LPV CONTROLLER SYNTHESIS FOR THE ROTARY INVERTED PENDULUM

by

Mustafa Emre Kırcalar

B.S., Electrical Engineering, Yıldız Technical University, 2007

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Systems & Control Engineering

Boğaziçi University

2011

ACKNOWLEDGEMENTS

First of all, I would like to thank and express my gratitude to my thesis supervisor, Prof. Eşref Eşkinat for his supports, time and patience in preparation of this thesis. Also, I am thankful to Prof. İbrahim Emre Köse for his contributions.

Special thanks to Prof. Carsten W. Scherer who provided IQC Synthesis Toolbox.

I would like to express my appreciation to my thesis committee: Prof. İbrahim Emre Köse and Assoc. Prof. Yağmur Denizhan for their participation.

Finally, my family deserves special acknowledgement for their patience and endless supports.

ABSTRACT

LPV CONTROLLER SYNTHESIS FOR THE ROTARY INVERTED PENDULUM

Linear parameter varying (LPV) controller design for nonlinear systems is mostly considered as challenging. Moreover, the available methods and tools suggested in the literature are scarce and not well tested on physical systems. In this thesis, the design of the LPV controller for the Rotary Inverted Pendulum (RIP) is presented. Starting from a linear fractional transformation structure (LFT) of the LPV plant, which is obtained by nonlinear dynamical equations of the RIP, controller synthesis Linear Matrix Inequality (LMI) conditions are developed. By involving LMIs, based on LPV synthesis theory, the controller design procedure is performed. Additionally, the designed LPV controller performance is evaluated by the provided simulation and experimental results for different conditions.

ÖZET

DÖNER TERS SARKAÇ İÇİN DPD DENETİMCİ SENTEZİ

Doğrusal olmayan sistemler için Doğrusal Parametre Değişkenli (DPD) denetimci tasarımı çoğunlukla sınayıcı olarak kabul edilir. Ayrıca, litaratürde önerilen mevcut yöntem ve araçlar kolaylıkla bulunamamakta ve fiziksel sistemler üzerinde iyi bir şekilde test edilememektedir. Bu tezde, Döner Ters Sarkaç (DTS) için DPD denetimci tasarımı takdim edilmiştir. DTS'nin doğrusal olmayan dinamik denklemlerinden elde edilmiş olan DPD sistemin Dorusal Kesirli Dönüşüm (DKD) yapısından başlayarak, denetimci sentezi için gereken Doğrusal Matris Eşitsizliği (DME) şartları oluşturuldu. Denetimci tasarım prosedürü DME'ni içeren DPD sentezi teorisi temel alınarak gerçekleştirildi. Ayrıca, tasarlanan DPD denetimci performansı simülasyon ve deneysel sonuçlar ile farklı durumlar için değerlendirildi.

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF SYMBOLS

α	Pendulum deflection angle
B_{eq}	Viscous damping coefficient
η_g	Gearbox efficiency
η_m	DC motor efficiency
g	Gravity
h	Distance of pendulum center of mass from ground
J_{eq}	Moment of inertia at the load
J_m	DC motor moment of inertia
K_g	Gear ratio
K_m	Back - emf constant
K_t	DC motor torque constant
L	Length of pendulum
m	Mass of pendulum
r	Rotating arm length
R_m	Armature resistance
θ	Rotating arm angle

LIST OF ACRONYMS/ABBREVIATIONS

LFR	Linear Fractional Representation
LFT	Linear Fractional Transformation
LMI	Linear Matrix Inequality
LPV	Linear Parameter Varying
LQR	Linear Quadratic Regulator
RIP	Rotary Inverted Pendulum
RLC	Resistance Inductance Capacitance

1. INTRODUCTION

1.1. Rotary Inverted Pendulum System

Control of Inverted Pendulum (IP) is one of the most classical and interesting application of control engineering. Many different types of IP systems, mainly including linear, rotary, double and triple versions, have been derived and several control algorithms have been implementing on these systems [2], [4], [9].

In aerospace industry the inverted pendulum model can be used as a benchmark tool for several algorithms in a way that it represents yaw and pitch of a rocket in flight [1].

Rotary Inverted Pendulum (RIP) is one of the member of IP family which was first developed at Tokyo Institute of Technology by Dr. Katsuhisa Furuta under the name of "Furuta Pendulum". The RIP employed in this thesis is produced by Quanser Inc. as a laboratory experiment tool [1].

Figure 1.1. Overview of RIP developed by Quanser Inc.

As shown in the Figure 1.1 [1], RIP system consists of three main components which

are the pendulum, the pendulum arm and the DC motor. Pendulum and arm are connected with a ball bearing and they are perpendicular to each other. The arm is fixed to the rotor of the DC motor. While the pendulum can freely rotates in the vertical plane, the pendulum arm rotate in the horizontal plane in conjunction with the rotor angle of the DC motor. DC motor has permanent magnets and no brushes. It is directly driven by applying DC voltage between 0 and 15 volts.

The nonlinear characteristics of the RIP makes it one of the challenging control engineering problems. Most commonly, stabilization problems in robotics can be given as a real life application of the RIP [18].

1.1.1. RIP Components

RIP consists of the following mechanical and electrical units which are shown in the next figure, Figure 1.2 [1] :

1. Base unit (Including DC motor, motor encoder and gears)
2. Thumbscrews
3. Coupling arm
4. Pendulum sensor (Pendulum encoder)
5. Shaft (Connects pendulum and encoder)
6. Fixture
7. Pendulum

The RIP has gears, thumbscrews, coupling arm, shaft, fixture and pendulum as mechanical units. The inverted pendulum plant without actuation and sensing units can be formed from the mechanical components listed previously. However, in order to design a proper feedback control system, the actuation and sensing units must be incorporated to the plant, which operate with electricity. Hence, the electrical components play a key role in RIP. The RIP basically comprises 4 main electrical components which are:

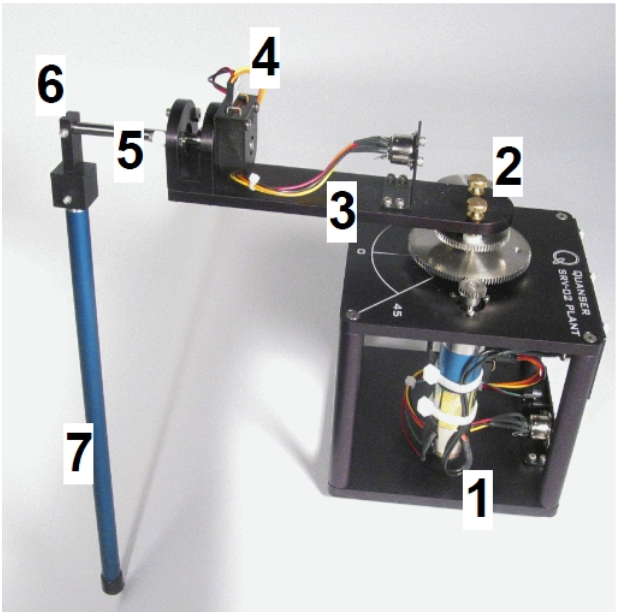

Figure 1.2. RIP components.

- Sensing and Actuating Units (Encoders and DC motor)
- Control Signal Interface Card (Transfers and smooths (digital to analog converter) the control signal sent by computer to amplifier)
- Amplifier (Amplifies the control signal and sends to the DC motor)
- PC and the control algorithm

A general control system block diagram structure is shown in Figure 1.3 [2]. In this diagram, measurement and actuator block represents the encoders and the DC motor; DAC block represents the control signal interface card; the digital controller block is also represents the PC and control algorithm.

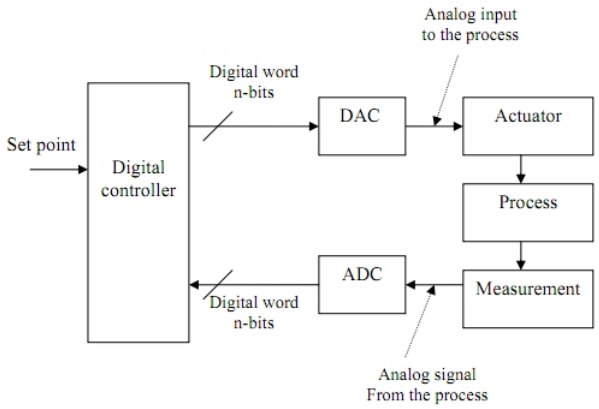

Figure 1.3. Digital control system block diagram.

The algorithm utilized in this thesis is based on Linear Matrix Inequalities (LMI). LMIs are a convex optimization method for the controller synthesis. LMIs and LMI techniques have emerged as a powerful design tools for system and control engineering. The following two elements make LMI techniques advantageous:

- A variety of design specifications and constraints can be expressed through LMIs.
- A problem can be solved exactly by efficient convex optimization algorithms with solvers [19], once formulated in terms of LMIs.

A brief mathematical introduction to LMIs is presented in Chapter 4; "ANALYSIS AND DESIGN: THEORETICAL BACKGROUND"

1.2. Problem Description

1.2.1. Problem Statement

In this thesis, linear parameter varying (LPV) controller design and synthesis are made for the RIP. The nonlinear parameters (some states of the state space RIP model) are supposed as time varying parameters in the mathematical model of the RIP. By this approach, the uncertainty-like time varying parameters are taken out from the nonlinear model to form a linear fractional representation (LFR) structure of the model. Since the uncontrolled parameter-dependent system stability can be expressed through LMIs [6], the stabilizing controllers are synthesized subject to some stability conditions for the closed loop system with control inputs and measured outputs [3], [8]. Eventually, the resulting feasible LMI conditions synthesize the controller.

1.2.2. Related Work

Both RIP and IP has been implemented with many different control algorithms so far. Linear and nonlinear controllers are designed for the stabilization of the RIP. Some of these papers are given in this section.

One of the papers that presents the design of a linear controller is "Gain-Scheduling Control of a Rotary Inverted Pendulum by Weight Optimization and H_∞ Loop Shaping Procedure [4]". In this paper, a gain-scheduling control method proposed by Hyde and Glover is applied. H_∞ loop shaping procedure is adopted as a controller synthesis method at each operating point. Moreover, the optimum weights required for the controller synthesis are found by the formulation of the weight optimization problem as a generalized eigenvalue minimization problem.

"Robust Predictive Control of the Furuta Pendulum [2]", is another paper that deals with the control of the LPV model of the Furuta Pendulum. Based on this model, a balancing gain-scheduling controller is designed using robust predictive control techniques. Also, the speed of the rotating arm is chosen as gain scheduling variable.

Methods based on polytope geometric techniques are used for stabilization of the rotary inverted pendulum are given in "Research on Control of Rotary Inverted Pendulum via Polytope Techniques Rotary Inverted Pendulum [9]". Polytope techniques are demonstrated for the controller synthesis in the paper.

A nonlinear controller design for the rotary inverted pendulum system using the input-state linearization method is provided in "Input-State Linearization of a Rotary Inverted Pendulum [21]". The RIP is linearized, and the conditions necessary for the system to be linearizable are discussed in the paper. Once the system is linearized, the linear servo controllers are designed based on the pole-placement method.

Additionally, mathematical modeling of IP has also been published as journal and conference papers. "Modeling of the Rotary Inverted Pendulum System [10]" titled paper models the RIP with Bond Graph method which consists of subsystems linked together by lines representing the bonds.

Another paper for the modeling of LPV systems is "Identification of Linear Parameter Varying Models [11]". In this paper the identification of a certain class of discrete-time nonlinear systems is handled. It is provided that the identification prob-

lem can be reduced to a linear regression, and provide compact formulae for the corresponding least mean square and recursive least-squares algorithms.

In addition to the articles related to the controller design and system modeling, the use of LMIs in control is presented in "Linear Matrix Inequalities in System and Control Theory [6]" comprehensively. This book solve problems from system and control theory using convex optimization methods. The book provides numerical solution methods for most LMIs and gives analytical solutions for a few special cases.

The organization of the thesis proceeds as follows: Mathematical modeling and the derivation of the LPV model are presented in Chapter 2. In Chapter 3, linear fractional representation (LFR) and linear fractional transformation (LFT) of uncertain systems are described. LPV controller synthesis discussed in Chapter 4. Implementation of the controller in MATLAB environment and simulation set-up are given in Chapter 5. Simulation results for different configurations are found in Chapter 6. In the final chapter, evaluation of the results and future studies are mentioned.

2. MATHEMATICAL MODELING OF THE RIP

In this chapter both nonlinear and linearized mathematical models of the RIP are derived. In mathematical modeling, a system model is represented by a set of ordinary differential equations in terms of state variables and a set of algebraic equations that relate the state variable to other system variables.

There are two main coordinates α and θ by which the displacement of the pendulum and the arm angles are defined respectively. These coordinates are shown in the figures 2.1 and 2.2 in detail and the angles are illustrated in a free body diagram and a simplified physical model drawings. In Section 2.3 the derived nonlinear linearized models are compared to the physical RIP to verify and validate the model accuracy.

2.1. Nonlinear Model of the RIP

Table 2.1. Symbols and descriptions.

Symbol	**Description**
L	Length to pendulum's center of mass
h	Distance of pendulum center of mass from ground
m	Mass of pendulum
J_{CoM}	Pendulum inertia about its center of mass
r	Rotating arm length
V_x	Velocity of pendulum center of mass in the x-axis
V_y	Velocity of pendulum center of mass in the y-axis
θ	Rotating arm angle
α	Pendulum deflection
A	Fixing point of pendulum
B	Center of mass of pendulum (CoM)

The nomenclature used in the derivation of the equations are listed in Table 2.1. For complete list of symbols, refer to "LIST OF SYMBOLS" section.

Simplified physical model of the RIP is shown in Figure 2.1 [17] below in which the geometry of RIP is depicted. Taking into consideration of the directions of both rotating arm and pendulum in motion the pendulum is displaced with α while the arm rotates an angle of θ.

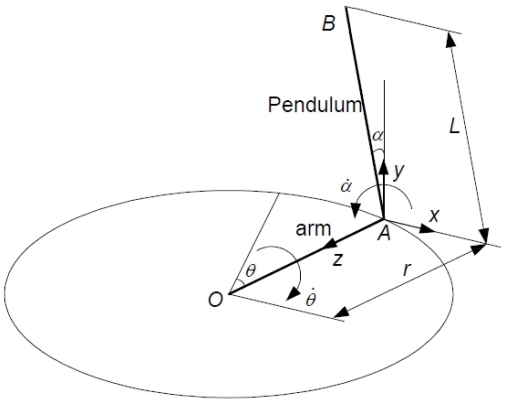

Figure 2.1. Simplified physical model of RIP.

Referring to Figure 2.1, the velocity of the pendulum at point B with respect to A has two components:

$$V_{Pendulum\,CoM} = -L\cos(\alpha)\dot{\alpha}\hat{x} - L\sin(\alpha)\dot{\alpha}\hat{y} \tag{2.1}$$

The rotating arm is also moving with the pendulum at a rate of $r\dot{\theta}$. Hence, the absolute velocity for the x and y velocity components of the pendulum mass at the point B can be expressed using equation (2.1) as

$$V_x = r\dot{\theta} - L\cos(\alpha)\dot{\alpha} \tag{2.2a}$$

$$V_y = -L\sin(\alpha)\dot{\alpha} \tag{2.2b}$$

With equations (2.2a) and (2.2b) the complete velocity of the pendulum is obtained. After this point, dynamical equation of motions of the RIP will be derived.

2.1.1. Derivation of the Dynamic Equations of the RIP

The dynamic equations of the system can be obtained through using Newtonian method. Figure 2.2 [17] illustrates free body diagram of both the rotating arm and the pendulum for which the dynamic equations are obtained.

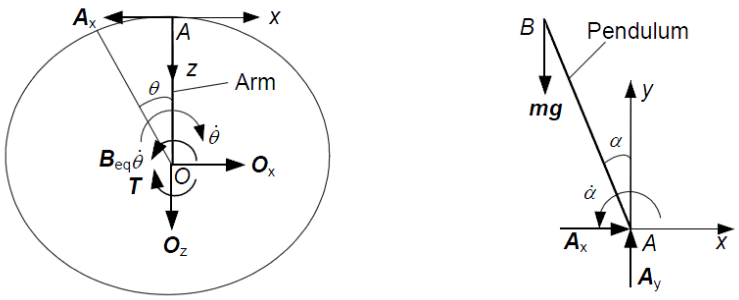

Figure 2.2. Free body diagram of the rotating arm and the pendulum.

