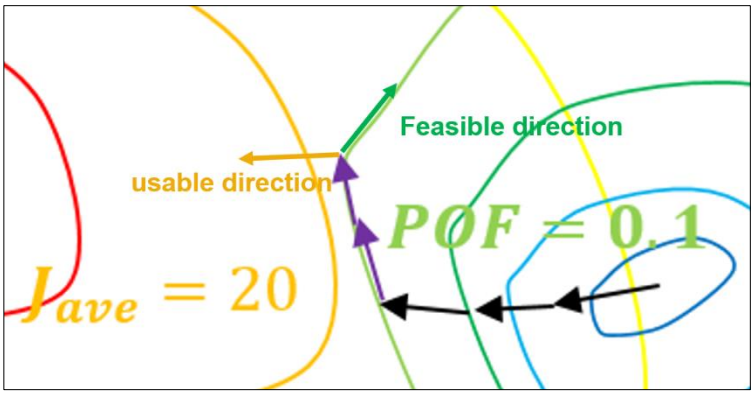


Figure 6.9 Notational example of two directions against each other

In above example, from the geometry point of view, the line search stops when the feasible direction and the usable direction go against each; algebraically, the product of the two research directions' vectors is negative, or the projection of one vector on the other is in an opposite direction [38].

It has been proved before that the average performance over controller design space is strictly non-concave, thus for sure the optimal point being found is also global optimal.

6.4.3 Extension to high dimension case

The generalized procedure from above method is that for a controller with $m * n$ dimensions, pre-determine $m * n - 1$ dimensions and calculate the unknown one so that it goes along the constant and constraint POF contour. E.g., for a controller with 3 dimensions, the $2D$ contour of circles shown in Figure 6. 2 expands to a $3D$ contour of spheres shown in Figure 6. 10.

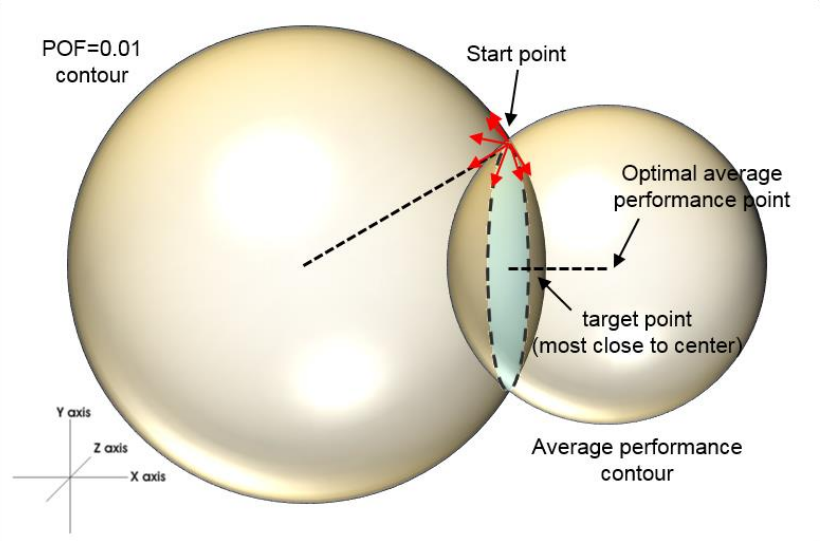


Figure 6. 10 3D POF and performance contour

For any point on the sphere, there are infinite tangent lines, highlighted in red arrows. Though above method guarantees a search direction towards better performance, it cannot guarantee the steepest descent direction, which exists uniquely from pre-selected controller's $m * n - 1$ dimensions. Such concern leads to the next research question.

RQ 2.6: How to find the best combination of controller's pre-determined $m * n - 1$ dimensions?

Hypothesis 2.6: The $m * n - 1$ dimensions can be "borrowed" from steepest descend search direction towards optimal average performance. An example is used to illustrate above concept.

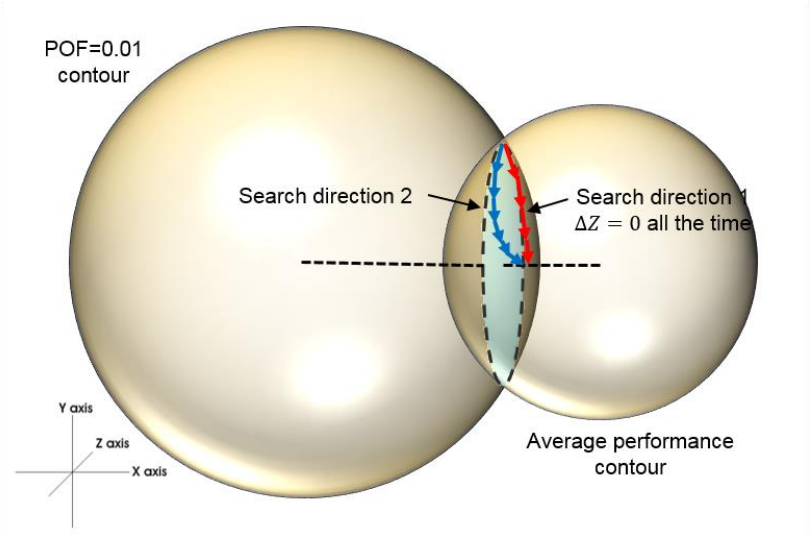


Figure 6.11 $\boldsymbol{3D}$ controller line search

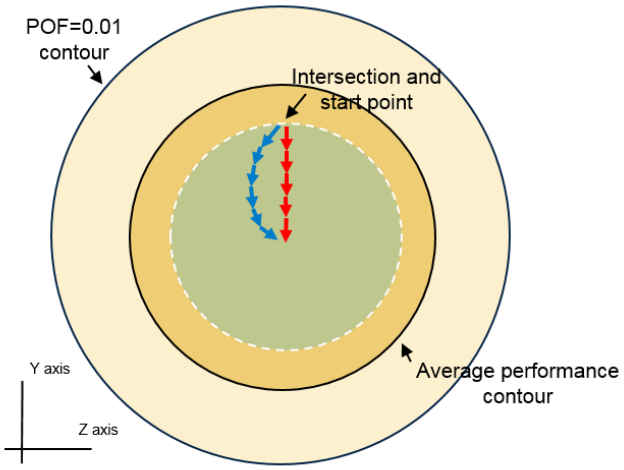


Figure 6.12 $\boldsymbol{3D}$ controller line search-side view

Example 6.10: Figure 6. **10** shows two spheres, denoting the constant POF and the average performance contour, intersect each other. To make it easier, it is assumed that the controller who gives the optimal average performance is located exactly on the center point of RHS sphere and all other constant average performance contours are all spheres. Thus the optimal point to be found who optimizes average performance while rides on the $POF = 0.01$ contour is located exactly on the intersection point of the line connecting

the center points of two spheres and the $POF = 0.01$ contour. Also it is assumed that the start line search point is located on the highest point on the intersection circle.

The steepest descent direction at the start point is $[\Delta x = 0.5, \Delta y = -0.5, \Delta z = 0]$. Thus, the two dimensions of $\Delta y = -0.5$ and $\Delta z = 0$ are borrowed and the calculated unknown dimension is $\Delta x = 0.1$. Thus the combined initial search direction is $[\Delta x = 0.1, \Delta y = -0.5, \Delta z = 0.0]$. Keep using the two dimensions borrowed from steepest descent direction, the path of line search is plotted in red arrows in Figure 6.11. Note that since Δz always equals to 0 through all steps, thus it is always valid to scale Δy to 0.5.

Compare to another search path highlighted in blue who deviates from the red one, it turns out that from Figure 6.12 who is the side view of Figure 6.11, though the path in blue still rides on constant POF contour and leads to the optimal point in the end, it is not as efficient as the line search path in red.

RQ 2.7: For high dimension cases, which dimensions to "borrow" from the steepest descent direction?

Hypothesis 2.7: There is no guaranteed answer. One possible answer would be to find the largest sum of $m * n - 1$ dimension's local gradient.

Proof of concept: As shown in Equation (6.34), geometrical meaning of the following equation is to maximize potential performance improvement at the next step of line search.

$$\max \qquad sum = \sum_{i=1}^{m*n-1} \frac{\partial J_{ave}}{\partial K_i} \tag{6. 34}$$

It is not guaranteed since there is variation introduced from the calculated unknown dimension, as it is derived from the POF contour, which is totally independent from the constant performance contour. Since it is not of the priority, further discussion of this topic will be addressed in the future.

6.5 POF constraint optimal average performance controller design

So far, Section 6.2 provides a method to search and update a controller consistently towards a better average performance. Section 6.3 provides a method to calculate POF for a given controller. Section 6.4 provides a method to search along the constraint POF contour and a stopping criteria. The combination all above fulfill a complete numerical line search method for the problem formulated in Section 6.1.3. Next a comprehensive procedure is provided so that a POF Constrained Optimal Average Performance Controller can be found to reduce the conservatism.

Algorithm 4	**Line search to a POF constrained optimal average performance controller**
Step 1	Start from $K_{minimax}$ with a performance equals to the evaluation criteria $J_{minimax} = J_c$, calculate a steepest descent direction towards the optimal average performance controller.
Step 2	Perform the line search and calculate current controller's POF at each search step, repeat until the constraint POF is reached.
Step 3	Find a new search direction along the constant POF contour by using the ΔPOF method; make sure the direction goes towards the better average performance; keep performing line search until the two directions from step 2 and step 3 go against each other.
Step 4	Check average performance, stability, etc.

6.6 Complexity analysis

As addressed in Section 5.6, the complexity of solving the Lyapunov equation will no longer be analyzed in this section. Instead, the focus will be shifted to analyze the complexity of the line search method.

Since there is no analytical method available, above numerical line search method will be compared to the sampling method. Given that the probability generations and calculations for both methods have the same complexity, the only thing left for

comparison is the size of discretized segments and the complexity of each line search step against the sampling method.

Should both methods aim to design a controller with an accuracy within 5% error around target $POF = 0.01$, as analyzed in RQ 2.3, there needs 100000 samples in each uncertainty channel. Also it is assumed that there are c uncertainty channels, total the number of samples increases exponentially and is of the order of magnitude $1 * e5 * c$, or equivalently $c * e5$ times of Lyapunov equation calculation. The complexity is of the order $O(n^c)$. Should one consider both uncertainty space and controller design space with $m * n$ dimensions to sample to design an optimized controller, the complexity increases to $O(d^{c+m*n})$.