The acceleration of the point B is obtained in X and Y directions by differentiating the equations (2.2a) and (2.2b) with respect to time as shown in the equations (2.3a) and (2.3b)

$$\dot{V_x} = r\ddot{\theta} + L\sin(\alpha)\dot{\alpha}^2 - L\cos(\alpha)\ddot{\alpha} \tag{2.3a}$$

$$\dot{V_y} = -L\cos(\alpha)\dot{\alpha}^2 - L\sin(\alpha)\ddot{\alpha} \tag{2.3b}$$

Applying Newton's Second Law to the pendulum in x direction at point A, we get

$$\sum F_x \;=\; m\dot{V_x} \;\Rightarrow\; mr\ddot{\theta} + mL\sin(\alpha)\dot{\alpha}^2 - mL\cos(\alpha)\ddot{\alpha} = F_{A_x} \tag{2.4}$$

And also in y direction at point A, we obtain

$$\sum F_y \;=\; m\dot{V_y} \;\Rightarrow\; mg - mL\cos(\alpha)\dot{\alpha}^2 - mL\sin(\alpha)\ddot{\alpha} = F_{A_y} \tag{2.5}$$

The moment of inertia of the pendulum about its center of mass is obtained by taking the distance between axis and rotation mass, R, as $2L$, since L is defined as to be half the pendulum length. Therefore the moment of inertia of the pendulum is

$$J_{CoM} = \frac{1}{12} M(R)^2 = \frac{1}{12} m(2L)^2 = \frac{1}{3} m(L)^2 \tag{2.6}$$

Utilizing Euler's Equation to the rotational motion of the pendulum at point B, we get

$$\begin{aligned} J_{CoM}\ddot{\alpha} = \sum M_B \Rightarrow \frac{1}{12} m(2L)^2\ddot{\alpha} &= F_{A_x} L\cos(\alpha) + F_{A_y} L\sin(\alpha) \\ \Rightarrow \frac{1}{3} m(L)^2\ddot{\alpha} &= F_{A_x} L\cos(\alpha) + F_{A_y} L\sin(\alpha) \end{aligned} \tag{2.7}$$

If we apply the same formula for the rotating arm about point O it yields

$$J_O\ddot{\theta} = \sum M_O \Rightarrow J_{eq}\ddot{\theta} = T - B_{eq}\dot{\theta} - F_{A_x} r \tag{2.8}$$

where B_{eq} is the viscous damping coefficient and T is the torque produced by the DC motor. The DC motor dynamics is provided in Subsection 2.1.2.

Taking equations (2.4) and (2.5) into equation (2.7), we get

$$\begin{aligned} \frac{1}{3} mL^2\ddot{\alpha} &= (mr\ddot{\theta} + mL\sin(\alpha)\dot{\alpha}^2 - mL\cos(\alpha)\ddot{\alpha})L\cos(\alpha) \\ &\quad +(mg - mL\cos(\alpha)\dot{\alpha}^2 - mL\sin(\alpha)\ddot{\alpha})L\sin(\alpha) \end{aligned} \tag{2.9a}$$

$$\begin{aligned} \Rightarrow \frac{1}{3} mL^2\ddot{\alpha} &= (mLr\cos(\alpha)\ddot{\theta} + mL^2\sin(\alpha)\cos(\alpha)\dot{\alpha}^2 - mL^2\cos^2(\alpha)\ddot{\alpha}) \\ &\quad +(mgL\sin(\alpha) - mL^2\sin(\alpha)\cos(\alpha)\dot{\alpha}^2 - mL^2\sin^2(\alpha)\ddot{\alpha}) \end{aligned} \tag{2.9b}$$

$$\Rightarrow \frac{4}{3} mL^2\ddot{\alpha} - mLr\cos(\alpha)\ddot{\theta} - mgL\sin(\alpha) = 0 \tag{2.9c}$$

If equation (2.4) is substituted into (2.8) it yields

$$J_{eq}\ddot{\theta} = T - B_{eq}\dot{\theta} - (mr\ddot{\theta} + mL\sin(\alpha)\dot{\alpha}^2 - mL\cos(\alpha)\ddot{\alpha})r \tag{2.10a}$$

$$J_{eq}\ddot{\theta} = T - B_{eq}\dot{\theta} - (mr^2\ddot{\theta} + rmL\sin(\alpha)\dot{\alpha}^2 - rmL\cos(\alpha)\ddot{\alpha}) \tag{2.10b}$$

$$T - B_{eq} = \dot{\theta}(mr^2 + J_{eq})\ddot{\theta} + rmL\sin(\alpha)\dot{\alpha}^2 - rmL\cos(\alpha)\ddot{\alpha} \tag{2.10c}$$

Finally, putting equations (2.9c) and (2.10c) together, the system's motion of equations would be obtained as follows

$$\frac{4}{3}mL^2\ddot{\alpha} - rmL\cos(\alpha)\ddot{\theta} - mgL\sin(\alpha) = 0 \tag{2.11a}$$

$$(mr^2 + J_{eq})\ddot{\theta} + rmL\sin(\alpha)\dot{\alpha}^2 - rmL\cos(\alpha)\ddot{\alpha} = T - B_{eq}\dot{\theta} \tag{2.11b}$$

2.1.2. The Complete Nonlinear and Linearized Models of the RIP

The mechanical system dynamic equations are obtained so far. However actuator dynamics, in our case the actuator is a dc motor, must be accurately taken into account in order to get a complete nonlinear system model. A simplified circuit diagram of a dc motor is demonstrated in the figure below

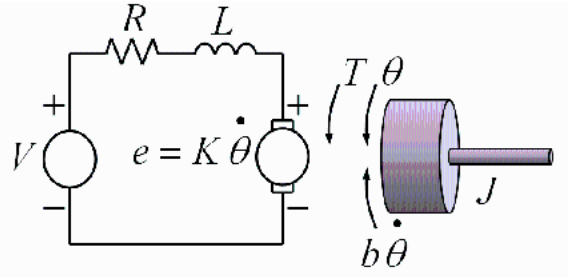

Figure 2.3. Simplified circuit diagram of DC motor.

The output torque applied by the dc motor to the load shaft is [1]

$$T_{output} = \eta_g \eta_m K_g K_t I_m \tag{2.12a}$$

$$= \eta_g \eta_m K_g K_t \frac{V_m - K_g K_t \dot{\theta}}{R_m} \tag{2.12b}$$

$$= \frac{\eta_g \eta_m K_g K_t}{R_m} V_m - \frac{\eta_g \eta_m K_g^2 K_t K_m}{R_m} \dot{\theta} \tag{2.12c}$$

$$= \frac{\eta_g \eta_m K_g K_t (V_m - K_g K_m)\dot{\theta}}{R_m} \tag{2.12d}$$

where η_g, η_m are the efficiency coefficients; K_g, K_m, K_t are some dc motor constants; V_m is the applied voltage; and R_m is the armature resistance. Thus, the complete nonlinear model of the system including actuator dynamics is obtained by taking equation (2.12d) into equation (2.11b) which are [1]

$$c\ddot{\alpha} - b\cos(\alpha)\ddot{\theta} - d\sin(\alpha) = 0 \tag{2.13a}$$

$$a\ddot{\theta} - b\cos(\alpha)\ddot{\alpha} + b\sin(\alpha)\dot{\alpha}^2 + U\dot{\theta} = \frac{\eta_g \eta_m K_g K_t}{R_m} V_m \tag{2.13b}$$

where

$$a = mr^2 + J_{eq} \qquad U = \frac{\eta_g \eta_m K_g^2 K_t K_m}{R_m} + B_{eq}$$

$$b = rmL \qquad W = \frac{\eta_g \eta_m K_g K_t}{R_m}$$

$$c = \frac{4}{3} mr^2$$

$$d = mgL$$

For the two accelerations $\ddot{\alpha}$ and $\ddot{\theta}$, solving equations (2.13) yield the nonlinear model below in equations (2.14).

$$\ddot{\theta} = \frac{1}{ac - b^2\cos^2(\alpha)}\left[bd\sin(\alpha)\cos(\alpha) - bc\sin(\alpha)\dot{\alpha}^2 - cU\dot{\theta} + cWV_m\right] \tag{2.14a}$$

$$\ddot{\alpha} = \frac{1}{ac - b^2\cos^2(\alpha)}\left[ad\sin(\alpha) - bU\cos(\alpha)\dot{\theta} - b^2\sin(\alpha)\cos(\alpha)\dot{\alpha}^2 + bW\cos(\alpha)V_m\right] \tag{2.14b}$$

Hence, the nonlinear state space representation of the RIP is

$$\dot{x} = A(x)x + B(x)u \tag{2.15a}$$

$$y = Cx + Du \tag{2.15b}$$

where $x \in \mathbb{R}^n$ is the state vector, $u \in \mathbb{R}^m$ is the control input, $y \in \mathbb{R}^p$ is the measured output; the system state and input vectors are now chosen as

$$x = \begin{pmatrix} \theta \\ \dot{\theta} \\ \alpha \\ \dot{\alpha} \end{pmatrix} \quad \text{and, } u = \begin{pmatrix} V_m \end{pmatrix} \text{ respectively.}$$

Also, $A(x), B(x), C$ and D are determined from equations (2.14) as follows

$$A(x) = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{cU}{ac-b^2\cos^2(\alpha)} & \left(\frac{bd\sin(\alpha)\cos(\alpha)}{ac-b^2\cos^2(\alpha)}\right)\frac{1}{\alpha} & -\frac{bc\sin(\alpha)\dot{\alpha}}{ac-b^2\cos^2(\alpha)} \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{bU\cos(\alpha)}{ac-b^2\cos^2(\alpha)} & \left(\frac{ad\sin(\alpha)}{ac-b^2\cos^2(\alpha)}\right)\frac{1}{\alpha} & -\frac{b^2\cos(\alpha)\sin(\alpha)\dot{\alpha}}{ac-b^2\cos^2(\alpha)} \end{pmatrix} \tag{2.16a}$$

$$B(x) = \begin{pmatrix} 0 \\ \frac{cW}{ac-b^2\cos^2(\alpha)} \\ 0 \\ \frac{bW\cos(\alpha)}{ac-b^2\cos^2(\alpha)} \end{pmatrix} \tag{2.16b}$$

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{2.16c}$$

$$D = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{2.16d}$$

Assuming $\alpha \approx 0$ and $\dot{\alpha} \approx 0$, the linearized model can be obtained by equations (2.13) as follows:

$$c\ddot{\alpha} - b\ddot{\theta} - d\alpha = 0 \tag{2.17a}$$

$$-b\ddot{\alpha} + a\ddot{\theta} + U\dot{\theta} = WV_m \tag{2.17b}$$

Again, solving equations (2.16) for the two accelerations $\ddot{\alpha}$ and $\ddot{\theta}$, yields the linearized model below.

$$\ddot{\theta} = \frac{1}{ac-b^2}\left(bd\alpha - cU\dot{\theta} + cWV_m\right) \tag{2.18a}$$

$$\ddot{\alpha} = \frac{1}{ac-b^2}\left(ad\alpha - bU\dot{\theta} + bWV_m\right) \tag{2.18b}$$

The linearized state space representation of the RIP model, which will be utilized to demonstrate the model accuracy, is obtained as follows:

$$\dot{x} = Ax + Bu \tag{2.19a}$$

$$y = Cx + Du \tag{2.19b}$$

where

$$A = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{cU}{ac-b^2} & \frac{bd}{ac-b^2} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{bU}{ac-b^2} & \frac{ad}{ac-b^2} & 0 \end{pmatrix} \tag{2.20a}$$

$$B = \begin{pmatrix} 0 \\ \frac{cW}{ac-b^2} \\ 0 \\ \frac{bW}{ac-b^2} \end{pmatrix} \tag{2.20b}$$

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{2.20c}$$

$$D = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{2.20d}$$

2.2. Derivation of Quasi-LPV RIP Model

Linear Parameter Varying (LPV) modeling is a modeling method that handles time varying parameters as constants in a nonlinear model. If the time varying parameters contain some of the system states, then this type of LPV modeling would be Quasi-LPV modeling. The plant description of an Quasi-LPV Model is:

$$\dot{x} = A(p)x + B(p)u \tag{2.21a}$$

$$y = C(p)x + D(p)u \tag{2.21b}$$

where p is the parameter vector and $p \in \mathbb{R}^l$. Emphasizing the definition again, if the parameter vector (scheduling vector) is a true exogenous signal, the system (2.21) is referred to as a LPV system and, if the parameter vector contains the states or the output the system (2.21) is called as a quasi-LPV system. LPV systems might be accepted as extension of linear time varying systems (LTV systems) when the parameter vector is determined.

In the RIP model the parameter or scheduling vector is defined as

$$p = \begin{pmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \end{pmatrix} := \begin{pmatrix} \cos(\alpha) \\ \sin(\alpha) \\ \frac{\sin(\alpha)}{\alpha} \\ \dot{\alpha} \end{pmatrix} \tag{2.22}$$

The system matrices $A(p), B(p), C, D$ can be also written as

$$A(p) = \begin{pmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{cU}{z} & \frac{bd\,p_1p_3}{z} & -\frac{bc\,p_2p_4}{z} \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{bU\,p_1}{z} & \frac{ad\,p_3}{z} & -\frac{b^2\,p_1p_2p_4}{z} \end{pmatrix} \tag{2.23a}$$

$$B(p) = \begin{pmatrix} 0 \\ \frac{cW}{z} \\ 0 \\ \frac{bW\,p_1}{z} \end{pmatrix} \tag{2.23b}$$

$$C = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \tag{2.23c}$$

$$D = \begin{pmatrix} 0 \\ 0 \end{pmatrix} \tag{2.23d}$$

where, parameter z is defined as $z := ac - b^2\cos^2(\alpha)$. Since $ac - b^2\cos^2(\alpha)$ denominator parameter causes some problems in controller design and it has no significant effect on the nonlinear model's accuracy, $\cos^2(\alpha)$ in the parameter is taken as its nominal value:"1", ($\cos^2(0)$). Therefore parameter z would be $z := ac - b^2$. The nonlinear system in equations (2.15) now becomes a parameter dependent system.

2.3. The RIP Model Verification

In this section, the question of how adequately the nonlinear model represents the physical system is answered. Both the linear and nonlinear models are simulated and their simulation outputs are compared. The simulation results show that how the linear and nonlinear models are distinguished [1]. After the simulation results, the nonlinear RIP model is compared to the physical RIP system.

The following figures illustrate the nonlinear and linearized Simulink models of the RIP.

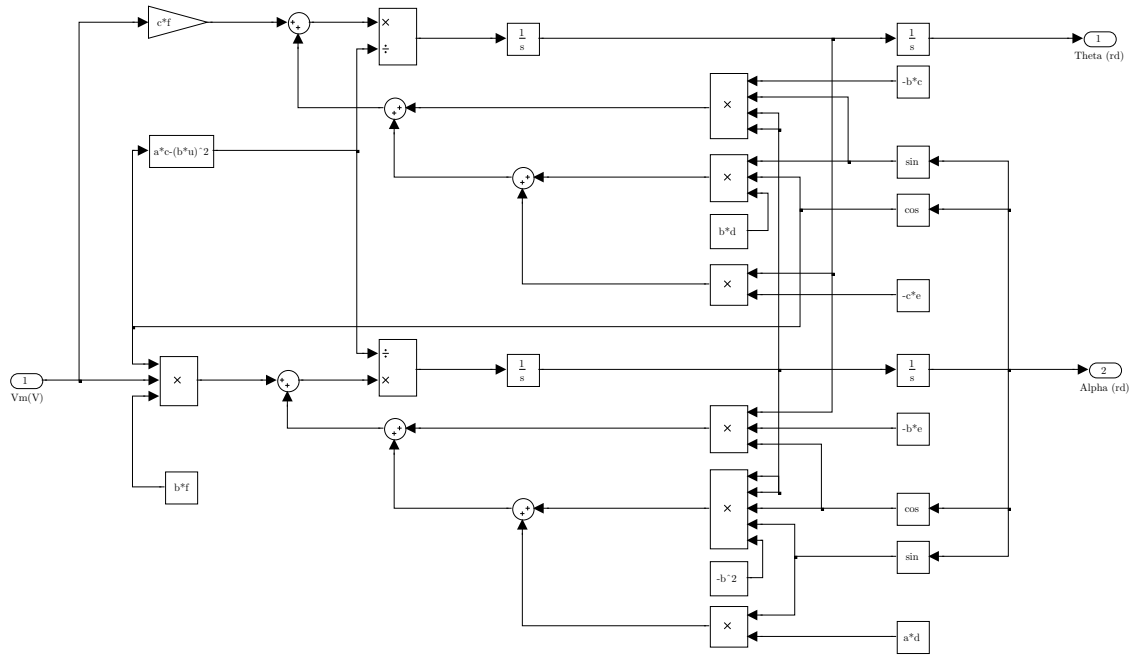

Figure 2.4. Nonlinear RIP simulink model.

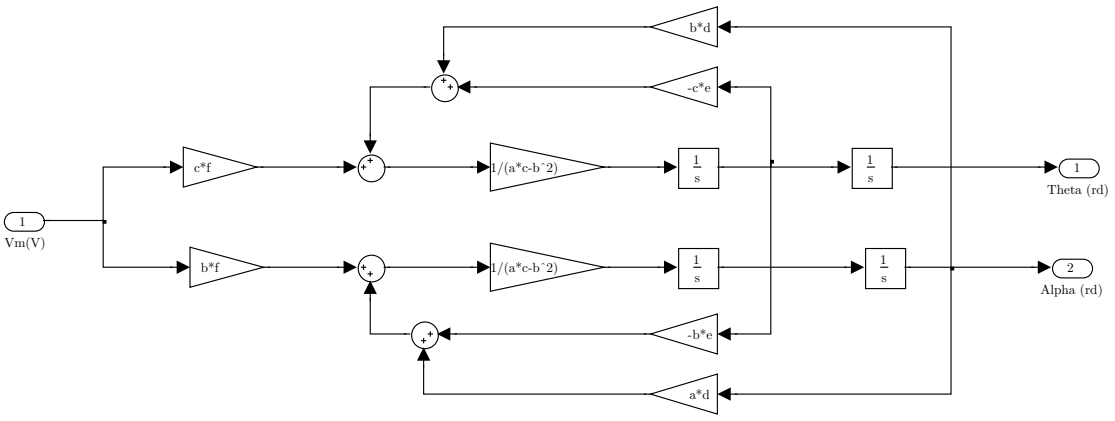

Figure 2.5. Linearized RIP simulink model.