It is assumed that discretization have the same granularity of sampling in the uncertainty channel, when the number of uncertainty channel increases, only the dimension of search direction increases linearly, or $O(r(m+n)d^c)$. Thus even it takes extra r numerical steps to find the controller search direction compared to sampling, it is still numerically efficient when the number of uncertainty channels increases.

Undeniable, sampling method is much easier to program as the discretization and line search method require attentions all the time to decide the validity of the calculated results, which channels and dimensions to choose from, etc.

6.7 Extensions

A useful extension of the linearization method in Section 6.3.2 is its direct application in the matrix calculations, i.e., in Lyapunov equation without Kronecker operations.

Start from the following Lyapunov equation, assuming that K is given. Instead of $'$, use $^\circ$ to denote the previously calculated value of each variable.

$$\left(\begin{bmatrix} A_{11}^{\circ} & A_{12}^{\circ} \\ A_{21} & A_{22} \end{bmatrix} + BK\right)^T P^{\circ} + P^{\circ}\left(\begin{bmatrix} A_{11}^{\circ} & A_{12}^{\circ} \\ A_{21} & A_{22} \end{bmatrix} + BK\right) + Q + K^T R K = 0 \quad (5.42)$$

$$x_0^{\mathrm{T}} P^{\circ} x_0 = J_c \quad (5.43)$$

With an increment of ΔA_{11} and ΔA_{12}, the Lyapunov equation becomes the following

$$\left(\begin{bmatrix} A_{11}^{\circ} & A_{12}^{\circ} \\ A_{21} & A_{22} \end{bmatrix} + \begin{bmatrix} \Delta A_{11} & \Delta A_{12} \\ 0 & 0 \end{bmatrix} + BK\right)^{T} (P^{\circ} + \begin{bmatrix} \Delta P_{11} & \Delta P_{12} \\ \Delta P_{21} & \Delta P_{22} \end{bmatrix}) + (P^{\circ} + \begin{bmatrix} \Delta P_{11} & \Delta P_{12} \\ \Delta P_{21} & \Delta P_{22} \end{bmatrix}) \left(\begin{bmatrix} A_{11}^{\circ} & A_{12}^{\circ} \\ A_{21} & A_{22} \end{bmatrix} + \begin{bmatrix} \Delta A_{11} & \Delta A_{12} \\ 0 & 0 \end{bmatrix} + BK\right) + Q + K^{T}RK = 0 \quad (5.\ 44)$$

$$x_0^{\mathrm{T}}(P^{\circ} + \begin{bmatrix} \Delta P_{11} & \Delta P_{12} \\ \Delta P_{21} & \Delta P_{22} \end{bmatrix})x_0 = J_c \quad (5.\ 45)$$

By assuming that ΔA_{11}, ΔA_{12} and all ΔP terms are small, expand above equation, ignore high order terms and eliminate from Equation (5.42)-(5.43), the following system of equations is finally reached,

$$\begin{bmatrix} \Delta A_{11} & \Delta A_{12} \\ 0 & 0 \end{bmatrix}^{T} P^{\circ} + (A^{\circ} + BK)^{T} \begin{bmatrix} \Delta P_{11} & \Delta P_{12} \\ \Delta P_{21} & \Delta P_{22} \end{bmatrix} + P^{\circ} \begin{bmatrix} \Delta A_{11} & \Delta A_{12} \\ 0 & 0 \end{bmatrix} + \begin{bmatrix} \Delta P_{11} & \Delta P_{12} \\ \Delta P_{21} & \Delta P_{22} \end{bmatrix} (A^{\circ} + BK) = 0 \quad (5.\ 46)$$

$$x_0^{\mathrm{T}} \begin{bmatrix} \Delta P_{11} & \Delta P_{12} \\ \Delta P_{21} & \Delta P_{22} \end{bmatrix} x_0 = 0 \quad (5.\ 47)$$

All terms in above system of equations are linear. Without loss of generality, there are two unknowns from uncertainty channels, four from matrix P and five balance equations. Thus as long as either one of ΔA_{11} or ΔA_{12} is assumed to be known, the other unknowns can be solved accordingly. In this context, above method is equivalent with the method provided in Section 6.3.2.

CHAPTER 7

CASE STUDY: HVAC CONTROL SYSTEM DESIGN

In this chapter, a case study of HVAC control system design will be provided. First, a physical model is built, followed by the state space representations, simplification, and linearization. Uncertainty channels will be identified next. Corresponding probability distributions are retrieved from observed sensor data. The state of art solutions, mainly adaptive and robust control will be surveyed. Then the proposed NEMCD and PCOAPCD will be used in a sequence to design a controller to reduce the conservatism. A comparison will be made between multiple controllers visited previously to highlight each controller's feature and fulfilled design objective. In the end, two other potential case studies, an airplane control and a financial control will be briefly mentioned.

The following flow chart diagram is provided to illustrate the solution procedure of the whole case study. It starts from building a model with normal LQR, identifying uncertainty parameters' ranges and thus designing a traditional robust minimax controller.

Next, ***Experiment 1*** will be carried out by utilizing the Norm Extend Minimax Controller Design method. Thus, a minimax controller can be designed so that the first level of conservatism from traditional minimax method can be reduced.

With the incorporation of uncertain parameters' probability distribution information and applying POF Constrained Optimal Average Performance Controller Design method, a free line search (***Experiment 2)*** will be implemented towards the optimal average performance controller. ***Experiment 3*** will focus on designing a new controller so that the tight performance constraint is relaxed and a tradeoff between average performance, worst-case performance and POF can be initiated. With the achievements from Experiment 2 and Experiment 3, the second level of conservatism can be reduced.

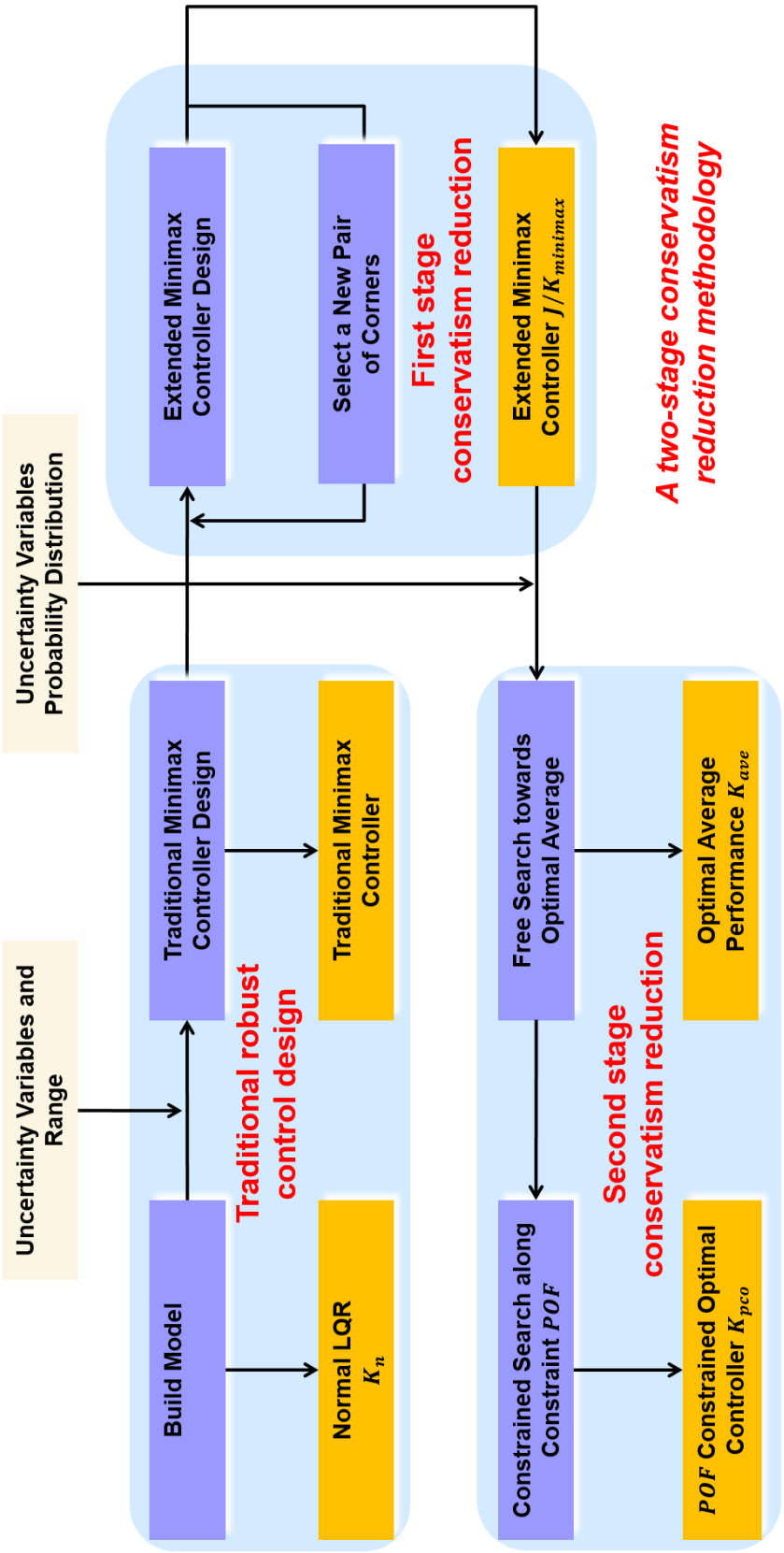


Figure 7.1 Methodology flow chart diagram

7.1 HVAC system introduction

HVAC (heating, ventilation, and air conditioning) is the technology of indoor and vehicular environmental comfort. It represents an important share of the electricity consumption (about 30%) in a building [39]. Considering the fact that buildings consume approximately 40% of total US energy [40], HVAC's share of total US energy is around 12%. This fact implies that important energy and economic savings can be achieved by improving the efficiency in HVAC system and therefore, a substantial reduction in the environmental impacts can be also achieved.

While an HVAC system involves many aspects such as design, construction, operation and maintenance, design is the foundation. For the HVAC system, the main goal is to ensure indoor and vehicular environmental comfort. Along with technology development, energy consumption and environmental friendly [41] have been used as other evaluation criteria. The HVAC industry is now regulated by standards organizations such as ASHRAE, International Mechanical Code. A series of regulations [42] have been established to support the industry and encourage high standards and achievement.