The two models are compared (Figure 2.6) so that the initial pendulum angle, α, is set to a very small value (0.00001 rad). Therefore, the hanging pendulum is allowed to fall. As can be seen by the simulation results shown in Figure 2.7, the linear model

correctly depicts the motion of the pendulum until the first 1.4 seconds or it accurately describes the system for the first 15 degrees and then begins to diverge from the actual motion [1].

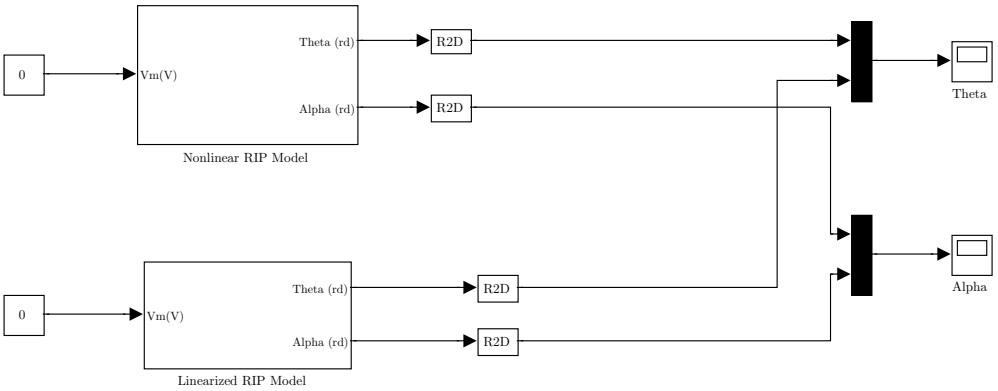

Figure 2.6. Nonlinear and linearized simulink model comparison.

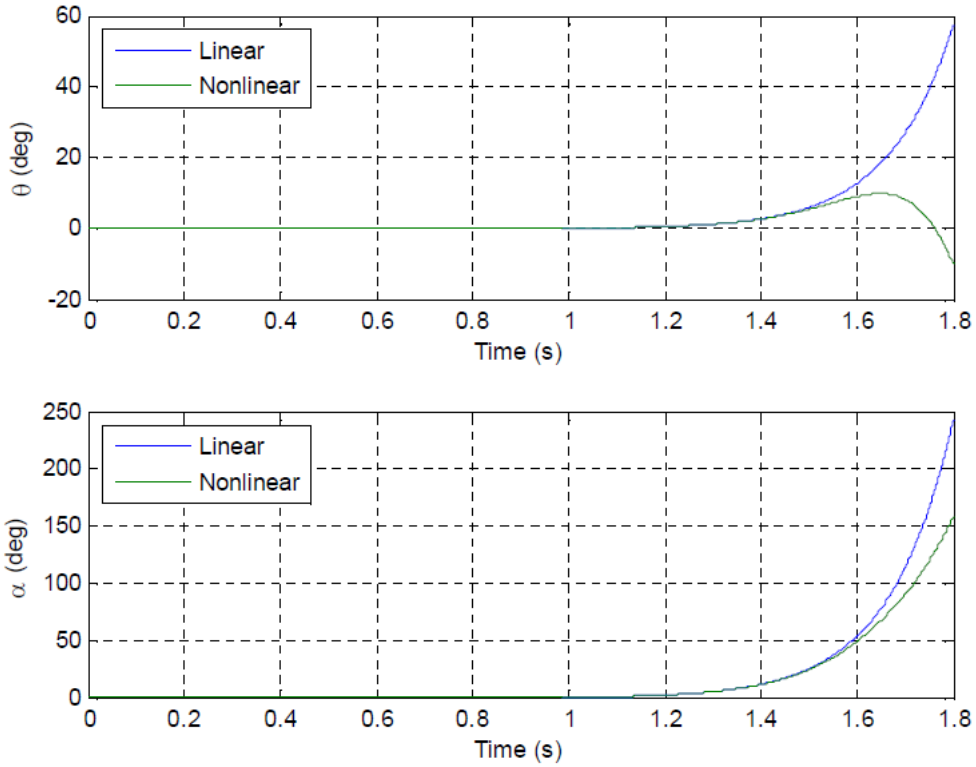

Figure 2.7. Pendulum angle α and arm angle θ values of linearized and nonlinear models.

To compare the models with the physical RIP system, the pendulum models are converted to hanging down position by taking α as $\pi+\alpha$. The nonlinear and linearized RIP system equations in (2.14) are now turn into

$$\ddot{\theta} = \frac{1}{ac - b^2\cos^2(\alpha)} \left[bd\sin(\alpha)\cos(\alpha) + bc\sin(\alpha)\dot{\alpha}^2 - cU\dot{\theta} + cWV_m \right] \tag{2.24a}$$

$$\ddot{\alpha} = \frac{1}{ac - b^2\cos^2(\alpha)} \left[-ad\sin(\alpha) + bU\cos(\alpha)\dot{\theta} - b^2\sin(\alpha)\cos(\alpha)\dot{\alpha}^2 - bW\cos(\alpha)V_m \right] \tag{2.24b}$$

And the equations of the linearized RIP system in (2.18) would be

$$\ddot{\theta} = \frac{1}{ac - b^2} \left(bd\alpha - cU\dot{\theta} + cWV_m \right) \tag{2.25a}$$

$$\ddot{\alpha} = \frac{1}{ac - b^2} \left(-ad\alpha + bU\dot{\theta} - bWV_m \right) \tag{2.25b}$$

Moreover, for the comparison of the two model with the physical system [17], below model in Figure 2.8 is denoted.

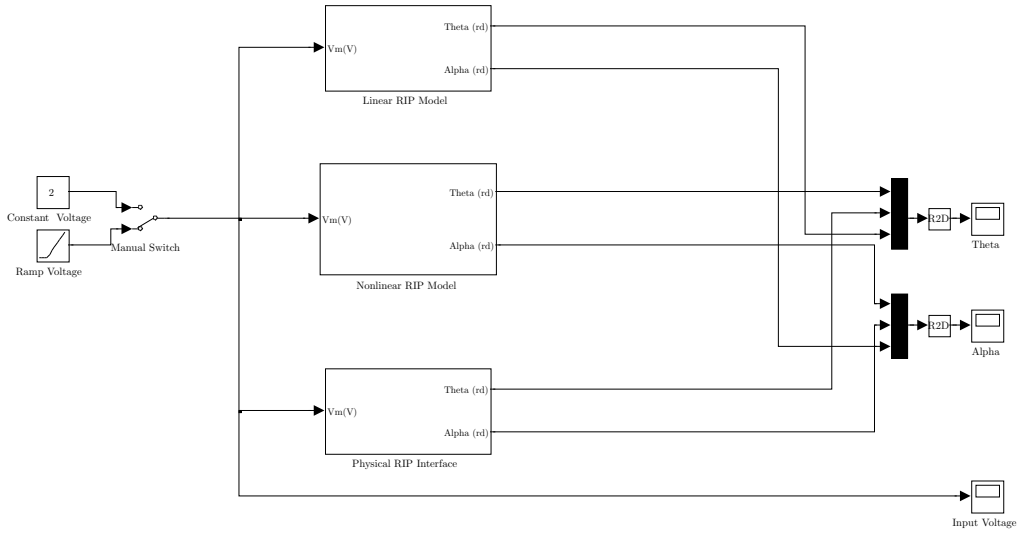

Figure 2.8. Comparison of three RIP models with simulink blocks.

After running the physical system with the created models in Figure 2.8, it yielded the following results for the ramp input voltage with slope 2 [17].

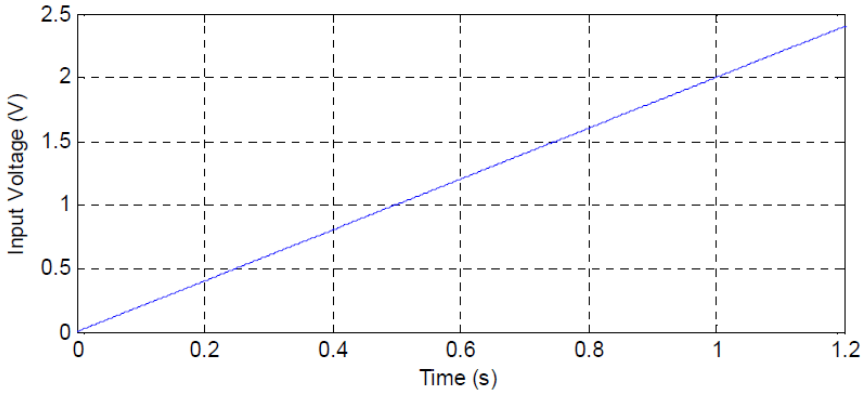

Figure 2.9. Input voltage.

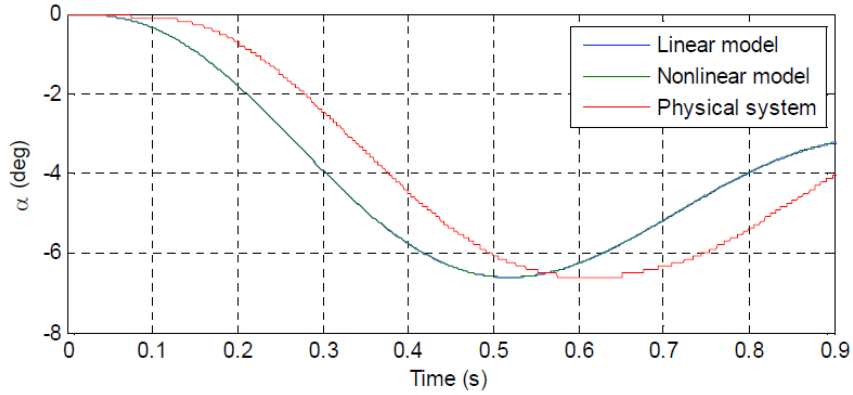

Figure 2.10. Pendulum angle α of the compared models.

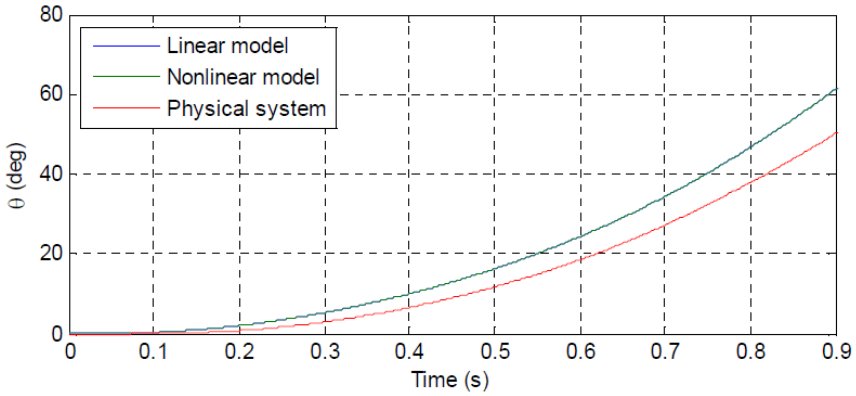

Figure 2.11. Arm angle θ of the compared models.

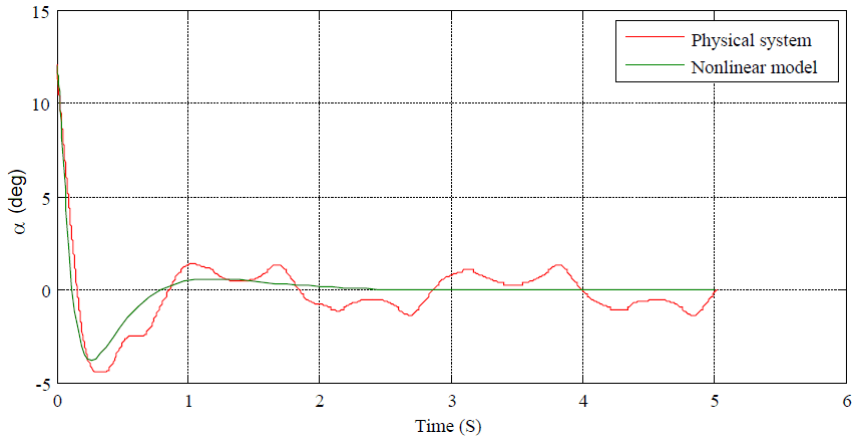

Figure 2.12. Pendulum angle α of the closed-loop system with LQR controller.

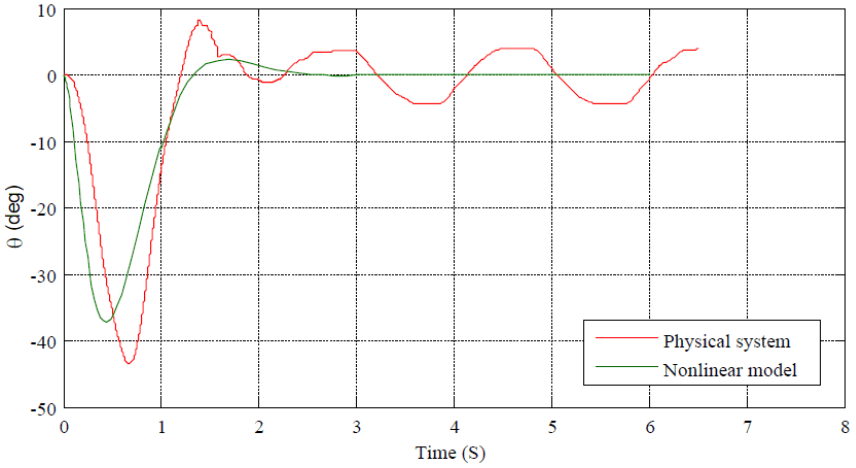

Figure 2.13. Arm angle θ of the closed-loop system with LQR controller.

It can be observed that the responses of the models are similar to the physical RIP system. They are almost in the same shape. However, since the model have some unmodeled dynamics such as friction, the physical RIP system has some delay which is illustrated in the Figures 2.10 and 2.11.

So far, the Quasi-LPV model has been obtained and verified with the physical RIP. It is observed that the Quasi-LPV RIP model (2.23) is in the form of ratios of polynomials which is called as "Linear Fractional Systems". This topic will be discussed further in the next chapter: "Linear Fractional Representation (LFR) and Uncertainty".

3. LINEAR FRACTIONAL REPRESENTATION AND UNCERTAINTY

There are many types of uncertainties in a real system with which a good control system should readily deal. To obtain satisfactory performance and stabilization, a controller should be designed to compensate for perturbations, model uncertainties and unmodeled dynamics. Some of these uncertainties may be known, partially known or completely unknown.

Most real physical systems do not preserve their parameters for a certain time interval. For example, resistivity of a circuit changes under different environmental conditions (temperature or pressure) or mass of an aircraft cannot be considered as constant due to the change in passenger or fuel loads. At this point, LPV systems would be a promising approach when modeling a system with known or partially known uncertainties.

3.1. Linear Fractional Transformation (LFT)

Consider the uncertain autonomous system below [13]

$$\dot{x} = G(p)x \tag{3.1}$$

where $p = (p_1, \ldots, p_l)^T \in \Delta$ is l-dimensional parameter vector which is included in a real matrix-valued function $G(p)$ and $x(t) \in \mathbb{R}^n$ is the state vector.

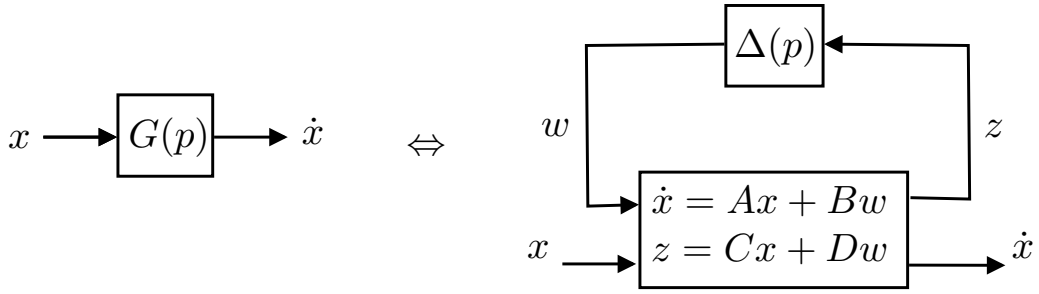

Figure 3.1. Linear fractional representation of (3.1).

Writing equation (3.1) into canonical LFR form [13]:

$$\begin{pmatrix} \dot{x} \\ z \end{pmatrix} = \overbrace{\begin{pmatrix} A & B \\ C & D \end{pmatrix}}^{\mathrm{H}} \begin{pmatrix} x \\ w \end{pmatrix}, \quad w = \Delta(p)z \tag{3.2}$$

This from, changing the equation(3.1) to equation (3.2) will be referred to as Linear Fractional Transformation where Δ linearly depends on p. Figure 3.1 depicts the block diagrams of the equations.

Definition 3.1.1. *A Linear Fractional Representation of $G(p)$ is a pair $(H, \Delta(p))$ where*

$$H = \begin{pmatrix} A & B \\ C & D \end{pmatrix}$$

is a constant partitioned matrix and Δ is a linear function of p such that $\forall p$ for which $I - D\Delta(p)$ is invertible and for all (η, ξ) there holds $\xi = G(p)\eta$ if and only if there exist vectors w and z such that:

$$\begin{pmatrix} \xi \\ z \end{pmatrix} = \overbrace{\begin{pmatrix} A & B \\ C & D \end{pmatrix}}^{\mathrm{H}} \begin{pmatrix} \eta \\ w \end{pmatrix}, \quad w = \Delta(p)z \ \textit{[13].} \tag{3.3}$$

A rational parameter dependent uncertain system is composed of two interconnected blocks; nominal Linear Time Invariant (LTI) system and uncertainties by which an LFT representation of the system is formed. The nominal LTI part is obtained through taking the varying parameters or uncertainties out. Consequently, putting these parameters into a Δ block yields the uncertainty block structure. Figure 3.2 shows a non-autonomous state space representation of an uncertain plant [3].

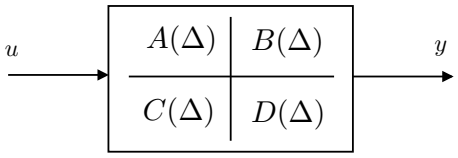

Figure 3.2. State-space representation of an uncertain plant.

If the parameter variation is pulled out from the non-autonomous system in Figure 3.2, the LFT representation is obtained as follows:

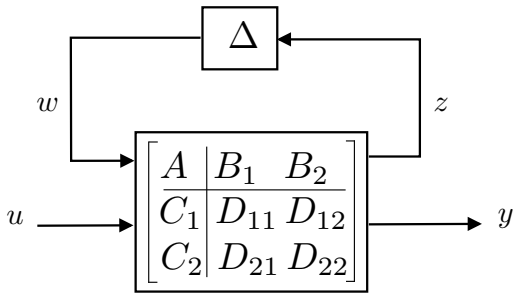

Figure 3.3. LFT representation of an uncertain plant.

where A, B_1, B_2, C_1, C_2, $D_{11}, D_{12}, D_{21}, D_{22}$ are defined in the equations below.

$$\dot{x} = Ax + B_1 w + B_2 u \tag{3.4a}$$

$$z = C_1 x + D_{11} w + D_{12} u \tag{3.4b}$$

$$y = C_2 x + D_{21} w + D_{22} u \tag{3.4c}$$

The relation between w and z is given as

$$w = \Delta(p) z \tag{3.5}$$

The uncertainty or scheduling matrix $\Delta(p)$ is assumed to be an affine function of the real scalar time-varying parameters $p = (p_1, p_2 \ldots, p_t) \in \mathbb{R}^t$ which means that for some fixed matrices $N_1, N_2 \ldots, N_t$; $\Delta(p) = p_1 N_1 + p_2 N_2 + \cdots + p_t N_t$. Moreover, the matrices $N_1,\ N_2 \ldots, N_t$ are not required to be square.

In a real vector space V with a set of points X, the minimal convex set containing X is defined as a *convex hull*. The convex hull Π, in which the parameter vector p varies, is given as $\Pi = co\{p^1, p^2, \ldots, p^i\}$, where $p^j = p_1^1, p_2^2, \ldots,\ p_t^j$ for $j = 1, 2, \ldots, i$; and zero is presumably contained in the convex hull Π.[14]

When the state-space matrices have rational dependence on the parameters p_1, p_2, $\ldots, p_t$ then the LFT formulation will come out a scheduling or parameter variation matrix Δ of the following special structure:

$$\Delta = \begin{bmatrix} p_1 I_{n_1} & 0 & \ldots & 0 \\ 0 & p_2 I_{n_2} & \ldots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \ldots & p_t I_{n_t} \end{bmatrix} \tag{3.6}$$

In this structure, the indices $n_1, n_2, \ldots, n_t$ denote the repetitiveness of the corresponding parameter variations $p_1, p_2, \ldots, p_t$ in the uncertain plant. These variations are bounded with finite numbers that could be normalized to some desired bounds.

The LFT structure can be obtained in two ways, both of them are similar.

- Upper LFT Form
- Lower LFT Form

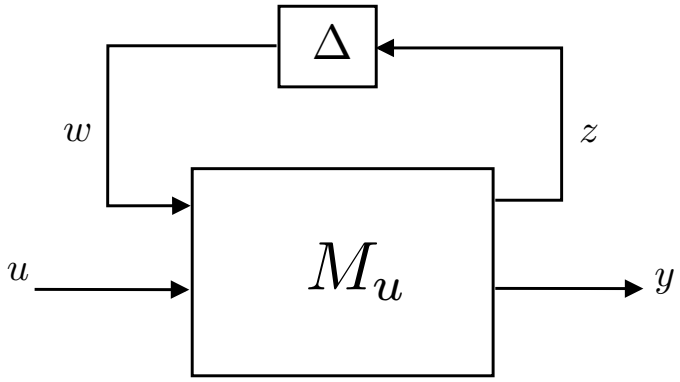

Figure 3.4. Upper LFT structure.

Definition 3.1.2. *In an $M - \Delta$ structure as shown in Figure 3.4 above, an upper LFT: $F_u(M_u, \Delta)$ is form that transfers u to y after closing the loop Δ.*

$$F_u(M_u, \Delta) = M_{21_u}\Delta(I - M_{11_u}\Delta)^{-1}M_{12_u} + M_{22_u} \tag{3.7}$$

where

$$M_u := \begin{bmatrix} M_{11_u} & M_{12_u} \\ M_{21_u} & M_{22_u} \end{bmatrix} \text{ [15].}$$

Definition 3.1.3. *In an $M - \Delta$ structure as shown in Figure 3.5 below, a lower LFT: $F_l(M_l, \Delta)$ is a form that transfers u to y after closing the loop Δ.*

$$F_l(M_l, \Delta) = M_{12_l}\Delta(I - M_{22_l}\Delta)^{-1}M_{21_l} + M_{11_l} \tag{3.8}$$

where

$$M_l := \begin{bmatrix} M_{11_l} & M_{12_l} \\ M_{21_l} & M_{22_l} \end{bmatrix} \text{ [15].}$$

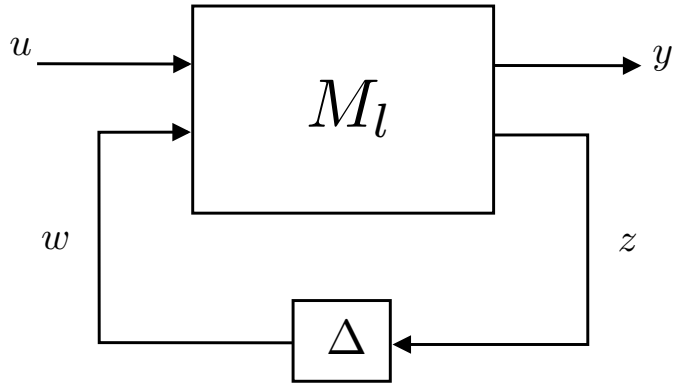

Figure 3.5. Lower LFT structure.

The matrix $(I - M_{11_u}\Delta)^{-1}$ in equation (3.7) cannot be inverted for all values of Δ unless $M_{11_u} = 0$. It is a requirement that invertibility is feasible for all corresponding values of Δ. For instance if the parameter variations are normalized between -1 and +1, it is expected to be invertible in the unit ball.

More comprehensively, the LFT structure can be denoted in a simple example. In this example, a simple series RLC circuit is given. Resistance "R" and capacitance "C" constants are the varying parameters of the circuit. Beginning with the mathematical model of the circuit that depends on the parameters including "R" and "C", the varying parameters are taken out from the system and put into a scheduling block. Therefore, the nominal system without considering the parameter variations is derived. At the end of the example, the obtained Δ block is connected to the nominal system to form an LFT structure.

Example 3.1.1. In the following figure a series RLC circuit with varying parameters resistance "R" and capacitance "C" is given. These parameters are varying between

$$R = [\underline{R}, \overline{R}], \tag{3.9a}$$

$$C = [\underline{C}, \overline{C}]. \tag{3.9b}$$

The mesh equation of the system is

$$L\ddot{q} + R\dot{q} + \frac{1}{C}\,q = V \tag{3.10}$$

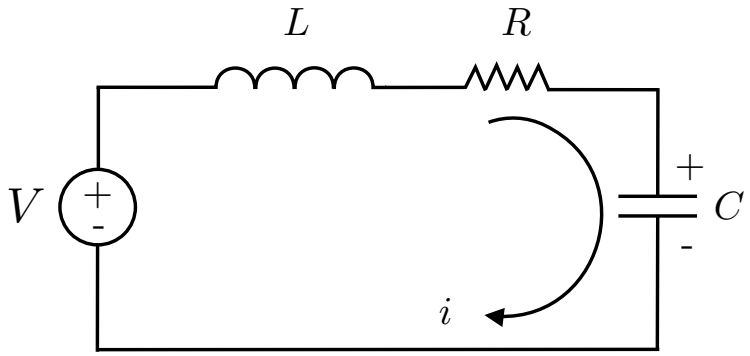

Figure 3.6. Series RLC circuit.

where $\dot{i} = \frac{dq}{dt}$, q denotes electric charge. The arithmetic means of the bounds give the nominal values of the varying parameters "R" and "C".

$$R_0 = \frac{\underline{R} + \overline{R}}{2}, \tag{3.11a}$$

$$C_0 = \frac{\underline{C} + \overline{C}}{2}. \tag{3.11b}$$

The variation of the parameters can be written as

$$R_v = \overline{R} - R_0, \tag{3.12a}$$

$$C_v = \overline{C} - C_0. \tag{3.12b}$$

The nominal values of the varying parameters are obtained. The mesh equation of the system is re-written by the separate incorporation of both the nominal and varying part of the parameters.

$$\underbrace{L\ddot{q} + R_0\dot{q} + \frac{1}{C_0}\,q}_{\text{Nominal System}} = V + R_v\dot{q} + C_v q, \quad Z_R := \dot{q} \quad \textit{and} \quad Z_C := q \tag{3.13a}$$

$$= V + R_v Z_R + C_v Z_C, \quad W_R = R_v Z_R \quad \textit{and} \quad W_C = C_v Z_C \tag{3.13b}$$

$$= V + W_R + W_C \tag{3.13c}$$

The equations can be put into the following form

$$\begin{pmatrix} \dot{q} \\ \ddot{q} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -\frac{1}{LC_0} & -\frac{1}{LR_0} \end{pmatrix} \begin{pmatrix} q \\ \dot{q} \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ \frac{1}{L} & \frac{1}{L} \end{pmatrix} \begin{pmatrix} W_C \\ W_R \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{1}{L} \end{pmatrix} V \tag{3.14a}$$

$$\begin{pmatrix} Z_C \\ Z_R \end{pmatrix} = \begin{pmatrix} 1 & 0 \\ 0 & 1 \end{pmatrix} \begin{pmatrix} q \\ \dot{q} \end{pmatrix} \tag{3.14b}$$

$$\begin{pmatrix} W_C \\ W_R \end{pmatrix} = \begin{pmatrix} C_v & 0 \\ 0 & R_v \end{pmatrix} \begin{pmatrix} Z_C \\ Z_R \end{pmatrix} \tag{3.14c}$$

The block diagram structure can be obtained as

$$\begin{pmatrix} \dot{q} \\ \ddot{q} \\ Z_C \\ Z_R \\ y \end{pmatrix} = \underbrace{\left(\begin{array}{cc:ccc} 0 & 1 & 0 & 0 & 0 \\ -\frac{1}{LC_0} & -\frac{1}{LR_0} & \frac{1}{L} & \frac{1}{L} & \frac{1}{L} \\ \hdashline 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ \hdashline 1 & 0 & 0 & 0 & 0 \end{array}\right)}_{M} \begin{pmatrix} q \\ \dot{q} \\ W_C \\ W_R \\ V \end{pmatrix} \tag{3.15a}$$

$$\begin{pmatrix} W_C \\ W_R \end{pmatrix} = \underbrace{\begin{pmatrix} C_v & 0 \\ 0 & R_v \end{pmatrix}}_{\Delta} \begin{pmatrix} Z_C \\ Z_R \end{pmatrix} \tag{3.15b}$$

$$F_u(M_u,\Delta) = \underbrace{\begin{pmatrix} 1 & 0 \\ 0 & 1 \\ \hdashline 1 & 0 \end{pmatrix}}_{M_{u_{21}}} \underbrace{\begin{pmatrix} C_v & 0 \\ 0 & R_v \end{pmatrix}}_{\Delta} \left(I - \underbrace{\begin{pmatrix} 0 & 1 \\ \frac{1}{LC_0} & \frac{1}{LR_0} \end{pmatrix}}_{M_{u_{11}}} \underbrace{\begin{pmatrix} C_v & 0 \\ 0 & R_v \end{pmatrix}}_{\Delta} \right)^{-1} \underbrace{\left(\begin{array}{cc:c} 0 & 0 & 0 \\ \frac{1}{L} & \frac{1}{L} & \frac{1}{L} \end{array}\right)}_{M_{u_{12}}}$$

$$+ \underbrace{\left(\begin{array}{cc:c} 0 & 0 & 0 \\ 0 & 0 & 0 \\ \hdashline 0 & 0 & 0 \end{array}\right)}_{M_{u_{22}}} \text{ [3]. (3.16)}$$

3.2. Normalization

Normalization of the parameter variations between -1 and $+1$ is a better way to analyze uncertain systems. The problem of "inverting" a non-invertible system can also be covered under the normalization topic to some extent.A parameter dependent system, which depends on the varying parameters $\sigma_1 \in [\underline{\sigma_1}, \overline{\sigma_1}]$ and $\sigma_2 \in [\underline{\sigma_2}, \overline{\sigma_2}]$, can be given as an example. Normalization consists of replacing σ_1, σ_2 by $\tilde{\sigma_1}, \tilde{\sigma_2}$ where

$$\sigma_1 = \frac{\overline{\sigma_1} + \underline{\sigma_1}}{2} + \frac{\overline{\sigma_1} - \underline{\sigma_1}}{2}\,\tilde{\sigma_1} \tag{3.17a}$$

$$\sigma_2 = \frac{\overline{\sigma_2} + \underline{\sigma_2}}{2} + \frac{\overline{\sigma_2} - \underline{\sigma_2}}{2}\,\tilde{\sigma_2} \tag{3.17b}$$

The parameters $\tilde{\sigma_1}, \tilde{\sigma_2}$ vary between -1 and +1 that is, $\tilde{\sigma_1}, \tilde{\sigma_2} \in [-1, +1]$.

More generally, replacing Δ by $P\Delta' Q + R$ yields normalization. The expressions of P,Q and R in the above example are

$$P = \begin{pmatrix} I_{n1} & 0 \\ 0 & I_{n2} \end{pmatrix} ; Q = \begin{pmatrix} I_{n1}\,\frac{\overline{\sigma_1}-\underline{\sigma_1}}{2} & 0 \\ 0 & I_{n2}\,\frac{\overline{\sigma_2}-\underline{\sigma_2}}{2} \end{pmatrix} ; R = \begin{pmatrix} I_{n1}\,\frac{\overline{\sigma_1}+\underline{\sigma_1}}{2} & 0 \\ 0 & I_{n2}\,\frac{\overline{\sigma_2}+\underline{\sigma_2}}{2} \end{pmatrix}$$

The new equivalent of the uncertain system in which Δ is replaced by Δ' is computed by the following lemma.

Lemma 3.2.1. *If the scheduling matrix Δ is replaced by $P\Delta'Q + R$ the upper LFT structures will be equivalent*

$$F_u\left(\begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}, \Delta\right) = F_u\left(\begin{bmatrix} M'_{11} & M'_{12} \\ M'_{21} & M'_{22} \end{bmatrix}, \Delta'\right) \tag{3.18}$$

where

$$M'_{11} = Q(I - M_{11}R)^{-1}M_{11}P \tag{3.19a}$$

$$M'_{12} = Q(I - M_{11}R)^{-1}M_{12} \tag{3.19b}$$

$$M'_{21} = M_{21}P + M_{21}R(I - M_{11}R)^{-1}M_{11}P \tag{3.19c}$$

$$M'_{22} = M_{22} + M_{21}R(I - M_{11}R)^{-1}M_{12} \ \textit{[15].} \tag{3.19d}$$

In this step, the implementation of this normalization methodology is applied to Example 3.1.1.