With above analysis and the following reasons, HVAC system design is selected as the case study in this research:

1. It is relatively simple compared to other high dimension/order systems.
2. The quadratic performance *J* has a flexible weight on controller's effort (energy usage), compare to other controller design criteria.
3. When room temperature is treated as the tracking target, performance deviates in both directions are not favored, which fits the quadratic performance evaluation criteria used in this research.
4. Has sensor data available

From the physical aspect, a complete HVAC system has multiple variations serving different design objects but is usually composed of an ***Air Handling Unit*** (AHU) and a ***conditioned room***. Figure 7.3 shows the structure of a traditional AHU. Its function is to

provide the desired amount of chilled or heated air to the monitored room to maintain a comfortable environment. The cooling coil in AHU serves as a heat exchange platform so that heat contained in return air can be absorbed by cooling medium and delivered to the air-cooled condenser. In this research, only cooling will be considered.

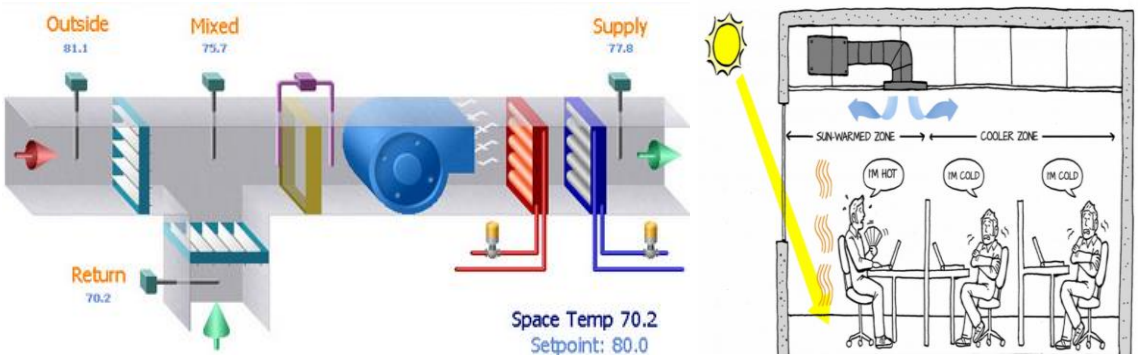


Figure 7.3 Air Handling Unit (AHU) structure [43] and monitored room

The monitored room is shown in Figure 7.3, i.e. a fully occupied classroom. It can be considered as a thermal model that obeys the energy conservation law. To simplify the model, all the air in the room is considered to have the same temperature, which is affected by the gained heat or lost heat. The heat gained or lost is determined by the room heat load, which is a function of supply air temperature, flow rate, and other heat sources. The corresponding equation will be given in next section. Other monitored metrics include humidity, CO2 density, and pressure. Some of them are heavily coupled with room temperature [44], but these metrics are not considered to simplify the system.

With above two physical plants defined, there is a control system embedded. For example, the particular air flow rate needed to maintain a constant room temperature is decided by the controller Variable Air Volume (VAV) [45]. The desired cooling medium is controlled by a Variable Speed Compressor so that the supply air has a regulated temperature. While there are multiple cooling medium and strategies available for the cooling purpose, cooling refrigerant and DX evaporator is used in this case study.

7.2 Traditional/robust/adaptive control applications in HVAC

In this section, a literature survey will be performed WRT to traditional PID control, optimal control, adaptive control, and robust control applications in HVAC system design.

7.2.1 PID/PI Controller Application in HVAC

It is proved that PI controller (without derivative term) is sufficient enough for HVAC system design [46].

To get a desired HVAC performance, tuning PID controller parameters is the most important. Since stability is not a critical point in the HVAC control system, the controller gain is usually set to be low. But it also leads to tedious and inaccurate response [46]. While there are many well-developed parameters tuning algorithms available in the modern industry, they are not necessarily applicable to HVAC controller design. A lot of research and papers were published to address such challenge. Nesler [47] provided a method to select PI parameters in the digital control of discharge air temperature. Hittle [48] also provided a solution in terms of parameter tuning.

As the control theory develops, the PI controller is gradually replaced by more advanced controllers due to its limitation such as energy inefficient, poor robust performance to disturbances. Nowadays many HVAC controller design papers with research purposes only use PI controller as a reference for a performance comparison to their proposed controllers.

7.2.2 Optimal Control Application in HVAC

Optimal control also has wide application in HVAC systems, especially considering that it takes over 12% of total US energy consumption. Yahiaoui [49] utilized LQG controller for an integrated building system. The objective function included both state variables and control effort. It proved that with LQG controller applied, the objective of simultaneously optimizing room comfort and energy consumption could be accomplished.

Similar results were observed by Zaheer-Uddin [50] by comparing LQG controller and traditional PI controller, though his objective function focused on disturbance rejection. Other effort was contributed from Kasahara, et al. [51]. Instead of a physics model, they built a multivariable autoregressive (AR) HVAC model from the experimental data. Optimal preview controller was used with an objective function of maintaining the room comfortableness. Their results were calculated through computer simulation and were validated by experiment.

7.2.3 Adaptive Control Application in HVAC

The key concept of adaptive control is that the controller gains are automatically tuned "on-line". When it is applied on HVAC, its benefit is evident as the discrepancy from desired performance can be compensated. Thus, performance degradation is recovered.

Nesler [47] developed an evolutionary approach in a effort of automated tuning methods for a traditional PI controller. Similar work was done by Park [52] as his work presented a recursive least-squares algorithm. Instead of PI controller, adaptive control can also be applied to optimal control, such as the work done by Wang [53].

For the derivations from traditional adaptive control, Sheikholeslami [54] used a brain emotional learning algorithm to control a multivariable HVAC system. Soyguder [55] added the fuzzy logic into a traditional PID type adaptive control. The neural network techniques applications in HVAC system are presented in many papers. Ferreira [56] adopted a neural network to obtain the room thermal comfort as well the energy saving. Saboksayr [57] used a similar approach to reach the energy saving for a system with decentralized controllers.

7.2.4 Robust control application in HVAC

The applications of robust control in HVAC systems are surveyed in this section. A basic application of traditional robust control method was done by Underwood [46]. The

results proved that though there was a loss of performance, the robust control did increase the stable margin when the plant had a varying gain.

Chen and Lee [2] proposed an adaptive robust controller to take into account of the uncertainties including thermal storage effect, heat, and moisture generation, or outside temperature and humidity changes. The assumption was that the uncertainty was bounded, but the bound was unknown. A comparison was made between the proposed controller and on/off control.

Kasahara, et al. [51] developed a "two-disk type, mixed sensitivity method" to solve PID parameters and got a robust design for an HVAC system with different lag time and percentage of parameter variation. A comparison was made between the proposed solution and a traditional PID parameter tuning method using Ziegler-Nichols rule.

7.3 System Modeling

In this section, an physical HVAC system will be modeled, followed by the realization, simplification, and linearization which convert it to a linear state space equation so that Experiment 1, 2 and 3 can be performed respectively.

7.3.1 Physical Modeling

The model to be used in this section is referred from [58] [59], in which a model was built from physics and validated against experiment data. In this research, simplifications of the model will be made, followed by assigning several parameters with new values from local sensor data. In this way, the uncertainties collected from sensors could be mapped into the model. The physical model is shown below.

In terms of the conditioned room, with the assumption of a perfect air mixing and the thermal inertia of indoor air leading to slow changes, the first equations is derived from the principle of energy conservation: the heat gained, including VAV delivered via the thermal difference between the supply air temperature T_1 and the room temperature T_2,

the space sensible heat load Q_{load} and the heat gain of the supply fan Q_{load}, drive the room temperature variation $\frac{dT_2}{dt}$,

$$C_p\rho V\frac{dT_2}{dt} = C_p\rho f(T_1 - T_2) + Q_{load} + Q_{spl} \quad (7.1)$$

where C_p is the thermal capacity, ρ is the air density and is assumed to be constant, V is the volume of conditioned space, f is the air volumetric flow rate.

The heat gain of supply fan increases with air flow rate, and can be written as

$$Q_{spl} = k_{spl}f \quad (7.2)$$

where k_{spl} is a coefficient.

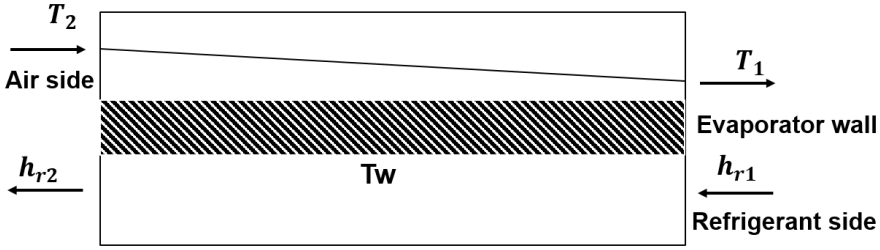


Figure 7.4 DX evaporator

The DX evaporator is shown in Figure 7.4. Free energy exchange is assumed between air and refrigerant. It is also assumed that the wall temperature T_w is the same through the whole wall due to a large thermal conductivity and small region on the wall. Again the principle of energy conservation is applied,

$$C_p\rho V_{h1}\frac{dT_1}{dt} = C_p\rho f(T_2 - T_1) + \alpha_1 Z_1(T_w - \frac{T_2+T_1}{2}) \quad (7.3)$$

where Z_1 is the air side heat transfer area of evaporator. V_{h1} is the air side volume of evaporator and is calculated as follows,

$$V_{h1} = Z_{s1}L_1 \quad (7.4)$$

where L_1 is the length of the region on the air side of evaporator. Z_{s1} is the air side cross area of the evaporator.

The air side convective heat transfer coefficients α_1 for the louver-finned evaporator [60] is evaluated as follows,

$$\alpha_1 = j_{e1}\rho v \frac{C_p}{Pr^{\frac{2}{3}}} \tag{7.5}$$

where j_{e1} is the Colburn factors, Pr is the *Prandtl* number, v is the air velocity.

Due to the significant difference of the thermal inertia between refrigerant and air, the dynamic responses to the changes on the air side is much slower than that on the refrigerant side. When the air side waits for a long time to fully respond, the refrigerant side is already in its steady-state for a quite a while. Thus, the same refrigerant mass flow rate at both inlet and outlet of the DX evaporator is assumed and written as follows,

$$M_{ref} = \frac{sV_{com}}{v_s}\left\{1 - 0.015\left[(P_c/P_e)^{\frac{1}{\beta}} - 1\right]\right\} \tag{7.6}$$

where s is the compressor speed, all other variables are thermal constants and are not further elaborated.

With all above assumptions, the energy balance equation for the evaporator wall can be written as follows,

$$(C_p\rho V)_w \frac{dT_w}{dt} = \alpha_1 Z_1 \left(\frac{T_2+T_1}{2} - T_w\right) - M_{ref}(h_{r2} - h_{r1}) \tag{7.7}$$

In this manner, there are three state variables $x = [T_1, T_2, T_w]^T$, two input signals $u = [f, s]^T$, along with the disturbance Q_{load}, corresponding to the last paragraph in Section 7.1. The state space equation's parameters are listed below.

$$A = \begin{bmatrix} \frac{1}{C_p\rho V_{h1}}(-C_p\rho f - \frac{1}{2}\alpha_1 Z_1) & \frac{1}{C_p\rho V_{h1}}(C_p\rho f - \frac{1}{2}\alpha_1 Z_1) & \frac{1}{C_p\rho V_{h1}}\alpha_1 Z_1 \\ \frac{1}{C_p\rho V}C_p\rho f & -\frac{1}{C_p\rho V}C_p\rho f & 0 \\ \frac{1}{(C_p\rho V)_w}(\frac{1}{2}\alpha_1 Z_1) & \frac{1}{(C_p\rho V)_w}(\frac{1}{2}\alpha_1 Z_1) & -\alpha_1 Z_1 \end{bmatrix}$$

The controller matrix

$$B = \begin{bmatrix} 0 & 0 \\ k_{spl} & 0 \\ 0 & -\frac{V_{com}}{v_s}\{1 - 0.015\left[\left(\frac{P_c}{P_e}\right)^{\frac{1}{\beta}} - 1\right]\}(h_{r2} - h_{r1}) \end{bmatrix}$$

Disturbance matrix

$$d = \begin{bmatrix} 0 \\ Q_{load} \\ 0 \end{bmatrix}$$

The values of constants are listed in Table 6.

Table 6 Constants

$\boldsymbol{C_p}$	$\boldsymbol{1.005\ kJ/kg}$	$\boldsymbol{(C_p \rho V)_w}$	$\boldsymbol{29kJ}$
$\boldsymbol{\rho}$	$1.2\ kg/m^3$	$\boldsymbol{A_1}$	$4.14m^3$
$\boldsymbol{V}$	$120m^3$	$\boldsymbol{V_{h1}}$	$0.04m^3$

7.3.2 Linearization

Since the model contains a lot of interaction terms and thus is non-linear, it needs to be linearized before LQR design can be applied.

In majority of the cases of linearization, the system is designed to be operated in the vicinity of an operation set point. As long as the control system can properly regulate the dynamic deviation of the controlled objectives from their set points, the controlled system can be well represented by a linearized model around the set points. Hence, the state variables x and control inputs u can be expressed as follows, respectively,

$$x = x_0 + \Delta x \quad (7.8)$$

$$u = u_0 + \Delta u \quad (7.9)$$

where x_0 and u_0 are the state and input variables evaluated at the steady-state operation point. At the same time, due to the fact that the load disturbance changes very slowly, it is

assumed to be a constant and thus disappears after the linearization. The change in the compressor speed is much slower than that of the supply fan speed and thus Δs is also removed.

By implementing above linearization, the system is linearized to the following equations,

$$C_p\rho V_{h1}\frac{d\Delta T_1}{dt} = C_p\rho f_s(\Delta T_2 - \Delta T_1) + C_p\rho(T_{2s} - T_{1s})\Delta f + \alpha_1 Z_1(\Delta T_w - \frac{\Delta T_2+\Delta T_1}{2}) \quad (7.\ 10)$$

$$C_p\rho V\frac{d\Delta T_2}{dt} = C_p\rho f_s(\Delta T_1 - \Delta T_2) + C_p\rho(T_{1s} - T_{2s})\Delta f + k_{spl}\Delta f \quad (7.\ 11)$$

$$(C_p\rho V)_w\frac{d\Delta T_w}{dt} = \alpha_1 Z_1\left(\frac{\Delta T_2+\Delta T_1}{2} - \Delta T_w\right) \quad (7.\ 12)$$

Now, the new state space representations are as follows, given that the new state variables are $x = [\Delta T_1, \Delta T_2, \Delta T_w]^T$, and the controller output variables $u = [\Delta f]^T$.

$$A_s = \begin{bmatrix} \frac{1}{C_p\rho V_{h1}}(-C_p\rho f_s - \frac{1}{2}\alpha_1 Z_1) & \frac{1}{C_p\rho V_{h1}}(C_p\rho f_s - \frac{1}{2}\alpha_1 Z_1) & \frac{1}{C_p\rho V_{h1}}\alpha_1 Z_1 \\ \frac{1}{C_p\rho V}C_p\rho f_s & -\frac{1}{C_p\rho V}C_p\rho f_s & 0 \\ \frac{1}{(C_p\rho V)_w}(\frac{1}{2}\alpha_1 Z_1) & \frac{1}{(C_p\rho V)_w}(\frac{1}{2}\alpha_1 Z_1) & -\alpha_1 Z_1 \end{bmatrix}$$

$$B_s = \begin{bmatrix} C_p\rho(T_{2s} - T_{1s}) \\ k_{spl} + C_p\rho(T_{1s} - T_{2s}) \\ 0 \end{bmatrix}$$

For the DX HVAC system, system is operated at the point $T_1 = 13.25℃$, $T_2 = 24℃$, $T_w = 13℃$ and $f = 0.347\ m^3/s$. A_s and B_s are calculated as follows,

$$A_s = \begin{bmatrix} -7.1177 & -0.7177 & 7.8354 \\ 0.0017 & -0.0017 & 0 \\ 0.0065 & 0.0065 & -0.3780 \end{bmatrix}$$

$$B_s = \begin{bmatrix} 12.9645 \\ -9.0111 \\ 0 \end{bmatrix}$$

7.3.3 Normal controller design

With the normal model built, a normal controller can be designed to regulate the system's performance by using LQR. The selection of weights Q and R in the performance equation is available from many methods. They can be decided directly if the preference between system's transient performance and control efforts is known in advance; a pole placement method can be used if there is desired performance already in mind. The method used in this case study is try and error [61], i.e., a set of values of Q and R are first selected and a LQR is calculated. The simulated performance is plotted and other evaluation criteria such as maximum overshoot value are examined. Then the weights are tuned accordingly to get the desired performance.

In this research, the selection of Q and R is not of the interest and is selected only to properly scale the transient performance and controller effort: $Q = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}$ and $R = 1000$. The calculated LQR is $K = [0.00093, -0.0314, 0.00127]$ with $J_{LQR} = 6.5359$. The transient performance and controller output are plotted against an initial room temperature disturbance of $\Delta T_2 = 1$ degree.

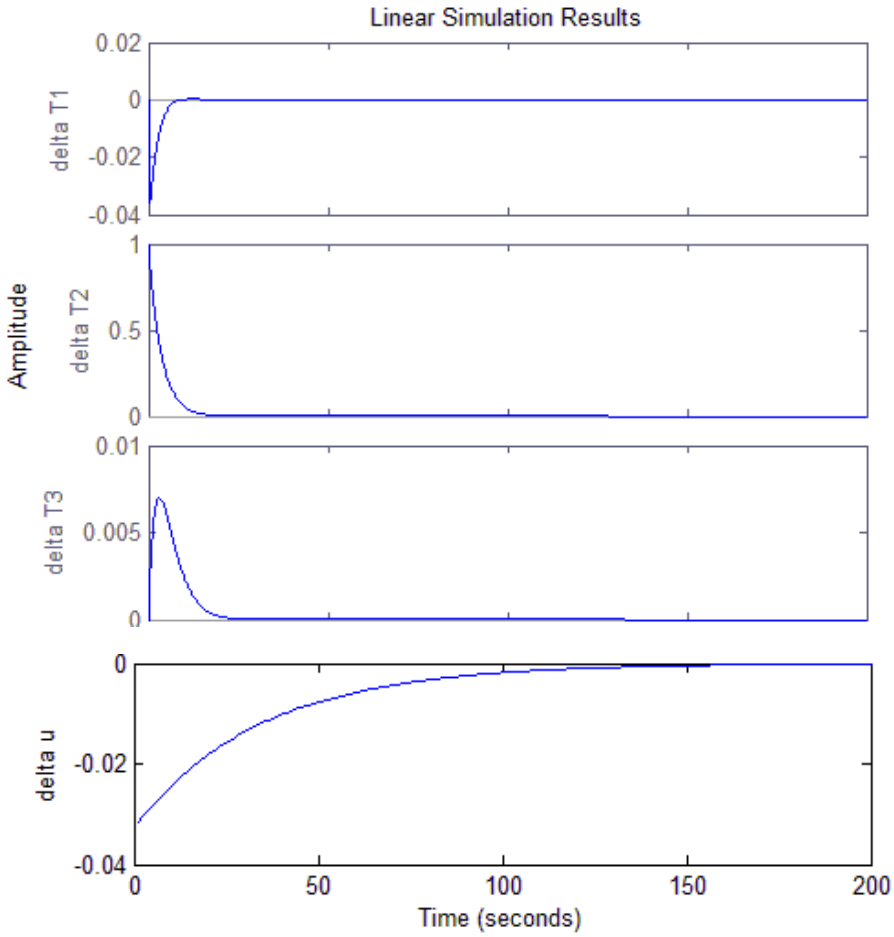


Figure 7. 5 Transient performance with $\boldsymbol{K_{LQR}}$

It can be seen that system is first-order and the response time is about 20 seconds, which is satisfying; since the model is assumed to be linear, controller saturation, if exists, is not within the concern in this research. Also note that a lot of LQR design purposely add an integrator to ensure LQR's signal tracking performance. However, since the signal tracking performance is not the priority in HVAC design, it is not adopted in this research.

7.4 Uncertainty identification

For the monitored room model as shown in Equation (7.10)-(7.12), it suffers from all three types of uncertainties identified in Section 2.4 and only parameter uncertainty is considered in this research.

7.4.1 Uncertainty channels identification

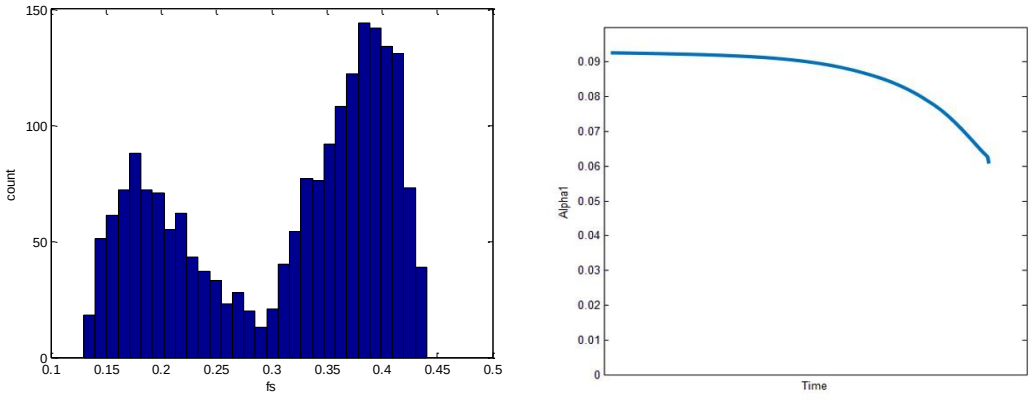
Through linearization, the supply air flow rate f_s is assumed to be constant at an operation point. However, it deviates when the operation point shifts. Thus the associated elements $(A_{11}, A_{12}, A_{21}, A_{22})$ in matrix A have associated uncertainty ranges and corresponding probability distributions. Note that the value of f_s is always the same for all the four elements and the four elements can be treated as a single uncertainty channel.