Example 3.2.1. The varying parameters: resistance "R" and capacitance "C" given in Example 3.1.1 can be normalized as follows

$$C = C_0 - C'\sigma_C, \quad \sigma_C = [-1, +1] \ \textit{where} \ C_0 := \frac{\overline{C} + \underline{C}}{2} \ \textit{and} \ C' := -\frac{\overline{C} - \underline{C}}{2} \tag{3.20a}$$

$$R = R_0 - R'\sigma_R, \quad \sigma_R = [-1, +1] \ \textit{where} \ R_0 := \frac{\overline{R} + \underline{R}}{2} \ \textit{and} \ R' := -\frac{\overline{R} - \underline{R}}{2} \tag{3.20b}$$

The mesh equation of the system can be re-written as

$$\underbrace{L\ddot{q} + R_0\dot{q} + \frac{1}{C_0}\,q}_{\text{Nominal System}} = V + R'\sigma_R\dot{q} + C'\sigma_C q, \quad Z_R := R'\dot{q} \quad \textit{and} \quad Z_C := C'q \tag{3.21a}$$

$$= V + \sigma_R Z_R + \sigma_C Z_C, \quad W_R = \sigma_R Z_R \quad \textit{and} \quad W_C = \sigma_C Z_C \tag{3.21b}$$

$$= V + W_R + W_C \tag{3.21c}$$

The equations can be put into the following form again as

$$\begin{pmatrix} \dot{q} \\ \ddot{q} \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ -\frac{1}{LC_0} & -\frac{1}{LR_0} \end{pmatrix} \begin{pmatrix} q \\ \dot{q} \end{pmatrix} + \begin{pmatrix} 0 & 0 \\ \frac{1}{L} & \frac{1}{L} \end{pmatrix} \begin{pmatrix} W_C \\ W_R \end{pmatrix} + \begin{pmatrix} 0 \\ \frac{1}{L} \end{pmatrix} V \tag{3.22a}$$

$$\begin{pmatrix} Z_C \\ Z_R \end{pmatrix} = \begin{pmatrix} C' & 0 \\ 0 & R' \end{pmatrix} \begin{pmatrix} q \\ \dot{q} \end{pmatrix} \tag{3.22b}$$

$$\begin{pmatrix} W_C \\ W_R \end{pmatrix} = \begin{pmatrix} \sigma_C & 0 \\ 0 & \sigma_R \end{pmatrix} \begin{pmatrix} Z_C \\ Z_R \end{pmatrix} \tag{3.22c}$$

The block diagram structure can be obtained as

$$\begin{pmatrix} \dot{q} \\ \ddot{q} \\ Z_C \\ Z_R \\ y \end{pmatrix} = \underbrace{\left(\begin{array}{cc:cc:c} 0 & 1 & 0 & 0 & 0 \\ -\frac{1}{LC_0} & -\frac{1}{LR_0} & \frac{1}{L} & \frac{1}{L} & \frac{1}{L} \\ \hdashline C' & 0 & 0 & 0 & 0 \\ 0 & R' & 0 & 0 & 0 \\ \hdashline 1 & 0 & 0 & 0 & 0 \end{array}\right)}_{M} \begin{pmatrix} q \\ \dot{q} \\ W_C \\ W_R \\ V \end{pmatrix} \tag{3.23a}$$

$$\begin{pmatrix} W_C \\ W_R \end{pmatrix} = \underbrace{\begin{pmatrix} \sigma_C & 0 \\ 0 & \sigma_R \end{pmatrix}}_{\Delta} \begin{pmatrix} Z_C \\ Z_R \end{pmatrix} \tag{3.23b}$$

$$F_u(M_u,\Delta) =$$

$$\underbrace{\left(\begin{array}{cc} C' & 0 \\ 0 & R' \\ \hdashline 1 & 0 \end{array}\right)}_{M_{u_{21}}} \underbrace{\begin{pmatrix} \sigma_C & 0 \\ 0 & \sigma_R \end{pmatrix}}_{\Delta} \left(I - \underbrace{\begin{pmatrix} 0 & 1 \\ \frac{1}{LC_0} & \frac{1}{LR_0} \end{pmatrix}}_{M_{u_{11}}} \underbrace{\begin{pmatrix} \sigma_C & 0 \\ 0 & \sigma_R \end{pmatrix}}_{\Delta} \right)^{-1} \underbrace{\left(\begin{array}{cc:c} 0 & 0 & 0 \\ \frac{1}{L} & \frac{1}{L} & \frac{1}{L} \end{array}\right)}_{M_{u_{12}}}$$

$$+ \underbrace{\left(\begin{array}{cc:c} 0 & 0 & 0 \\ 0 & 0 & 0 \\ \hdashline 0 & 0 & 0 \end{array}\right)}_{M_{u_{22}}} \quad [3]. \quad (3.24)$$

It must be again stated that uncertain systems can be represented in LFT structure if the time varying parameters are in the rational polynomial form in the equations. For the LFR representation, LFR Toolbox [15] is employed which provides a number of useful processes including obtaining the LFT structure and normalization of the RIP quasi-LPV model. This toolbox and RIP model in LFT structure will be explained in detail in Chapter 5. Additionally, LFT structure of an uncertain system can also be used in the design of the controller. In the next chapter, performance and stability analysis and controller design methodology will be presented.

4. ANALYSIS AND DESIGN: THEORETICAL BACKGROUND

This chapter explains stability, performance analysis and the controller design of an LFT structured LPV system. Some basic definitions of matrix analysis and a brief introduction to Linear Matrix Inequalities (LMIs) are presented in the first section. The stability and performance conditions of a parameter dependent system will be presented in terms of LMIs in the second section. In the third section, a controller design methodology will be presented.

4.1. Linear Matrix Inequalities

An $n \times n$ real symmetric matrix $M = M^T$ is said to be [7], [12]

- Positive Definite if $x^T M x > 0,\ \forall x \in \mathbb{R}^n,\ x \neq 0$
- Positive Semi-Definite if $x^T M x \geq 0,\ \forall x \in \mathbb{R}^n,\ x \neq 0$
- Negative Definite if $x^T M x < 0,\ \forall x \in \mathbb{R}^n,\ x \neq 0$
- Negative Semi-Definite if $x^T M x \leq 0,\ \forall x \in \mathbb{R}^n,\ x \neq 0$

Additionally, sign definiteness of a matrix can be determined by its eigenvalues(λ). A $n \times n$ real symmetric matrix $M = M^T$ is said to be [7], [12]

- Positive Definite if and only if $\lambda_i > 0,\ \forall i = 1 : n$
- Positive Semi-Definite if and only if $\lambda_i \geq 0,\ \forall i = 1 : n$
- Negative Definite if and only if $\lambda_i < 0,\ \forall i = 1 : n$
- Negative Semi-Definite if and only if $\lambda_i \leq 0,\ \forall i = 1 : n$

Let $A_1, A_2, \cdots, A_n = A_n^T \in \mathbb{R}^{n \times n}$ be real symmetric matrices and $y_1, y_2, \cdots, y_n \in \mathbb{R}^n$ be variables, a linear matrix inequality has the form [6]:

$$A_0 + y_1 A_1 + y_2 A_2 + \cdots + y_n A_n \succ 0$$

The symbols $\prec, \preceq, \succeq, \succ$ are used to describe the sign of a matrix. A system of k individual LMIs [13] :

$$\begin{gathered} F_0^1 + x_1 F_1^1 + x_2 F_2^1 + \cdots + x_n F_n^1 \prec 0 \\ \vdots \\ F_0^m + x_1 F_1^m + x_2 F_2^m + \cdots + x_n F_n^m \prec 0 \end{gathered}$$

is equivalent to the following single LMI.

$$\begin{pmatrix} F_0^1 & & 0 \\ & \ddots & \\ 0 & & F_0^m \end{pmatrix} + \sum_{k=1}^{n} x_k \begin{pmatrix} F_k^1 & & 0 \\ & \ddots & \\ 0 & & F_k^m \end{pmatrix} \prec 0$$

which is very advantageous in a way that it yields simplicity and reduces computational complexity for the solution.

Generally, it is very hard to find analytical solutions to LMIs. However, thanks to today's powerful computers, numerical solutions by employing convex optimization techniques can be obtained. There are many ways to solve LMIs numerically. One of the most popular and efficient ways is solving them in the MATLAB environment. Several MATLAB based solvers are developed for the numerical solution such as: LMI-LAB, SEDUMI SDPT3 and many more [19] . In this thesis, LMILAB [20] based IQC Synthesis Toolbox [14] is used for parsing and the solution and the details of the solver is given in Chapter 5. For more applications, see [5], [6]

4.2. Stability and Performance Analysis

The quadratic stability of an uncertain autonomous system is given in the following LMI form which is called as *Lyapunov Inequality.*

Theorem 4.1.1. *The uncertain autonomous system $\dot{x} = A(p)x$ with $p \in \boldsymbol{p}$ is said to be quadratically stable if there exists $X \succ 0$ with*

$$A(p)^T X + XA(p) \prec 0, \quad \forall_p \in \boldsymbol{p}$$

provided that $V(x) = x^T Xx$ is defined as quadratic Lyapunov function.

Consider the uncertain autonomous system illustrated in Figure 3.1 where

$$w_u := w,$$
$$z_u := z$$

and

$$\dot{x} = Ax + Bw_u \tag{4.1a}$$
$$z_u = Cx + Dw_u, \quad w_u = \Delta(z_u) \ \textit{[6]}. \tag{4.1b}$$

In this representation, A is assumed to be Hurwitz and Δ is in the block diagonal form. Moreover, the following theorem with the LMI conditions give the robust stability LMI conditions.

Theorem 4.1.2. *Assume that $F(p) = A+B\Delta(p)(I-D\Delta(p))^{-1}C$. Recall the definition of quadratic stability: There exists $X \succ 0$ with*

$$F(p)^T X + XF(p) \prec 0, \quad \forall_p \in \boldsymbol{p} \tag{4.2}$$

LFR is well-posed, (4.2) holds, if there exists a multiplier $\boldsymbol{P}$ with

$$\begin{pmatrix} \Delta(p) \\ I \end{pmatrix}^T \boldsymbol{P} \begin{pmatrix} \Delta(p) \\ I \end{pmatrix} \succeq 0, \quad \forall_p \in \boldsymbol{p} \tag{4.3}$$

that also satisfies

$$\begin{pmatrix} I & 0 \\ A & B \end{pmatrix}^T \begin{pmatrix} 0 & X \\ X & 0 \end{pmatrix} \begin{pmatrix} I & 0 \\ A & B \end{pmatrix} + \begin{pmatrix} 0 & I \\ C & D \end{pmatrix}^T \boldsymbol{P} \begin{pmatrix} 0 & I \\ C & D \end{pmatrix} \prec 0 \ \textit{[13]}. \tag{4.4}$$

Exogenous input called as *disturbance* can be added to uncertain autonomous system in (4.1). In this case, the system can remain still stable with the incorporation of the disturbances. Thus, an extra channel is required to include the effect of disturbance on the system. This channel is called as *Performance Channel* and the system is now turn into the following structure.

$$\dot{x} = Ax + B_1 w_u + B_p w_p \tag{4.5a}$$

$$z_u = C_1 x + D_{11} w_u + D_{1p} w_p \ \textit{and}, \quad w_u = \Delta(z_u) \tag{4.5b}$$

$$z_p = C_p x + D_{1p} w_u + D_{pp} w_p \tag{4.5c}$$

Figure 4.1 illustrates an uncertain system with performance channel.

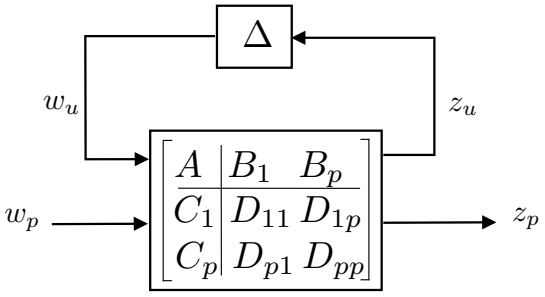

Figure 4.1. Uncertain system with performance channel.

Consider the following manners of an LTI system with state space description

$$\dot{x} = Ax + Bw, \quad x(0) = 0$$

$$z = Cx + Dw$$

or input-output description $z = Tw$ and transfer matrix $T(s) = C(sI - A)^{-1}B + D$, which are used to quantify the effect of w (disturbance) onto z (measured output).

- w_p and z_p are deterministic signals: system gain
- w_p white noise: asymptotic output variance
- w_p impulses: energy of outputs

In our case, the system gain or L_2 norm is mostly used for analysis.

Definition 4.1.1. *The L_2 norm of a vector is defined as*

$$\|x\|_2 := \sqrt{\int_0^\infty \|x(t)\|^2 \, d(t)} \tag{4.6}$$

If the norms for w_p and z_p are chosen to quantify the size of the signals then the worst system amplification would be system or L_2 gain which will be formulated rigorously as follows

$$\|\boldsymbol{T}\|_{i2} := \sup_{0<\|w_p\|<\infty} \frac{\|Tw_p\|}{\|w_p\|} \quad \textit{[3].} \tag{4.7}$$

The following theorem gives the stability conditions for the system in (4.5) regarding the L_2 norm below γ.

Theorem 4.1.3. *Consider the LPV system with performance channel in (4.5). Let $\gamma > 0$ and A is Hurwitz, then the LPV system in (4.5) remains stable with L_2 norm below γ if there exists $X = X^T \succ 0$ and matrices $Q = Q^T$, S and $R = R^T$ such that*

$$\left(\begin{array}{ccc} I & 0 & 0 \\ A & B_1 & B_p \\ \hline 0 & I & 0 \\ C_1 & D_{11} & D_{1p} \\ \hline 0 & 0 & I \\ C_p & D_{p1} & D_{pp} \end{array}\right)^T \left(\begin{array}{cc|cc|cc} 0 & X & 0 & 0 & 0 & 0 \\ X & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & Q & S & 0 & 0 \\ 0 & 0 & S^T & R & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & -\gamma I & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{I}{\gamma} \end{array}\right) \left(\begin{array}{ccc} I & 0 & 0 \\ A & B_1 & B_p \\ \hline 0 & I & 0 \\ C_1 & D_{11} & D_{1p} \\ \hline 0 & 0 & I \\ C_p & D_{p1} & D_{pp} \end{array}\right) \prec 0 \qquad (4.8)$$

and

$$\begin{pmatrix} \Delta(p) \\ I \end{pmatrix}^T \begin{pmatrix} Q & S \\ S^T & R \end{pmatrix} \begin{pmatrix} \Delta(p) \\ I \end{pmatrix} \succ 0, \quad \forall_p \in \boldsymbol{p} \ \textit{[3]}. \qquad (4.9)$$

4.3. Controller Synthesis

Throughout this section, controller synthesis procedure is explained. The LMI conditons required for controller synthesis tht guarantee robust stability and performance are also provided.

In the next figure, it is illustrated that a control channel is added to Figure 4.1. Hence, a control input and output is incorporated in the uncertain system. The equations for Figure 4.2 are

$$\dot{x} = Ax + B_1 w_u + B_p w_p + B_2 u \qquad (4.10a)$$

$$z_u = C_1 x + D_{11} w_u + D_{1p} w_p + D_{12} u \qquad w_u = \Delta(z_u) \qquad (4.10b)$$

$$z_p = C_p x + D_{1p} z_u + D_{pp} z_p + D_{p2} u \qquad (4.10c)$$

$$y = C_2 x + D_{21} w_u + D_{2p} w_p \qquad (4.10d)$$

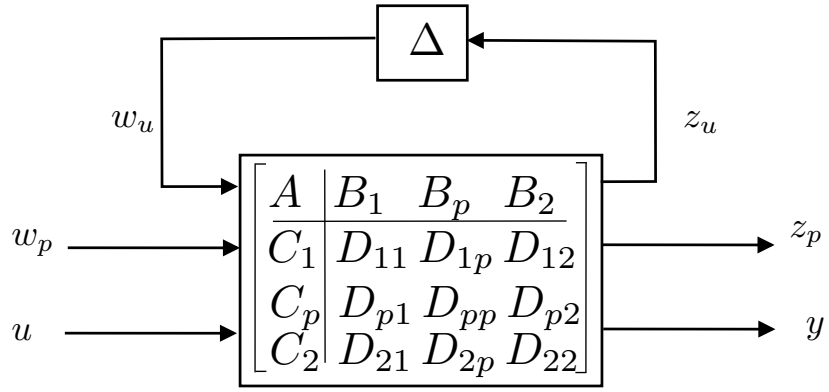

Figure 4.2. Uncertain system with control and performance channel.

Our aim is to design a controller with the following structure.

$$\dot{x}_K = A_K x_K + B_{K_2} y + B_{K_1} w_{K_u} \tag{4.11a}$$

$$u = C_{K_2} x_K + D_{K_{22}} y + D_{K_{21}} w_{K_u} \tag{4.11b}$$

$$z_{K_u} = C_{K_1} x_K + D_{K_{12}} y + D_{K_{11}} w_{K_u}, \qquad w_{K_u} = \Delta(z_{K_u}). \tag{4.11c}$$

The controller block is also illustrated in Figure 4.3 below.

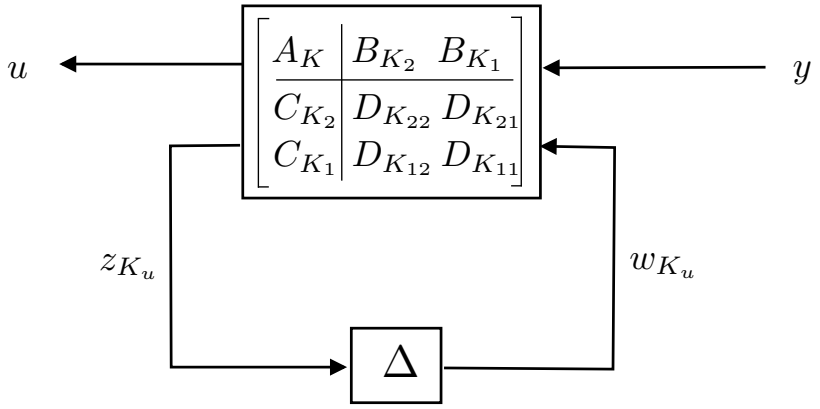

Figure 4.3. Controller block of the uncertain system.