On the other side, the air side convective heat transfer coefficient α_1 is decided by the louver fin's material. Both erosion, rust and potential wearing will degrade the coefficient. Thus there is also uncertainty associated with it and the affected elements include $(A_{11}, A_{12}, A_{13}, A_{31}, A_{32}, A_{33})$. Similarly, they can be treated as a single uncertainty channel.

Thus, the system to be dealt with in this case study is a $3D2C$ system. Note that the two channels are independent from each other.

7.4.2 Probability incorporation

The probability distribution of f_s is collected from around 2000 sampled sensor data with a sampling frequency a quarter hour within a time span about a month. The PDF is shown in Figure 7.6 (a).



(a) (b)

Figure 7.6 Probability distribution of f_s and α_1

The two separated clusters are due to the different settings in the day and night schedule. According to the ASHRAE handbook [62], for a classroom there is a need of 4-10 air change per hour. Thus the maximum value of $f_s \cong 0.441 m^3/s$ corresponds to about 13 air change per hour at fully occupied hours while the minimum value of $f_s \cong 0.128 m^3/s$ corresponds to about 4 air change per hour at night. Thus the value of f_s has the following uncertain range $f_s \in [0.128, 0.441]$.

Although a polynomial fit can be used to profile the probability distribution, this research uses a lookup table with a high granularity and linear interpolation if the value to be looked up falls between two discrete values, instead of profiling the PDF curve. The reason is, no analytical method will be used when the probability information is involved, thus a lookup table not only provides a calculation efficiency, but also preserves the accuracy.

The probability distribution of α_1 is not directly measurable, but the degradation level can be estimated [63]. Given the normal convective heat transfer rate is calculated as $\alpha_1 = 0.0913$, the degraded worst-case rate is selected as $\alpha_1 = 0.06$. The degradation curve (against time) is plotted in Figure 7.6 (b). It is assumed that degradation is slow in the beginning and accelerated when it approaches the end of life circle. Thus the value of α_1 has the following uncertain range $\alpha_1 \in [0.06, 0.0913]$.

7.5 Experiment 1: NEMCD

For the $3D2C$ uncertainty system, there are two channels and four corners $[f_s, \alpha_1] \in [0.128, 0.0913], [0.441, 0.0913], [0.128, 0.06], [0.441, 0.06]$ to be examined. Name above four corners 1,2,3,4 respectively for easier reference.

To apply NEMCD, first step (Step 1 in Algorithm 2) is to calculate each corner's LQR (let K_{LQR_i} denotes K_{LQR} at corner i) and make a comparison (Step 2 in Algorithm 2). The result is shown in Table 7.

Table 7 Experiment 1: $\boldsymbol{LQR}$ for all four corners

Corner	$[\mathbf{0.128}, \mathbf{0.0913}]$	$[\mathbf{0.441}, \mathbf{0.0913}]$	$[\mathbf{0.128}, \mathbf{0.06}]$	$[\mathbf{0.441}, \mathbf{0.06}]$
$\boldsymbol{J(K_{LQR_1})}$	6.5359	6.6450	8.1149	7.7624
$\boldsymbol{J(K_{LQR_2})}$	6.6489	6.5201	8.1383	7.5025
$\boldsymbol{J(K_{LQR_3})}$	6.5534	6.5669	8.0891	7.6099
$\boldsymbol{J(K_{LQR_4})}$	6.7241	6.5305	8.1928	7.4908

It turns out that this corner 3 has the worst performance for all four calculated LQR_i. Thus the worst-case point is located at this corner exclusively. In this context, there is only a need to calculate LQR at corner 3. Thus $K_{minimax} = K_{LQR_3} = [0.0009773, -0.03179, 0.00199]$. The calculated $J_{minimax} = J\big([0.128, 0.06], K_{LQR_3}\big) = 8.0891$.

Remark: Though there is no need to apply Algorithm 2 Step 3-6 on above case study and makes it less interesting, but it will make PCOAPCD easier as when it comes to the performance violation, there is only one corner who yields performance violation when the controller doesn't deviate from $K_{minimax}$ too much. Anyway, the numerical examples in Chapter 5 should be sufficient enough to demonstrate NEMCD.

7.6 Experiment 2: PCOAPC, free search

With the calculated $K_{minimax}$ and $J_{minimax}$ and the probability distribution information, this section starts from a free line search towards K_{ave} without the POF constraint. The line search trace is plotted in Figure 7.7.

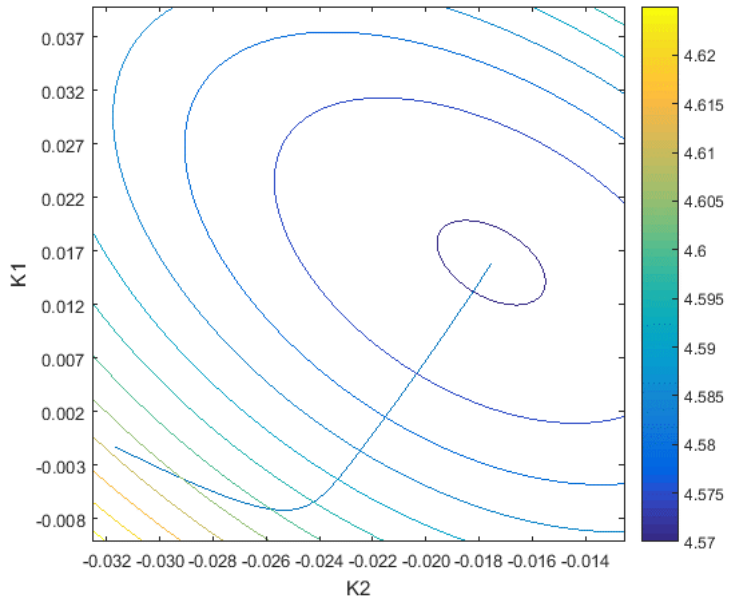


Figure 7.7 Experiment 2: free search towards optimal average controller

With $B_s \in \mathbb{R}^{3*1}$, the controller search space is $3D$. But there is no way to view a $3D$ contour plot. Thus only a $2D$ contour plot is shown with $K_3 = 0.0361$ constantly, which is exactly the value from K_{ave}. It can be seen from Figure 7.7 that the controller search does yield a satisfying result, as the line search converges right to the center of the contour. Also note that when the line search starts (left bottom corner), the value of K_3 is far different from 0.0361 and thus the contour plot is totally different from the one shown in Figure 7.7. This is why the contour and the line search direction are not perpendicular with each other at the left bottom corner.

7.7 Experiment 3: PCOAPC, POF constrained search

To validate PCOAPC, the POF constraint is set to 0.01 with the evaluation criteria from Experiment 1 $J_{minimax} = 8.0891$. The uncertainty channel f_s is averagely discretized into 100 segments and each time POF is calculated, it is the uncertainty channel α_1's value being solved as the unknown with the value of f_s given for each discretized segment.

For each step in the free line search towards K_{ave}, POF is consistently monitored. It turns out that at the point $K = [0.0078, -0.0201, -0.0280]$, constraint POF is reached. Then a line search along the constraint and constant POF contour is performed via ΔPOF method. The new line search trace is shown in Figure 7.8.

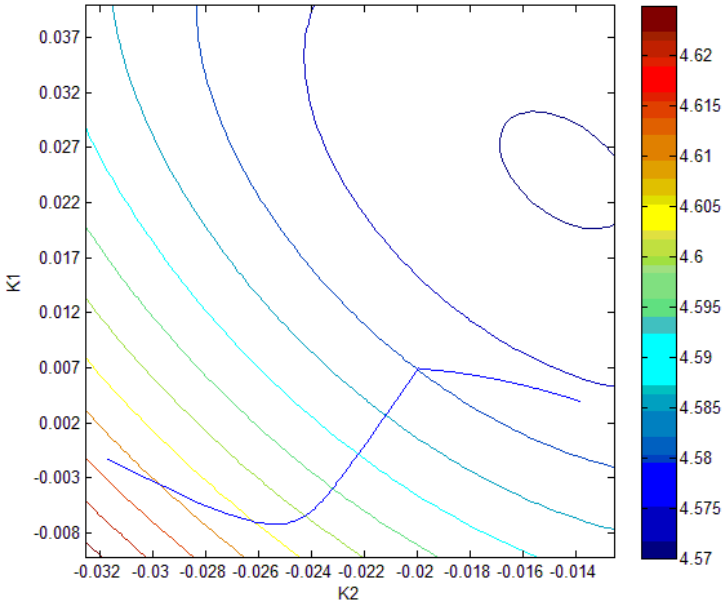


Figure 7.8 Experiment 3: free search and search along constraint POF contour

By using PCOAPC, the direction of K_3 is "borrowed" from the steepest descent search direction. The corresponding $\Delta\alpha_1$ for each segment of f_s is calculated with given $\Delta K_2 = 0.00001$ and as a function of ΔK_1. Then each segment's POO is calculated based on $\Delta\alpha_1$ and summed up to get ΔPOF so that ΔK_1 is solved for each line search step by regulating $\Delta POF = 0$. For a validation purpose, similar to Figure 6.8, a probability truncate plot with two intersection curves is shown in Figure 7. 9.