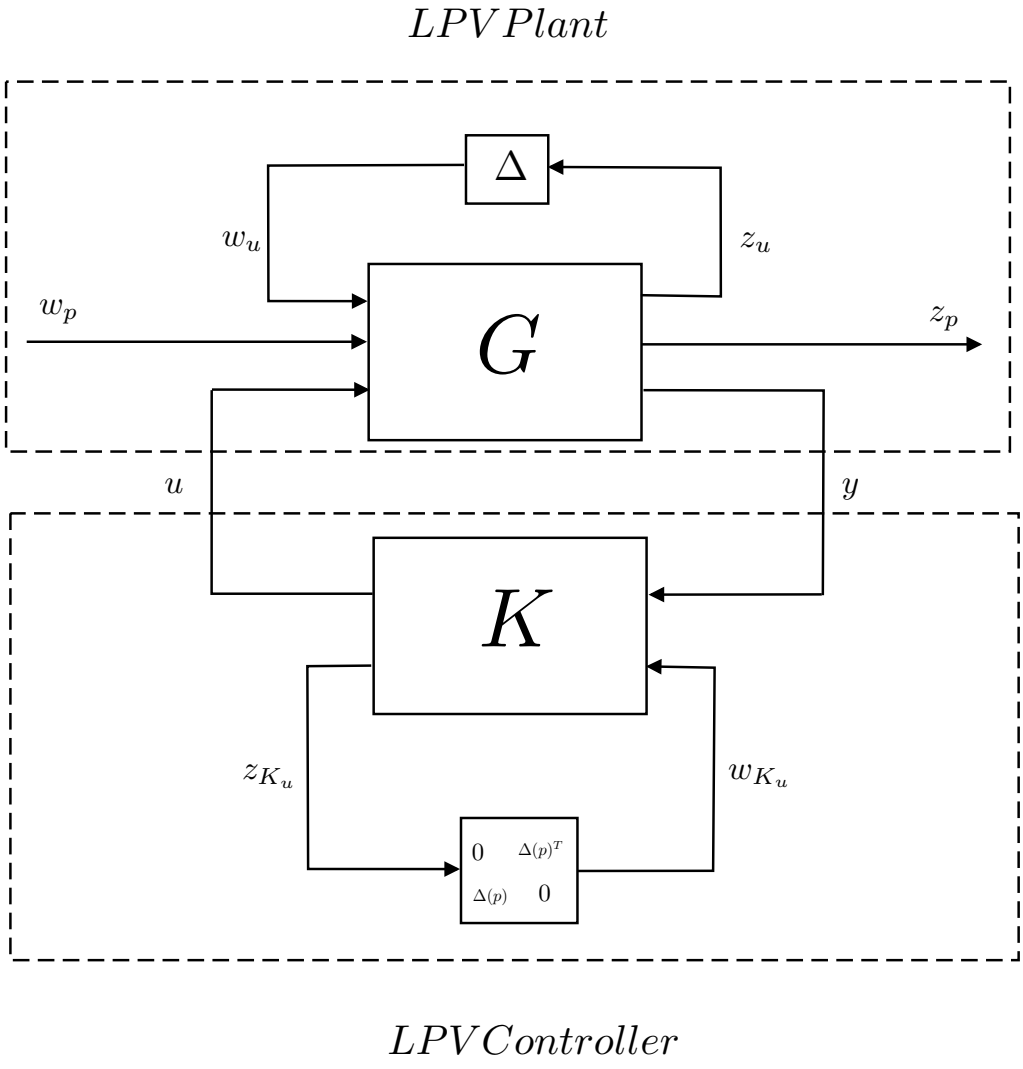

Figure 4.4. Closed-loop structure of LPV plant with LPV controller.

The closed-loop LPV system can be expressed in the following form [3]

$$G_{Cl} = \left[\begin{array}{c|c} A_{Cl} & B_{Cl} \\ \hline C_{Cl} & D_{Cl} \end{array}\right] \tag{4.12a}$$

$$A_{Cl} = A^l + B_2^l K C_2^l, \qquad B_{Cl} = \widehat{B_1^l} + B_2^l K D_{21} \tag{4.12b}$$

$$C_{Cl} = \widehat{C_1^l} + D_{12}^l K C_2^l, \quad D_{Cl} = \widehat{D_{11}^l} + \widehat{D_2^l} K D_{21}^l \tag{4.12c}$$

where

$$\widehat{B_1^l} := \begin{pmatrix} B_1^l & B_1^l \end{pmatrix}, \ \widehat{C_1^l} := \begin{pmatrix} C_1^l \\ C_p^l \end{pmatrix}, \ \widehat{D_{11}^l} := \begin{pmatrix} D_{11}^l & D_{1p}^l \\ D_{p1}^l & D_{pp}^l \end{pmatrix}, \ \widehat{D_{12}^l} := \begin{pmatrix} D_{12}^l \\ D_{p2}^l \end{pmatrix} \tag{4.13}$$

and

$$A^l := \begin{pmatrix} A & 0 \\ 0 & 0 \end{pmatrix}, \ C_2^l := \begin{pmatrix} C_2 & 0 \\ 0 & 0 \\ 0 & I_n \end{pmatrix}, \ B_2^l := \begin{pmatrix} B_2 & 0 & 0 \\ 0 & 0 & I_n \end{pmatrix}, \ D_{p2}^l := \begin{pmatrix} D_{p2} & 0 & 0 \end{pmatrix} \tag{4.14}$$

$$C_p^l := \begin{pmatrix} C_p & 0 \end{pmatrix}, \ D_{1p}^l := \begin{pmatrix} D_{1p} \\ 0 \end{pmatrix}, \ D_{p1}^l := \begin{pmatrix} D_{p1} & 0 \end{pmatrix}, \ D_{pp}^l := D_{pp} \tag{4.15}$$

$$C_1^l := \begin{pmatrix} C_1 & 0 \\ 0 & 0_{m\times n} \end{pmatrix}, \ D_{12}^l := \begin{pmatrix} D_{21} & 0 & D_{2p} \\ 0 & I_m & 0 \\ 0 & 0 & 0 \end{pmatrix}, \ B_1^l := \begin{pmatrix} B_1 & 0 \\ 0 & 0_{n\times m} \end{pmatrix} \tag{4.16}$$

$$D_{11}^l := \begin{pmatrix} D_{11} & 0 \\ 0 & 0_{m\times n} \end{pmatrix}, \ K := \begin{pmatrix} D_{K_{22}} & D_{K_{21}} & C_{K_2} \\ D_{K_{12}} & D_{K_{11}} & C_{K_1} \\ B_{K_2} & B_{K_1} & A_K \end{pmatrix}, \ D_{12}^l := \begin{pmatrix} D_{12} & 0 & 0 \\ 0 & I_{n\times m} & 0 \end{pmatrix} \tag{4.17}$$

Additionally, n and m denotes the number of states and repeated parameters respectively.

The aim is to design a controller that would guarantee the asymptotic stability of the closed-loop system in Figure 4.4 and to ensure that the performance inequality $\|w_p \to z_p\|_{i2} < \gamma$ is satisfied for all permissible parameter trajectories and some scalar γ. Therefore, the following theorem is used in the controller design.

Theorem 4.2.1. *There exists a controller (4.11) for LPV plant (4.10) such that the closed-loop system in Figure 4.4 is asymptotically stable and satisfies the performance inequality* $\|w_p \rightarrow z_p\|_{i2} < \gamma$ *for some* γ *and all time varying matrices* $\Delta \in co\{\Delta(p^1), \ldots \Delta(p^m)\}$ *if there exist matrices* X, Y, Q, R, S, $\widetilde{Q}, \widetilde{R}, \widetilde{S}$ *such that*

$$\begin{pmatrix} X & I \\ I & Y \end{pmatrix} \succ 0 \tag{4.18}$$

$$\Psi^T \left(\begin{array}{c} * \\ * \\ \hline * \\ * \\ \hline * \\ * \end{array}\right)^T \left(\begin{array}{cc|cc|cc} 0 & X & 0 & 0 & 0 & 0 \\ X & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & Q & S & 0 & 0 \\ 0 & 0 & S^T & R & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & -\gamma I & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{\gamma} I \end{array}\right) \left(\begin{array}{ccc} I & 0 & 0 \\ A & B_1 & B_p \\ \hline 0 & I & 0 \\ C_1 & D_{11} & D_{1p} \\ \hline 0 & 0 & I \\ C_p & D_{p1} & D_{pp} \end{array}\right) \Psi \prec 0 \tag{4.19}$$

$$\Phi^T \left(\begin{array}{c} * \\ * \\ \hline * \\ * \\ \hline * \\ * \end{array}\right)^T \left(\begin{array}{cc|cc|cc} 0 & Y & 0 & 0 & 0 & 0 \\ Y & 0 & 0 & 0 & 0 & 0 \\ \hline 0 & 0 & \widetilde{Q} & \widetilde{S} & 0 & 0 \\ 0 & 0 & \widetilde{S}^T & \widetilde{R} & 0 & 0 \\ \hline 0 & 0 & 0 & 0 & -\gamma I & 0 \\ 0 & 0 & 0 & 0 & 0 & \frac{1}{\gamma} I \end{array}\right) \left(\begin{array}{ccc} -A^T & -C_1^T & -C_p^T \\ I & 0 & 0 \\ \hline -B_1^T & D_{11}^T & D_{p1}^T \\ 0 & I & 0 \\ \hline -B_p^T & -D_{1p}^T & -D_{pp}^T \\ 0 & 0 & I \end{array}\right) \Phi \succ 0 \tag{4.20}$$

$$Q \prec 0, \ \begin{pmatrix} \Delta(p^j) \\ I \end{pmatrix}^T \underbrace{\begin{pmatrix} Q & S \\ S^T & R \end{pmatrix}}_{P} \begin{pmatrix} \Delta(p)^j \\ I \end{pmatrix} \succ 0, \ j = 1, \ldots, m \tag{4.21}$$

$$\widetilde{R} \succ 0, \ \begin{pmatrix} I \\ -\Delta(p^j)^T \end{pmatrix}^T \underbrace{\begin{pmatrix} \widetilde{Q} & \widetilde{S} \\ \widetilde{S}^T & \widetilde{R} \end{pmatrix}}_{\widetilde{P}} \begin{pmatrix} I \\ -\Delta(p^j)^T \end{pmatrix} \prec 0, \ j = 1, \ldots, m \tag{4.22}$$

where Ψ *denotes a basis matrix of* $\ker\,[C_2\ D_{21}\ D_{2p}]$ *and* Φ *denotes a basis matrix of* $\ker\,[B_2^T\ D_{12}^T\ D_{p2}^T]$ *[13].*

After finding the multipliers and the Lyapunov matrix, controller K can be obtained through the following procedure. First, state space equations of the closed-loop system can be written as [15]

$$\begin{pmatrix} \dot{\xi} \\ z_d \\ z_c \\ z_p \end{pmatrix} = \begin{pmatrix} \boldsymbol{A} & \boldsymbol{B_d} & \boldsymbol{B_c} & \boldsymbol{B_p} \\ \boldsymbol{C_d} & \boldsymbol{D_{dd}} & \boldsymbol{D_{dc}} & \boldsymbol{D_{dp}} \\ \boldsymbol{C_c} & \boldsymbol{D_{cd}} & \boldsymbol{D_{cc}} & \boldsymbol{D_{cp}} \\ \boldsymbol{C_p} & \boldsymbol{D_{pd}} & \boldsymbol{D_{pc}} & \boldsymbol{D_{pp}} \end{pmatrix} \begin{pmatrix} \xi \\ w_d \\ w_c \\ w_p \end{pmatrix}, \quad w_d = \Delta(p) z_d, \quad w_c = \Delta_c(p) z_c. \tag{4.23}$$

The performance level γ^* for the closed-loop is achieved if there exists

$$\boldsymbol{X} \succ 0, \qquad \boldsymbol{P} = \begin{pmatrix} Q & S & Q_{12} & S_{12} \\ S^T & R & S_{12}^T & R_{12} \\ Q_{21} & S_{21} & Q_{22} & S_{22} \\ S_{21}^T & R_{21} & S_{22}^T & R_{22} \end{pmatrix} \tag{4.24}$$

that satisfy

$$\begin{pmatrix} \Delta(p) & 0 \\ I & 0 \\ 0 & \Delta_c(p) \\ 0 & I \end{pmatrix}^T \boldsymbol{P} \begin{pmatrix} \Delta(p) & 0 \\ I & 0 \\ 0 & \Delta_c(p) \\ 0 & I \end{pmatrix} \succ 0, \quad \forall_p \in \Pi \tag{4.25}$$

and

$$\begin{pmatrix} * \\ * \\ * \\ * \\ * \\ * \\ * \\ * \\ * \end{pmatrix}^T \begin{pmatrix} 0 & \boldsymbol{X} & 0 & 0 & 0 & 0 & 0 & 0 \\ \boldsymbol{X} & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & Q & S & Q_{12} & S_{12} & 0 & 0 \\ 0 & 0 & S^T & R & S_{12}^T & R_{12} & 0 & 0 \\ 0 & 0 & Q_{21} & S_{21} & Q_{22} & S_{22} & 0 & 0 \\ 0 & 0 & S_{21}^T & R_{21} & S_{22}^T & R_{22} & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -\gamma^* I & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & \frac{1}{\gamma^*} I \end{pmatrix} \begin{pmatrix} I & 0 & 0 & 0 \\ \boldsymbol{A} & \boldsymbol{B_d} & \boldsymbol{B_c} & \boldsymbol{B_p} \\ 0 & I & 0 & 0 \\ \boldsymbol{C_d} & \boldsymbol{D_{dd}} & \boldsymbol{D_{dc}} & \boldsymbol{D_{dp}} \\ 0 & 0 & I & 0 \\ \boldsymbol{C_c} & \boldsymbol{D_{cd}} & \boldsymbol{D_{cc}} & \boldsymbol{D_{cp}} \\ 0 & 0 & 0 & I \\ \boldsymbol{C_p} & \boldsymbol{D_{pd}} & \boldsymbol{D_{pc}} & \boldsymbol{D_{pp}} \end{pmatrix} \prec 0 \tag{4.26}$$

After that point the steps are given to construct the LPV controller based on full block scalings [15].

- Based on X and Y, the matrix $\boldsymbol{X}$ is constructed such that

$$\boldsymbol{X} = \begin{pmatrix} X & * \\ * & * \end{pmatrix}, \quad \textit{and} \quad \boldsymbol{X}^{-1} = \begin{pmatrix} Y & * \\ * & * \end{pmatrix} \tag{4.27}$$

  Therefore the matrix X is extended by suitable blocks such that $\boldsymbol{X}$ is positive definite and the inverse has Y as its left-upper block. Generally, Cholesky factorizations of X and Y are used for the extension algorithm.

- The already computed multipliers P and $\widetilde{P}$ which are

$$P = \begin{pmatrix} Q & S \\ S^T & R \end{pmatrix}, \quad \textit{and} \quad \widetilde{P} = \begin{pmatrix} \widetilde{Q} & \widetilde{S} \\ \widetilde{S}^T & \widetilde{R} \end{pmatrix} \tag{4.28}$$

  are similarly extended to

$$\boldsymbol{P} = \begin{pmatrix} Q & S & Q_{12} & S_{12} \\ S^T & R & S_{12}^T & R_{12} \\ Q_{21} & S_{21} & Q_{22} & S_{22} \\ S_{21}^T & R_{21} & S_{22}^T & R_{22} \end{pmatrix}, \quad and \quad \boldsymbol{P}^{-1} = \begin{pmatrix} \widetilde{Q} & \widetilde{S} & * & * \\ \widetilde{S}^T & \widetilde{R} & * & * \\ * & * & * & * \\ * & * & * & * \end{pmatrix} \tag{4.29}$$

Moreover, the size of $\Delta_c(p)$ can also be obtained by the number of positive and negative eigenvalues of the matrix $\boldsymbol{P} - \boldsymbol{P}^{-1}$.

- The parameters describing the controller is computed based on *Projection Lemma* after having determined the matrices $\boldsymbol{P}$ and $\boldsymbol{X}$.

The details of the derivation of the extension matrices and the computation of controller can be found in reference [16].

5. DESIGN AND IMPLEMENTATION OF THE LPV CONTROLLER

This chapter deals with the implementation of the theoretical results to the RIP system and it consists of four main sections; Defining system parameters, construction of LFT structure, controller design and simulation configuration. First, LFT representation of the RIP system is developed via LFR toolbox which can be founded in detail in reference [15]. Second, the LMIs (4.18), (4.19), (4.20), (4.21), (4.22) in Theorem 4.2.1, which are key to controller design, are solved using IQC Synthesis Toolbox [15] to obtain the matrix variables $X, Y, Q, R, \widetilde{Q}, \widetilde{R}$ and scalar variable γ. IQC Synthesis Toolbox utilizes LMILAB, a MATLAB toolbox created for the numerical solution of LMIs, is known to be slower when compared to some other LMI solvers such as SeDuMi. The details of each process will be given in advance.

5.1. LFT Structure of the RIP

LFT form of the RIP is derived by LFR Toolbox [15] developed by J.F. Magni. With this toolbox modeling, manipulation, order reduction and approximation of uncertain systems in LFT form can be made. The toolbox works in the MATLAB environment and it contains Simulink patch for simulation of uncertain systems in LFT structure as well.

To define the varying parameters, `lfrs` function is used. Varying parameters in (2.26) is defined as `p1, p2, p3,` and `p4` in the toolbox environment. Δ matrix is constructed in alphabetical order according to LFR Toolbox's internal procedures.

`A, B, C, D` corresponds to the system matrices $A(p), B(p), C, D$ in (2.27), and defined by putting the parameters `p1, p2, p3,` and `p4` into the equation.

After constructing the LFR system, the parameters should be normalized between [-1 1] to get better results in the design process. Thanks to LFR toolbox, the

normalization procedure is facilitated by a single function named `normalizelfr`.

However, before applying normalization procedure, minimizing the RIP LFR structure with `minlfr` yields n-D order reduction. This function takes the full order LFR structure as an input variable and minimize the order of repetitive varying parameters. For instance, `Sysmin=minlfr(sys)`.