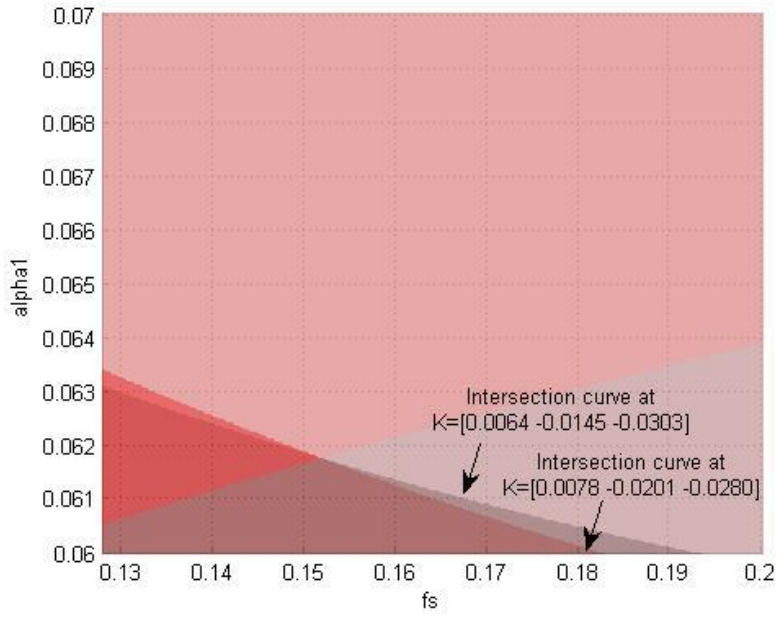


Figure 7. 9 POF truncate plot

The two intersection curves from left middle to right bottom shape two quasi-triangle regions, whose mathematical meaning is that any point falls into these regions violates the performance constraint and contributes to POF for a given controller. Though the detailed probability distribution is not shown in the plot, and also note that the uncertainty space is not the whole uncertainty space (Figure 7. 9 only shows $f_s \in [0.128,0.2]$ and $\alpha_1 \in [0.006,0.007]$), the calculated POF of the two regions all equals to the POF constraint of 0.01, which is exactly the geometrical meaning of searching along the constant and constraint POF.

It can be seen that the two curves intersect each other in the middle (around $f_s = 0.152, \alpha_1 = 0.0618$). There is another intersection curve generated from the two performance response surfaces and comes from the left bottom to the right middle. It plays no role and thus no more analysis is given.

Also, note that there is no POF contour shown in Figure 7.8 since it is very numerically expensive to profile the whole POF contours.

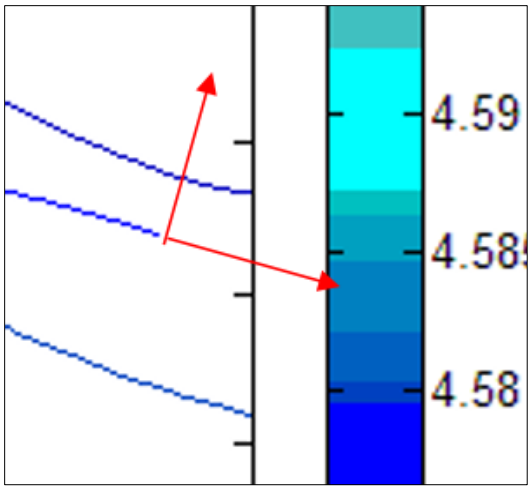


Figure 7.10 Two search directions perpendicular to each other

It turns out that the line search stops at $K = [0.0064, -0.0145, -0.0303]$. From Figure 7.10 it can be seen that the two search directions, highlighted in red arrows, are perpendicular with each other and thus the line search stops as there is no useful and feasible direction further exists.

7.8 Results analysis

A summary of comparisons between visited controllers is listed in Table 8. Three metrics are used: the worst-case performance J_{max}, average performance J_{ave} and POF.

Table 8 Controllers comparison

	K	J_{max}	J_{ave}	POF
Normal LQR	$[0.00093, -0.0314, 0.00127]$	8.1160	4.6013	0.06
Minimax (robust)	$[0.00098, -0.0319, -0.0020]$	8.0891	4.6122	0
Optimal average	$[0.0161, -0.0176, -0.0361]$	8.1437	4.5697	0.07
POF constraint	$[0.0064, -0.0145, -0.0303]$	8.3761	4.5791	0.01

With normal LQR (K_n), it gives a mediocre performance: neither best nor worst among the three metrics. With the minimax robust controller $(K_{minimax})$, as expected it yields the best worst-case performance $J_{max} = 8.0891$, and thus POF strictly equals to 0. With the optimal average performance controller (K_{ave}), it gives the best average performance $J_{ave} = 4.5697$, though it has the largest POF as well a relatively poor worst-case performance. Further analysis yields that though K_{ave} further degrades the worst-case point, which is located on the corner 3 where $(fs, \alpha_1) = [0.128, 0.06]$, but it also improves the performance at corner 2 where $(fs, \alpha_1) = [0.441, 0.0913]$, which has a larger probability distribution weight compared to corner 3 and thus is more favorable in terms of the average performance improvement. The POF constrained optimal average performance controller K_{pco} gives an even worse worst-case performance, but it also takes advantage of the probability distribution weight at other corners and thus gives the second best average performance while makes sure that the POF still meets the constraint.

In this context, with the incorporation of probability information and the tradeoff between worst-case performance, average performance and POF, the conservatisms brought from the traditional worst-case based robust control are reduced.

7.9 Other applications

Besides the HVAC system, the proposed methodology can be applied to any control system exposed to parameter uncertainties, as long as the system can be written in the form of a linear state space equations and the quadratic performance J is used as the evaluation criteria.

7.9.1 Aircraft Control

Aircraft control is another good case study, where there are more uncertainties: aircraft weight which decreases as fuel is consumed, CG location [64], aerodynamic force and moment, airplane's attitude and corresponding control mode [65], etc. They are contained

in the following simplified dynamic equations as uncertain parameters [66], which include $\mu, C_L, C_D, C_W, i_{yy}, \sigma, \eta$.

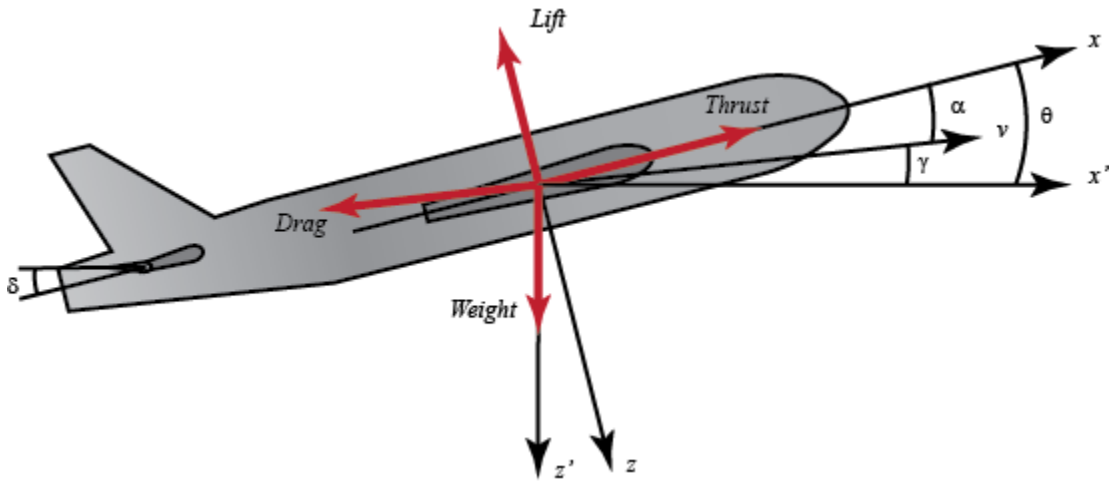


Figure 7. 11 Airplane in flight [66]

$$\dot{\alpha} = \mu\Omega\sigma[-(C_L + C_D)\alpha + \frac{1}{(\mu - C_L)}q - (C_W \sin\gamma)\theta + C_L]$$

$$\dot{q} = \frac{\mu\Omega}{2i_{yy}}[[C_M - \eta(C_L + C_D)]\alpha + [C_M + \sigma C_M(1 - \mu C_L)]q + (\eta C_W \sin\gamma)\delta]$$

$$\dot{\theta} = \Omega q$$

Commonly, the airplane's performance regulations are contained in the transient performance domain such as overshoot value, rise time, settling time, and steady-state error. Thus, the quadratic performance J can be well applied in this case, as long as a relatively large value is selected for weight matrix Q. What makes it even more interesting is that stability would become critical evaluation criteria. Even 0.01% stability POF is not acceptable.

Note that above equations are simplified from real physics, which contains high order and non-linear terms. Thus, airplane case study is not selected in this research. Another reason that it is not selected is due to the lack of data, thus the uncertain parameters' ranges and probability distribution information cannot be retrieved accurately enough.

7.9.2 Financial Control

In non-physical field, such as financial control, the uncertainties and variations can be amplified even it is small at the beginning since the response process is slow and many things could trigger the occurrence of parameter uncertainty.

The stock market is an example of the system prone to oscillatory "hunting", governed by positive and negative feedback resulting from cognitive and emotional factors among market participants [67]. For example, when stocks are rising (a bull market), the belief that further rises are probable gives investors an incentive to buy (positive feedback—reinforcing the rise); but the increased price of the shares, and the knowledge that there must be a peak after which the market falls, ends up deterring buyers (negative feedback—stabilizing the rise).

Once the market begins to fall regularly (a bear market), some investors may expect further losing days and refrain from buying (positive feedback—reinforcing the fall), but others may buy because stocks become more and more of a bargain (negative feedback—stabilizing the fall).

How to predict investors' behavior is very important in the stock market as it produces huge uncertainty in the market. Since it is occupant related, ***Agent-Based Modeling*** would be a good method to predict uncertainty parameter's probability distributions. The merit of the proposed methodology could get well applied in the financial control model.

However, it is not selected neither due to the author's lack of expertise in the financial area.

CHAPTER 8

CONCLUSION

This chapter concludes this thesis with conclusions, a summary of originality and contributions, and potential extensions to be performed in the future.

8.1 Conclusion

In this thesis, the research objective is first proposed, followed by the introduction of the plant and control system design. After the introduction of three types of uncertainties, especially the parameter uncertainty who changes the system's performance, the research objective is then highlighted. A literature survey is performed and yields that robust/adaptive controls are the state-of-art solutions.

However, it is also observed that the main gap in robust control, mainly minimax control, is that the designed controller is over-conservative since it has a norm bounded uncertainty range (first level of conservatism) and is based on the worst-case scenario and thus lacks the performance characteristic that could be achieved for the most likely cases (second level of conservatism).

In order to reduce such conservatisms, several research questions and hypotheses are proposed. The deliverable is a Two-stage Conservatism Reduction Methodology including a Norm Extended Minimax Control Design method to reduce the first level of conservatism and a POF Constrainted Optimal Average Performance Controller Design method to reduce the second level of conservatism. The validations of these hypotheses to the research questions are discussed in detail in each chapter and summarized in Table 9.