After the order reduction procedure, `normalizelfr` can be used as `Minnormsys= normalizelfr(Sysmin,parname,dmin,dmax,dnom)`. The inputs of the function are the system to be normalized: `Sysmin`, string array for parameter names: `parname`, other three row vectors: `dmin, dmax, dnom` which contains the maximum, the minimum and the nominal values of the parameters respectively.

The maximum and the minimum bounds for the varying parameters are chosen such that the performance of the controller cannot be degraded by the range of large bounds. These bounds are given in the following table.

Table 5.1. Bounds of the parameter variations.

	$p_1 := \cos(\alpha)$	$p_2 := \sin(\alpha)$	$p_3 := \frac{\sin(\alpha)}{\alpha}$	$p_4 := \dot{\alpha}$
Max	1.01	0.5	1.01	1
Min	0.86	-0.5	0.95	-1

After the normalization procedure, input-output LFR form of the RIP is obtained by `abcd2lfr` function as `sysio=abcd2lfr([A B;C D],4)`, where `[A B;C D]` denotes minimized and normalized system matrices and `4` denotes the number of states of the system. The output `sysio` of `abcd2lfr` function defined as an LFR object in the MATLAB workspace where the system matrices and scheduling matrix Δ are

$$\begin{pmatrix} A & B_1 \\ C_1 & D_{11} \end{pmatrix} := \texttt{sysio.a},\ \begin{pmatrix} B_2 \\ D_{12} \end{pmatrix} := \texttt{sysio.b},\ \begin{pmatrix} C_2 & D_{21} \end{pmatrix} := \texttt{sysio.c},$$

$$D_{22} := \texttt{sysio.d},\ \Delta := \texttt{sysio.e}$$

Hence, the LFT structure of the RIP is obtained as in Figure 3.3.

5.2. System Set-Up

This section explains the system set-up of the RIP LFT structure illustrated in Figure 4.2. Control input and performance weights are chosen and added to the LFT representation of the RIP with performance channel as in Figure 5.1. There are two different types of weights used in the RIP system. They are added to the control input and performance channels.

The performance shaping weights, W_e are an important component of the design. They are to be chosen typically in the form of low order band-pass filters. The performance shaping weights are chosen as the following form to minimize the area of the output signals by putting an integrator $1/s$ after the output channels. Since these integrators lead the system to be unstable, a small value is added to the denominators.

$$W_e = \begin{pmatrix} \frac{1}{s+10^{-4}} & 0 \\ 0 & \frac{1}{s+10^{-4}} \end{pmatrix}$$

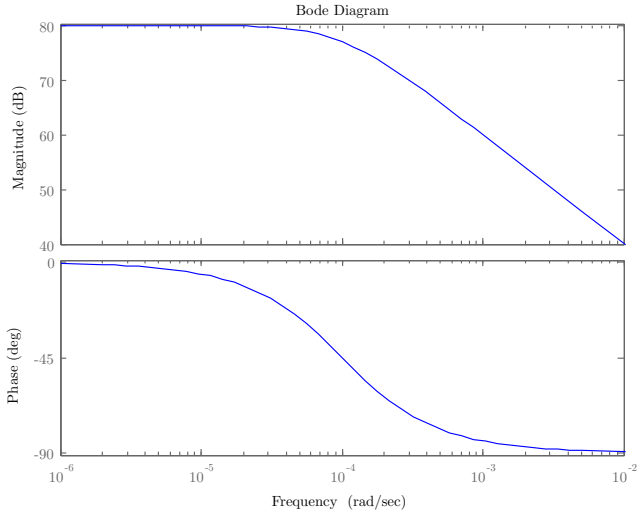

Figure 5.1. Bode plot of the performance weight, W_e.

The constraint on the control input signals are to be taken into account to reflect magnitude and rate constraints by appropriate choices of the actuator output weighting filter, W_u. After numerous trials the particular choice for this weight was set to

$$W_u(s) = \frac{s + 0.4}{s + 14.22}$$

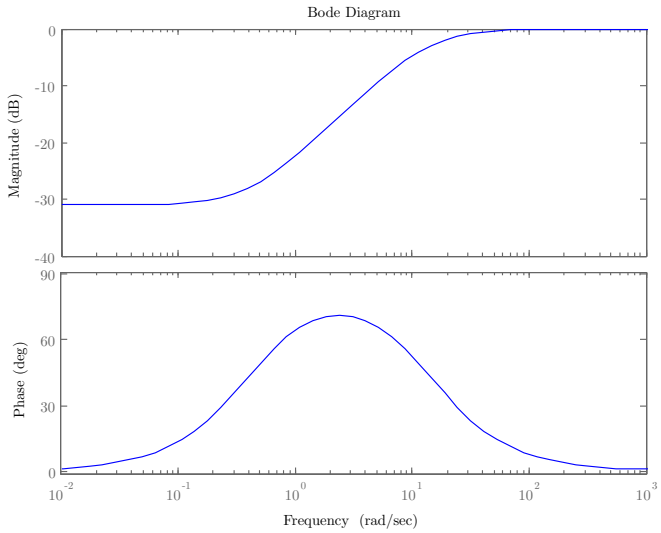

Figure 5.2. Bode plot of the actuator output weight, W_u.

Simulink interconnection of Figure 4.2 is shown in Figure 5.1.

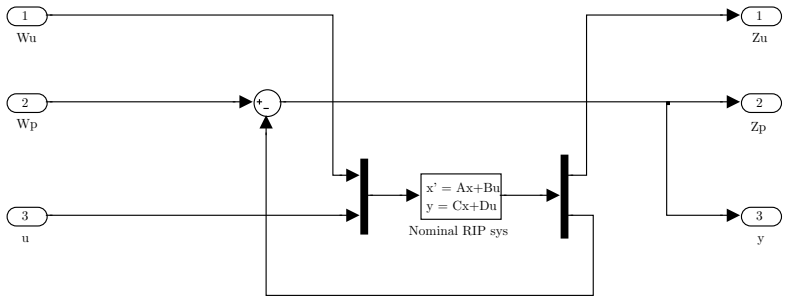

Figure 5.3. Simulink system set-up of the RIP without weights.

However, these weighting filters are not added directly to the input and output channels of the RIP system. After using `linmod` function of MATLAB, which creates a new system with new A, B, C, D matrices by adding new input and output channels to the nominal system, the weights will be incorporated to the system. The `linmod` MATLAB function is used as `[Al,Bl,Cl,Dl]= linmod('systemsetup')`, where `systemsetup` is the name of the Simulink file, `Al, Bl, Cl, Dl` are the new system matrices that are

$$Al := A \tag{5.1a}$$

$$Bl := \begin{pmatrix} B_1 & B_p & B_2 \end{pmatrix} \tag{5.1b}$$

$$Cl := \begin{pmatrix} C_1 \\ C_p \\ C_2 \end{pmatrix} \tag{5.1c}$$

$$Dl := \begin{pmatrix} D_{11} & D_{1p} & D_{12} \\ D_{p1} & D_{pp} & D_{p2} \\ D_{21} & D_{2p} & D_{22} \end{pmatrix} \tag{5.1d}$$

Using `linmod` function, the new system matrices of Figure 5.1 were obtained. In the next section weight incorporation and LMI solutions required by the controller design procedure are explained.

5.3. Controller Synthesis

This section mainly deals with the solutions of the synthesis inequalities in Theorem 4.2.1 and the controller design for the RIP. For all these procedures IQC Synthesis Toolbox is employed which provides a user-friendly approach in synthesizing the controller.

In this toolbox, to initialize an LPV synthesis problem, the user has to invoke the function `lpvpb`, which will contain all the data pertinent to the problem. This data will include the LTI part of the plant, weighting filters, scheduling matrix blocks, solutions of the synthesis conditions, controller, etc.

Once `lpvpb` object is created, the system is defined with `lpvsys` function as `prob = lpvsys(lpvpb,GG,[5 2 1;5 2 2])`, where `GG` is the the LTI part of the RIP, `[5 2 1;5 2 2]` defines the uncertainty, performance, input-output channels with quantities respectively.

The maximum and the minimum bounds of the parameter variations are included in the RIP `lpvpb` object with `lpvdel` function as `prob = lpvdel(prob,2,[0.86;1.01],'s')`.

The weighting functions W_e, W_u previously defined in Section 5.2 are added to the system at this step. Figure 5.2 illustrates the weights. With the function `lpvwt`, the weights are configured and connected to the RIP system object. `prob = lpvwt(prob,iostr,indices,wt)` is the general expression for the function. `iostr` parameter can be in or out by which the weights are determined whether to be placed at input or output channels of the RIP system. Additionally, `indices` parameter determines the order of the channel. And finally, the weigting functions are given in `wt` parameter. The following figure illustrates the interconnected RIP system with weights.

The most important part of this chapter is the solution of the synthesis inequalities.

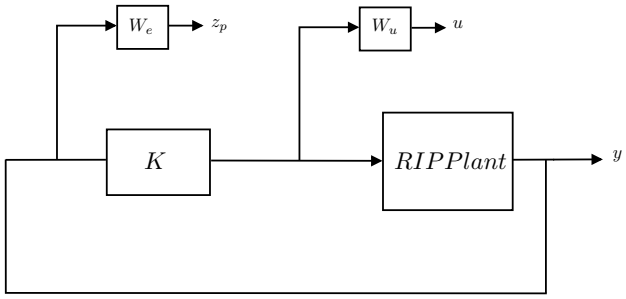

Figure 5.4. The RIP system with weights.

To solve them `lpvsyn` function of IQC Synthesis Toolbox is utilized. The function `lpvsyn` is ultimately aims to solve the aforesaid optimization problem and appending the solutions to the `lpvpb` object `prob`. With this function the matrix variables $X, Y, Q, R, \widetilde{Q}, \widetilde{R}$ and scalar variable γ are readily obtained. `prob = lpvsyn(prob)` is the general expression of this function where `prob` is the previously defined system object. The function `lpvsyn` utilizes MATLAB's Robust Control and LMI toolboxes. After the optimization process with `lpvsyn` function, scalar variable γ is obtained as 5.34.

After finding the matrix variables $X, Y, Q, R, \widetilde{Q}, \widetilde{R}$ and scalar variable γ by solving Theorem 4.2.1, the next step is find the extension and augmented matrices to guarantee the stability condition for closed-loop. For this step `prob = lpvcon(prob,dis)` function is used both finding the extension, augmented matrices and computing the controller matrices. The dimensions of the controller matrices are: $A_K^{5\text{x}5}, B_K^{5\text{x}12}, C_K^{11\text{x}5}, D_K^{11\text{x}12}$.

More information about IQC Synthesis Toolbox can be found in reference [15] in detail.

5.4. Simulation Configuration

Configured simulation set-up is shown in Figure 5.5. The nonlinear model of the RIP is used in the simulation to demonstrate the designed controller performance.

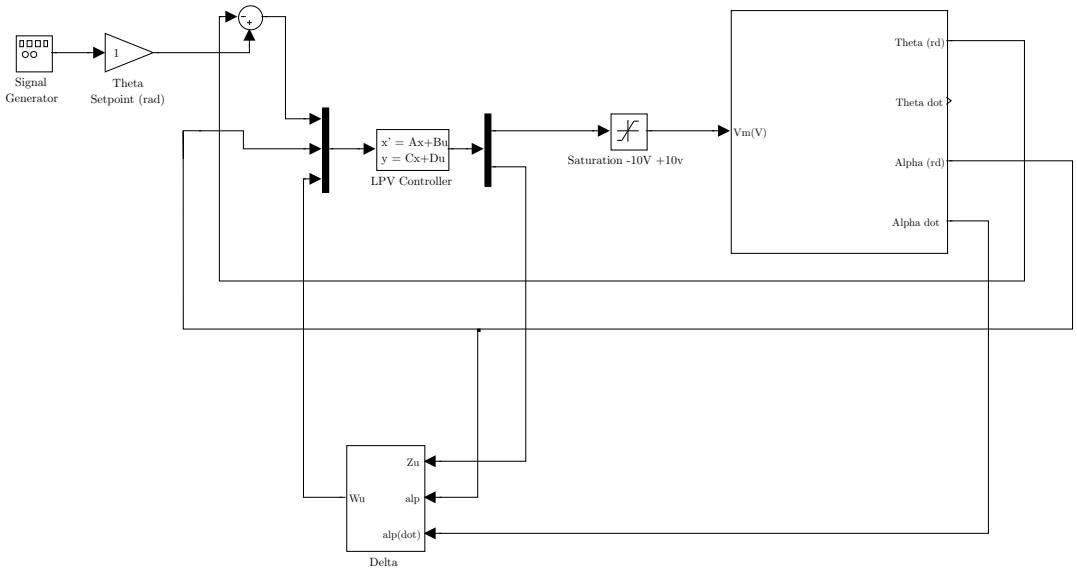

Figure 5.5. Simulation environment.

In this environment, a saturation block is added after the control signal to take the DC motor's voltage limits into consideration. However for the implementation of the the controller on the physical RIP model, the following Simulink file in Figure 5.4 is developed.

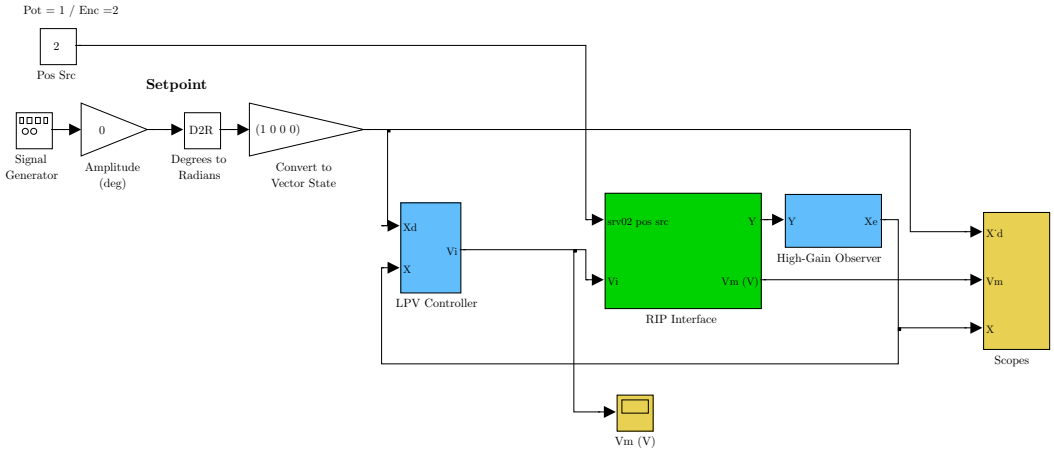

Figure 5.6. Physical RIP simulink environment.

In the model above, RIP interface block transmits the control signal via an interface card to the amplifier, hence to the dc motor. It also receives encoder signals or position information for the controller. High gain observer block smooths the encoder signals and eliminates high frequency encoder data. More details about Quanser RIP system can be found in reference [1].

Simulation results of the model and implementation of the controller to the physical RIP system is given in the next chapter.

6. SIMULATION AND EXPERIMENTAL RESULTS

6.1. LPV Controller Implementation on the Simulated RIP System

In this chapter, the performance of the designed LPV controller is demonstrated. In the first section, the designed LPV controller is compared to that of LQR controller which are simulated in three different operation conditions by giving an initial angles to the pendulum.

Table 6.1. Simulation conditions for α.

Condition	**Initial Pendulum Angle α (Degree)**	**Uncertainty (% of nominal value) Length of Pendulum**
1	5°	0
2	12°	0
3	4°	15

In the second scenario, reference tracking performance of the pendulum arm is tested for three different conditions with two distinct arm angles. Then, the comparison between physical and simulated RIP system in two different operation conditions is presented in the second section of this chapter.

Table 6.2. Simulation conditions for θ.

Condition	**Arm Angle θ (Degree)**	**Uncertainty (% of nominal value) Length of Pendulum**
1	28.5°(0.5 Rad)	0
2	57.3°(1 Rad)	0
3	22.9°(0.4 Rad)	15

- **Condition 1 (Simulation)**

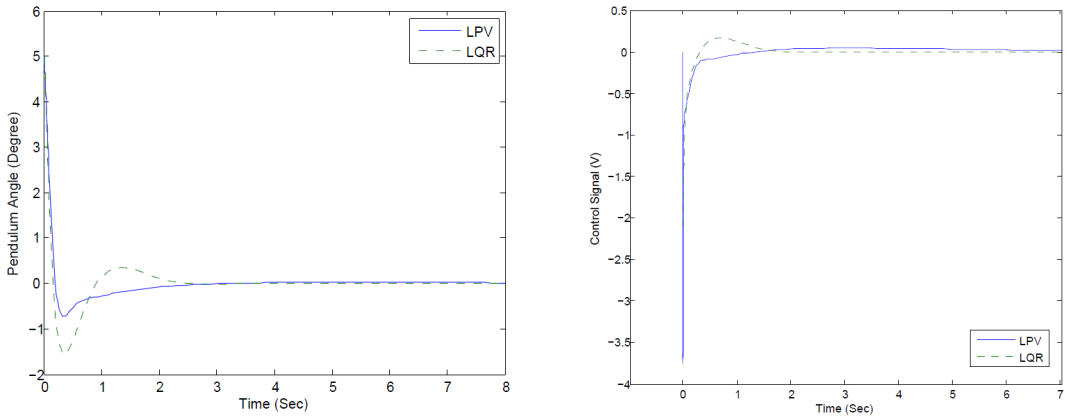

Figure 6.1. Pendulum angles α and control signals in condition 1.