Table 9 Summary of research questions and hypotheses

Index	Core content	Validation method	Section
1.1	The worst-case point is always located on bound/corner	Theoretical	5.2
1.2	Intersection point of two performance curves is a candidate	Observation led	5.3.2
1.3/1.4	Intersection point can be analytically calculated when there is one unknown in the controller	Mathematical equations	5.4.1/5.4.2
1.5	There will be infinite intersection points when there are more than one unknowns in the controller	Mathematical equations	5.4.3
1.6/1.7	Use the Triangle based gradient method and line search to find a global minimax controller	Mathematical equations/numerical algorithm	5.5/5.6
2.1	Discretize the whole uncertainty space to calculate average performance	Mathematical equations	6.2
2.2	Discretize whole uncertainty space to find a direction towards optimal average performance	Mathematical equations	6.2
2.3	Discretize the regions profiled by intersection curves to calculate the POF	Mathematical equations/numerical algorithm	6.3.2
2.4	Discretize the regions profiled by intersection curves to calculate a search direction along constraint the POF contour	Mathematical equations/numerical algorithm	6.4
2.5	Stop the line search when useful and feasible directions are against each other	Theoretical proof	6.4

A case study of HVAC control system design is given to demonstrate the whole design method. The HVAC control system design is selected due to the fact that it has relatively simple physical representations and mathematical equations; the energy save is

much favored due to HVAC system's high portion in total US energy consumption; there is real sensor data available.

After the physical model is built, simplified, and linearized, a normal LQR is calculated as the normal controller. Next, the uncertain parameters within the plant are further identified. Their uncertain ranges and the associated probability distributions are created from the sensor data.

Next, several experiments are carried out in the case study in a sequence to fulfill the proposed methodology so that the conservatisms from traditional worst-case based robust control design method are reduced. These experiments are summarized in the following.

Experiment 1 utilizes the Norm Extend Minimax Controller Design method. First, all corner points within the parameter uncertainty space are identified, followed by calculating LQR and corresponding performance for each corner point. A comparison is made and yields that the worst-case point is located at one corner point exclusively. Thus the Norm Extend Minimax Controller is equivalent with the LQR at this corner.

After the worst-case performance with reduced conservatism is derived from ***Experiment 1*** and used as the performance evaluation criteria, ***Experiment 2*** and ***Experiment 3*** can be performed by using the POF Constrained Optimal Average Performance Controller Design method after the incorporation of uncertain parameters' probability distribution information.

After the whole parameter uncertainty space is averagely discretized and each discretized segment's POO is calculated from the probability distribution, a free line search is performed in ***Experiment 2***. The search direction is updated at every step via calculating the local gradient of the average performance WRT most current controller. The start point of the line search is exactly the minimax controller from Experiment 1. The line search ends at the optimal average performance controller.

Experiment 3 focuses on designing a new controller so that the tight performance constraint is relaxed and a tradeoff between the average performance, worst-case

performance and POF can be initiated. Beyond the procedures used in ***Experiment 2***, ***Experiment 3*** extends it further by constantly monitoring the POF for each search step. After the POF constraint is met for the first time, a new search direction along the constraint and constant POF contour is calculated by using the ΔPOF method. The line search stops when the two directions from previous steps go against each other. The POF constrained optimal average performance controller is then reached as the final deliverable.

A comparison of visited controller is given to highlight each controller's fulfilled design objective. The normal LQR gives a mediocre performance, as it is only designed for the normal plant, without considering any parameter uncertainty. As expected, the minimax robust controller from ***Experiment 1*** yields the best worst-case performance among all visited controllers and a POF strictly equals to 0. Without the POF constraint, the optimal average performance controller from ***Experiment 2*** gives the best average performance at a cost of the largest POF as well a relatively poor worst-case performance. As the final deliverable, the POF constrained optimal average performance controller from ***Experiment 3*** gives the second best average performance while makes sure that the POF still meets the constraint.

In this context, the feature and advantage of the POF constrained optimal average performance controller is evident: with the incorporation of probability information and the tradeoff between worst-case performance, average performance and POF, the two levels of conservatisms brought from the traditional worst-case based robust control are reduced.

8.1 Originality and Contribution

A summary of proposed methodology is listed in Table 10 by comparing to the robust controller design method, in terms of the selection/method of evaluation criteria, design/noise variables identification, and the methods for design space exploration.

Table 10 Comparison between Robust Control Method and Proposed Method

Category	Robust Control System Design	Proposed Method
Evaluation criteria	Stability Others (viability)	POF and average performance
Identify design variables, noise variables	Controller type Parameter uncertainty range	Controller type Parameter probability distribution
Evaluation methodology	Dynamic simulation System estimation Find response extremes	Norm extended minimax performance Tradeoff between average performance/POF
Design space exploration	Robust control design from response extremes	Properties analysis Gradient based and numerical line search method

In terms of the evaluation criteria, the "hard" constraint is relaxed to "soft" the constraint in the form of POF. In terms of the design variables identification, both methods consider the controller as the design variables, which is true for any type of control system design. In terms of the noise variables identification, robust control only cares about their uncertain ranges, but the proposed method needs uncertain parameters' probability distributions besides uncertain ranges. Robust control only seeks the system's response extreme (worst-case scenario) from the specified uncertain range while the proposed method uses numerical methods to calculate POF. The proposed method also provides line search methods to find a norm extended minimax controller as well a POF constrained optimal average performance controller.

To summarize, some concepts are innovatively applied in the control system design, such as extending the norm-bounded uncertainty space, optimal average performance controller, the tradeoff between POF and average performance. While it is true that some of the techniques utilized are not new and many of them are well studies over the century, such as the line search, discretized-summation, linearization, but it is unique and original to integrate aforementioned techniques into a control system design methodology. Such organized and integrated methodology is the originality and the major contribution from this thesis.

8.2 Future work

In this thesis, there are several topics briefly mentioned but not expanded due to the size of the thesis, or not research interest, such as a possible proof of position of the worst-case point via Hessian Matrix, the mathematical proof of necessary and sufficient conditions for an intersection point to be the worst-case point. They will be covered in the future research work. More case studies will be addressed, should there are real data and models available.

So far the uncertainty is limited to the system characteristic matrix A. Such assumption can be expanded to controller matrix B, even observer matrix C, D, as the LQE is of dual with LQR. The extension also covers weight matrix Q, R, if the designer has a non-constant preference between the transient performance and controller effort. For example, in the HVAC example, there could be different weights in Q, R at different time, i.e., at day time, system's performance in time domain is emphasized while at night, transient performance requirement is not of great importance while the energy save is more preferred. Thus there will be a large value in Q at day time and a large value in R at night time, if proposed method extends to weight matrix Q, R.

The concept of the tradeoff between the average quadratic performance and POF can be extended to other domains, e.g., stability, as long as a mapping can be created from the

controller to the selected performance evaluation criteria. Correlations between uncertainties can also be brought into consideration.

APPENDIX I
KRONECKER MATRIX ALGEBRA

Let vec operator is given as below

$$\text{vecA} \triangleq \begin{bmatrix} \text{col}_1(A) \\ \vdots \\ \text{col}_m(A) \end{bmatrix} \in \mathbb{R}^{nn} \tag{I.1}$$

Let $A \in \mathbb{R}^{n*m}$ and $B \in \mathbb{R}^{l*k}$. Then the Kronecker product $A \otimes B \in \mathbb{R}^{nl*mk}$ of A and B is the partitioned matrix

$$A \otimes B \triangleq \begin{bmatrix} A_{(1,1)}B & A_{(1,2)}B & \cdots & A_{(1,m)}B \\ \vdots & \vdots & \ddots & \vdots \\ A_{(n,1)}B & A_{(n,2)}B & \cdots & A_{(n,m)}B \end{bmatrix} \tag{I.2}$$

Let $A \in \mathbb{R}^{n*n}$ and $B \in \mathbb{R}^{m*m}$. Then the Kronecker sum $A \oplus B \in \mathbb{R}^{nm*\mathrm{n}\mathrm{m}}$ of A and B is

$$A \oplus B \triangleq A \otimes I_m + I_n \otimes B \tag{I.3}$$

Let $A, B, C \in \mathbb{R}^{n*n}$. Then, there exists a unique matrix $X \in \mathbb{R}^{n*n}$ satisfying

$$AX + XB + C = 0 \tag{I.4}$$

If and only if $B^T \oplus A$ is nonsingular. If $B^T \oplus A$ is nonsingular, then X is given by

$$X = -\text{vec}^{-1}[(B^T \oplus A)^{-1}\text{vec}C] \tag{I.5}$$

If, in addition, $B^T \oplus A$ is asymptotically stable, then X is given by

$$X = \int_0^\infty e^{At} C\, e^{Bt} dt \tag{I.6}$$

Proof of first statement: first, note that equation (I.4) is equivalent to

$$\begin{aligned} 0 &= \text{vec}(AXI + IXB) + \text{vec} \\ &= (I \otimes A)\text{vec}X + (B^T \otimes I)\text{vec}X + \text{vec}C \\ &= (I \otimes A + B^T \otimes I)\text{vec}X + \text{vec}C \\ &= (B^T \oplus A)B^T \oplus A + \text{vec}C \end{aligned} \tag{I.7}$$

Thus equation (I.7) has a unique solution vec$X$ if and only if $B^T \oplus A$ is nonsingular, which confirms the first statement.

Next, it is assumed that $B^T \oplus A$ is nonsingular, and it follows from equation (I.7) that

$$\text{vec}X = -(B^T \oplus A)^{-1}\, \text{vec}C$$

That proves equation (I.5).

The proof of the second statement is not shown as it is not of interest here.

REFERENCES

[1] Goudaa, M. M., Underwood, C. P. and Danahera, D., "Modelling the robustness properties of HVAC plant under feedback control," BUILDING SERV ENG RES TECHNOL November 2003 vol. 24 no. 4 271-280.

[2] Chen, Y.H., Lee, M. and Wepfer, W.J., "ADAPTIVE ROBUST CONTROL SCHEME APPLIED TO A SINGLE-ZONE HVAC SYSTEM," IEEE, American Control Conference, page 1076 – 1081, San Diego, CA, USA, 23-25 May 1990,

[3] Curricular Linux Environment at RICE University, "Controlling Engineering Systems"

[4] "A Brief Building Automation History" http://www.building-automation-consultants.com/building-automation-history.html

[5] Bennett, S., "A history of control engineering", 1930-1955. IET. p. p. 48 (1993)

[6] Bernstein, D.S. and Tsiotras, P., "Optimal Control." Princeton Univ. Press, Princeton, NJ, 2008.

[7] Doyle, J. C., "Guaranteed Margins for Lqg Regulators," IEEE Transactions on Automatic Control, vol. 23, no. 4, pp. 756–757, 1978.

[8] Landaum, I. D., "Adaptive Control: The Model Reference Approach." New York: Marcel Dekker, 1979.

[9] Kant, K., "Computer-Based Industrial Control," PHI Learning Pvt. Ltd., 2004, p558

[10] K. M. Passino, Yurkovich, S., "Fuzzy Control," Addison Wesley Longman, Menlo Park, CA, 1998

[11] Nguyen, D.H., "Neural networks for self-learning control systems," IEEE Control System Magazine, 1990

[12] Galperin, G. and Viola, P., "Machine Learning for Prediction and Control," MIT

[13] Zames, G. (1996). "Input-output feedback stability and robustness, 1959-85". Control Systems Magazine, IEEE 16 (3): 61–66

[14] Urban, T.J., "Synthesis of Missile Autopilots Robust to the Presence of Parametric Variations," MS thesis

[15] Cha ńho, J., Pereira, P., Rafael, S. and Pires, A.J., "A Simple PID Controller with Adaptive Parameter in a dsPIC," Proceedings of the 9th Spanish-Portuguese Congress on Electrical Engineering, Marbella, Spain, 30 June–2 July 2005.

[16] Lavretsky, E., "Adaptive Control: Introduction, Overview, and Applications."

[17] Landau, I.D., “From Robust Control to Adaptive Control,” Control Eng.Practice, vol 7,no10, pp1113-1124.

[18] Annaswamy, A.M., Lavretsky, E., Dydek, Z.T., Gibson, T.E. and Matsutani, M., “Recent results in robust adaptive flight control systems,” International Journal of Adaptive Control and Signal Processing, Volume 27, Issue 1-2, pages 4–21,January-February 2013.

[19] Siqueira, A. A. G., Terra, M. H. and Bergerman, M., “Robust Control of Robots”.

[20] Michelle, K.R., Dimitri M.N., “Forecasting Technology Uncertainty in Preliminary Aircraft Design,” 4th World Aviation Congress and Exposition, San Francisco, CA, October 19-21, 1999.

[21] Petersen, I.R., Ugrinovskii, V.A. and Savkin, A.V., “Robust Control Design Using H-∞ Methods,” Springer-Verlag London, 2000

[22] Yoon M.G., Ugrinovskii, V.A. and Petersen, I.R. “On the worst-case disturbance of minimax optimal control,” Journal of Automatica Volume 41, Issue 5, May 2005, Pages 847–855

[23] Rehman, O.U. and Petersen, I.R., “Robust Dynamic State Feedback Guaranteed Cost Control of Nonlinear Systems using Copies of Plant Nonlinearities,” arXiv:1404.3808, April 2014

[24] Iftar, A. and Ozguner, U., “Minimax optimal control for uncertain systems,” Proceedings of the American Control Conference, pp. 2338-2342, Baltimore, MD, U.S.A., June 1994.

[25] Smith, B.A., “Probabilistic Parameter Uncertainty Analysis of Single Input Single Output Control Systems”, NASA/TM-2005-213280

[26] Guillemin, V. and Pollack, A., “Differential Topology,” Prentice-Hall, 1974

[27] Green, M. and Smith, M.C., “Continuity properties of LQG controllers,” Systems & Control Letters**,** Volume 26, Issue 1, 8 September 1995, Pages 33–39

[28] Lenarcic, J., Bajd, T. and Stanišić,. M.M, “Robot Mechanisms,” Springer, 2013

[29] Gelfand, I. M.; Kapranov, M. M.; Zelevinsky, A. V. “Discriminants, resultants and multidimensional determinants.” Birkh äuser. p. 1. 1994.

[30] Haddad, W.M., and Tadmor, G., “Reduced order LQG controllers for linear time varying plants,” Decision and Control, 1992., Proceedings of the 31st IEEE Conference on Decision and Control

[31] Stanford University, ee363, lecture notes, “Linear quadratic Lyapunov theory”, http://stanford.edu/class/ee363/lectures/lq-lyap.pdf

[32] Hwang, C.L. and Masud, A.S.M., “Multiple objective decision making, methods and applications: a state-of-the-art survey.” Springer-Verlag. Retrieved 29 May 2012.

[33] Free, R.C., “21st Century Economics: A Reference Handbook,” SAGE, 2010, p. 897

[34] Michiel, H., ed., "Newton method", Encyclopedia of Mathematics, Springer, 2001

[35] Sun, W., Yua, Y.X., “Optimization Theory and Methods: Nonlinear Programming,” Springer, 2010

[36] Groves et al., “Survey Methodology,” Wiley: New York.

[37] Hamilton Institute. “The Binomial Distribution,” http://www.hamilton.ie/ollie/EE304/Binom.pdf

[38] lehigh University, IE417 Lecture 22 http://coral.ie.lehigh.edu/~ted/files/ie417/lectures/Lecture22.pdf

[39] Perez-Lombard, L., Ortiz, J., and Pout, C., “A Review on Buildings Energy Consumption Information,” Energy and Buildings, Vol. 40, Issue 3, 2008.

[40] U.S. Energy Information Administration, “How much energy is consumed in residential and commercial buildings in the United States?” http://www.eia.gov/tools/faqs/faq.cfm?id=86&t=1.

[41] LEED Technical and Scientific Advisory Committee, “The treatment by LEED of the environmental impact of HVAC refrigerants,” 2004.

[42] ASHRAE Standards & Guidelines, https://www.ashrae.org/standards-research--technology/standards--guidelines

[43] Niagara, Applications In C-Store and Petroleum Retailing, http://www.tridium.com/cs/markets_/_applications/cstores_and_petroleum_retailing

[44] Pierre, P., “A to Z of Thermodynamics.” Oxford University Press, 1998

[45] KMC Controls, “Zone control with Variable Air Volume controls (VAV),” http://www.kmccontrols.com/products/Zone_control_with_Variable_Air_Volume_Controls.aspx

[46] Underwood, C. P., “HVAC CONTROL SYSTEMS MODELLING, ANALYSIS AND DESIGN”, E&FN SPON, Routledge, 1999.

[47] Nesler, C. G. and Stoecker, W. F., “Selecting the proportional and integral constants in the direct digital control of discharge air temperature,” ASHRE Transactions (Part 2B), 90, 834-845, 1984.

[48] Hittle, D. C. and Haines, R. W., “Dynamic response and tuning.” ASHRAE Journal, pages 40-43, September 1997.

[49] Yahiaoui, A., Hensen, J. L. M., Soethout, L. and Paassen, A. H. C., "Design of model based LQG control for integrated building systems," Proceedings of the 8th IASTED Int. Conf. on Control and Applications, 24-26 May, pp. 6. Montreal: International Association of Science and Technology for Development.

[50] Zaheer-uddin, M., "Optimal control of a single zone environmental space," Building and Environment, 27 (1), p 93-103

[51] Kasahara, M., Matsuba, T., Hashimoto, Y., Murasawa, I., Kimbara, A., Kamimura, K. and Kurosu, S., "Optimal Preview Control for HVAC System," ASHRAE Transactions, 1998, Vol 104, pt. 1A, San Francisco

[52] Park, C. and Alexander J. D., "An adaptive controller for heating and cooling systems: Modeling, implementation, and testing." In American Society of Mechanical Engineers (Paper), page 8p, New York, NY, 1982.

[53] Wang, S. and Jin, X., "Model-based optimal control of VAV air-conditioning system using genetic algorithm," Building and Environment, Volume 35, Issue 6, 1 August 2000, Pages 471–487.

[54] Sheikholeslamia, N., Shahmirzadib, D., Semsarc, E., Lucasd, C. and Yazdanpanahd, M. J., "Applying brain emotional learning algorithm for multivariable control of HVAC systems," Journal of Intelligent & Fuzzy Systems 17 (2006) 35–46.

[55] Soyguder, S., Karakose, M. and Alli, H., "Design and simulation of self-tuning PID-type fuzzy adaptive control for an expert HVAC system," Expert Systems with Applications, Volume 36, Issue 3, Part 1, April 2009, Pages 4566–4573.

[56] Ferreirab, P.M., Ruanoa, A.E., Silva, S. and Conceicao, E. Z. E., "Neural networks based predictive control for thermal comfort and energy savings in public buildings," Energy and Buildings, Volume 55, December 2012, Pages 238–251.

[57] Saboksayr, S. H., Patel, R. V., and Zaheer-Uddin, M., "Energy-efficient operation of HVAC systems using neural network based decentralized controllers." In Proceedings of the American Controls Conference, number pt. 6, pages 4321-4325, 1995. Jun. 21-23.

[58] Qi, Q. "Multiple Control of Air Temperature and Humidity in a Space Served by a Direct Expansion (DX) Air Condition (A/C) system", Phd Thesis

[59] Chen W. "Modeling and Control of a Direct Expansion (DX) Variable-air-volume (VAV) Air Conditioning (A/C) System," Phd Thesis

[60] Wang C.C., Lin Y. T., Lee C. J. "Heat and momentum transfer for compact louvered fin-and-tube heat exchangers in wet conditions." Int. J. Heat and Mass Transfer, Vol. 43, No 18, pp. 3443-3452 (2000)

[61] MSU ME851 Lecture 14,
http://www.egr.msu.edu/classes/me851/jchoi/lecture/Lect_14.pdf

[62] ANSI/ASHRAE Standard 62.2-2013: Ventilation and Acceptable Indoor Air Quality in Low-Rise Residential Buildings". Atlanta, GA: American Society of Heating, Refrigerating and Air-Conditioning Engineers. 2013.

[63] Verma, P., Bullard, C.W. and Hrnjak, P.S., "DESIGN STRATEGIES FOR REDUCING PERFORMANCE DEGRADATION DUE TO FROSTING OF DISPLAY CASE HEAT EXCHANGERS," Proceedings of IIR Conference New Technologies in Commercial Refrigeration, Urbana, 227-237, 11 pp.

[64] "Effect of CGOn Aircraft Efficiency", http://www.westwingsinc.com/cgeffect.htm

[65] Gong, H.J., Zhen Z.Y., Li, X., Jiang, J., Wang, X.H., "Automatic Flight Control System Design of Level Change Mode for a Large Aircraft," International Journal of Advanced Robotic Systems, 2013

[66] Aircraft Pitch: System Modeling
http://ctms.engin.umich.edu/CTMS/index.php?example=AircraftPitch&section=System Modeling

[67] Ide, K. and Sornette, D., "Oscillatory Finite-Time Singularities in Finance," Population and Rupture, Physica A.