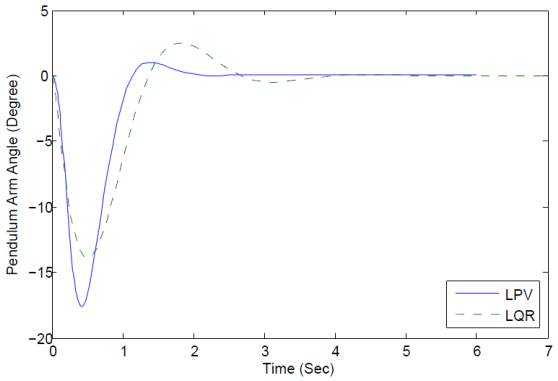

Figure 6.2. Arm angles θ in condition 1.

- **Condition 2 (Simulation)**

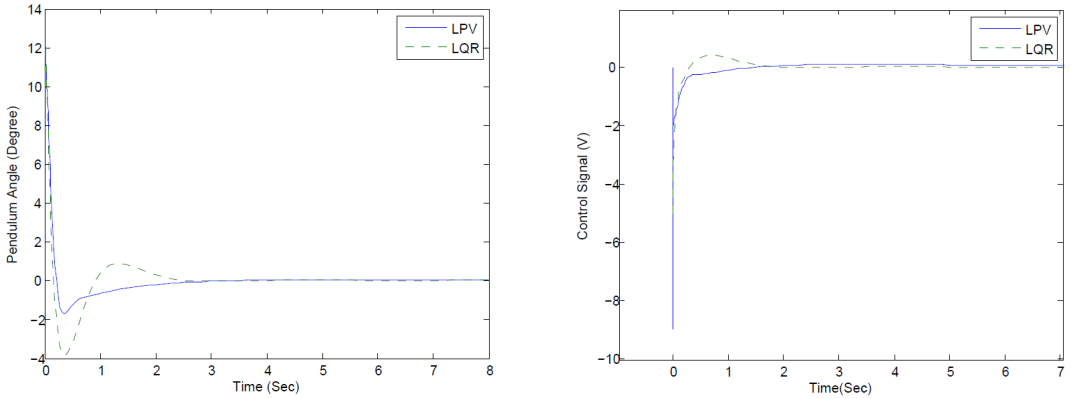

Figure 6.3. Pendulum angles α and control signals in condition 2.

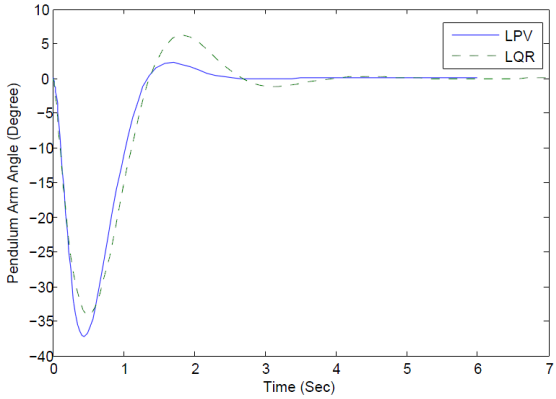

Figure 6.4. Arm angles θ in condition 2.

- **Condition 3 (Simulation)**

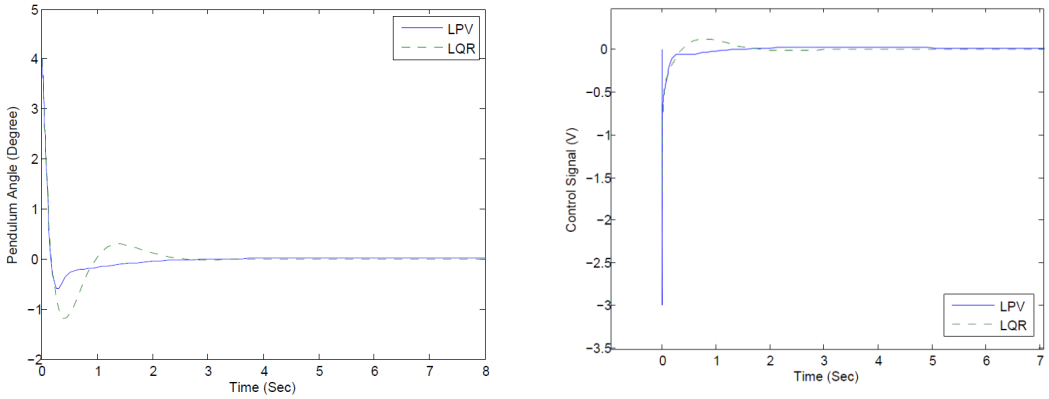

Figure 6.5. Pendulum angles α and control signals in condition 3.

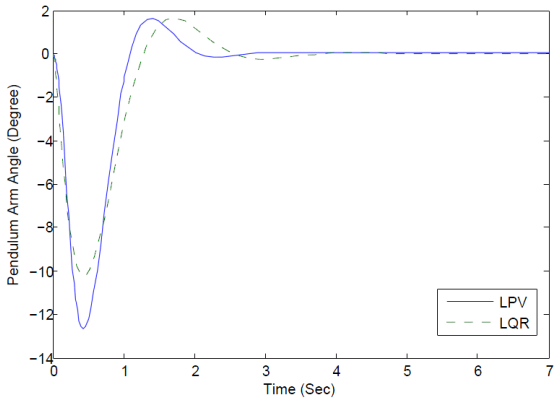

Figure 6.6. Arm angles θ in condition 3.

Pendulum Arm Reference Tracking

- **Condition 1 (Simulation)**

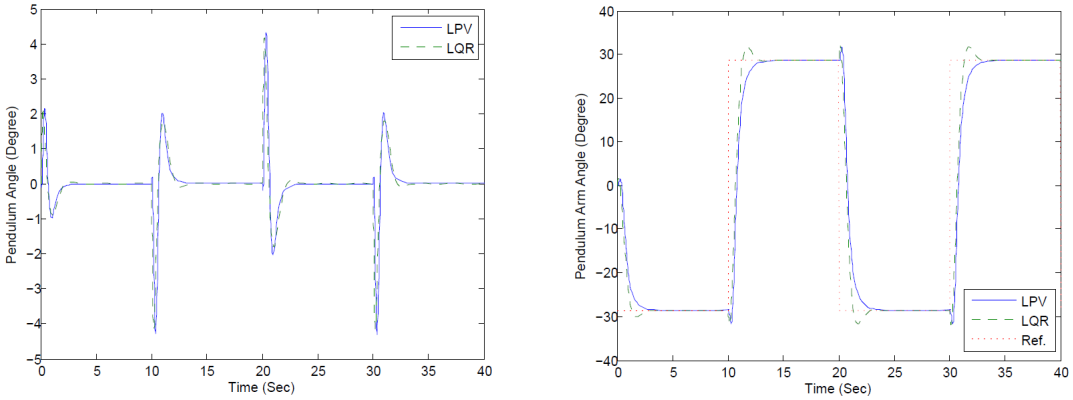

Figure 6.7. Pendulum angles α and arm angles θ in condition 1.

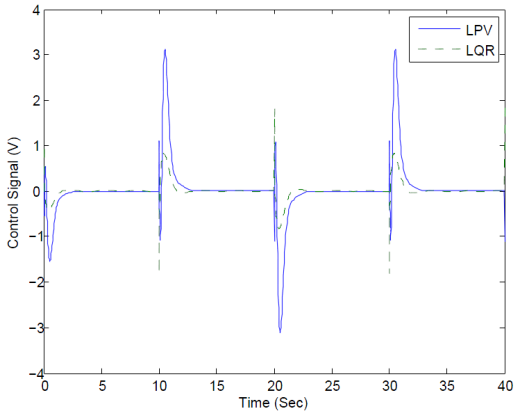

Figure 6.8. Control signal in condition 1.

- **Condition 2 (Simulation)**

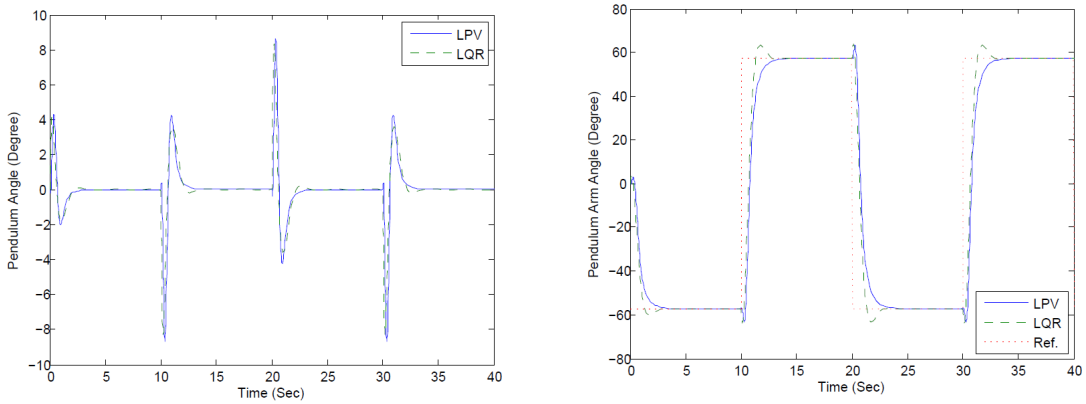

Figure 6.9. Pendulum angles α and arm angles θ in condition 2.

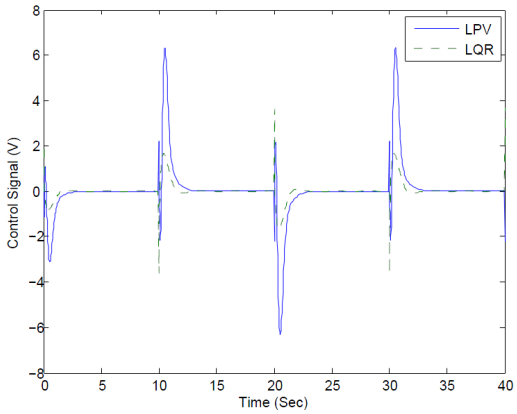

Figure 6.10. Control signal in condition 2.

- **Condition 3 (Simulation)**

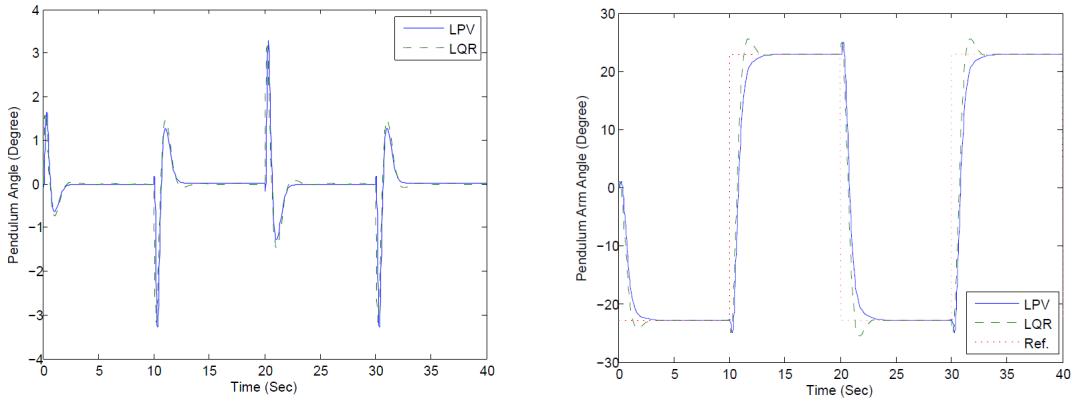

Figure 6.11. Pendulum angles α and arm angles θ in condition 3.

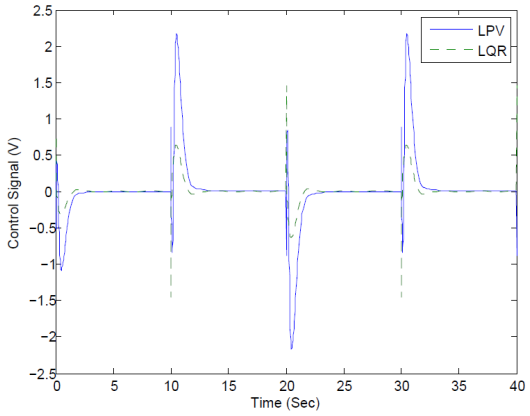

Figure 6.12. Control signal in condition 3.

6.2. LPV Controller Implementation on the Physical RIP System

Initially, when the designed LPV controller was applied on the experimental system, it was seen that the arm angle θ drifted. Therefore, the simulation results were contradicted by the initial experimental results. Each step in the controller design process has been checked to resolve the problem and it has been noticed that some entries in the LPV controller matrices C and D were found to be smaller than 1e-6. When these entries were rounded to zero, no θ drift occurred and the problem in the experiment was solved.

The designed LPV controller is implemented on the physical RIP system with 2 different conditions. Below table summarizes the conditions that the system tested in.

Table 6.3. Physical RIP system conditions for α.

Condition	**Initial Pendulum Angle α (Degree)**
1	5°
2	12°

- **Condition 1 (Experiment)**

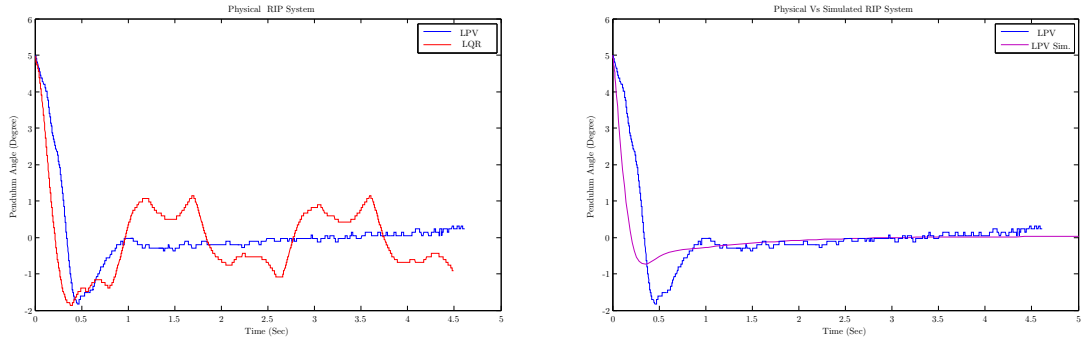

Figure 6.13. LPV and LQR controller results for the pendulum angle α.

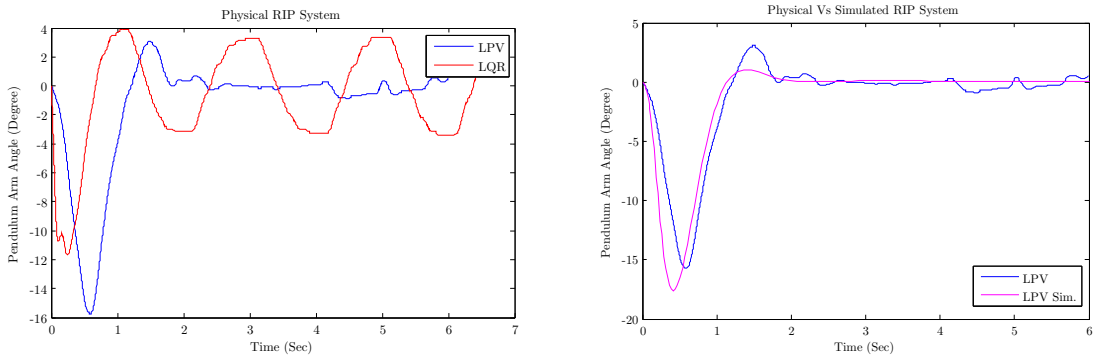

Figure 6.14. LPV and LQR controller results for the arm angle θ.

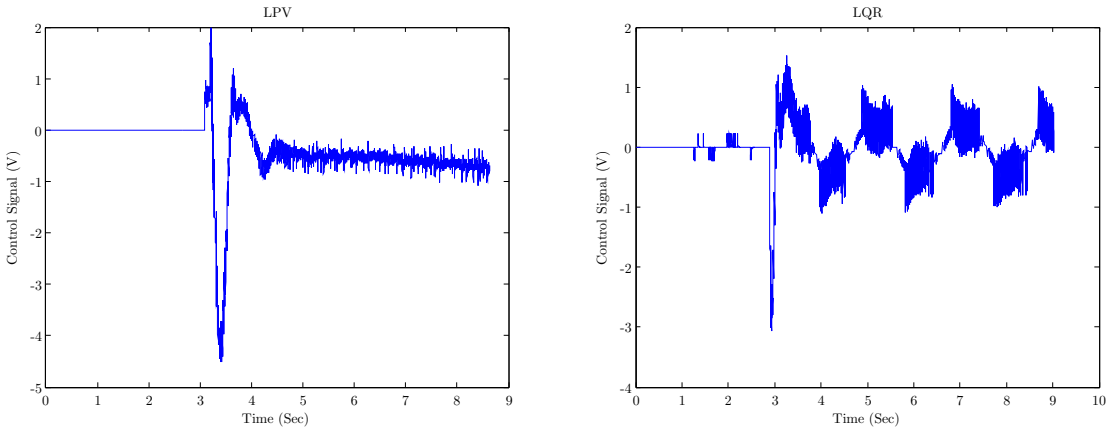

Figure 6.15. LPV and LQR control signals.

- **Condition 2 (Experiment)**

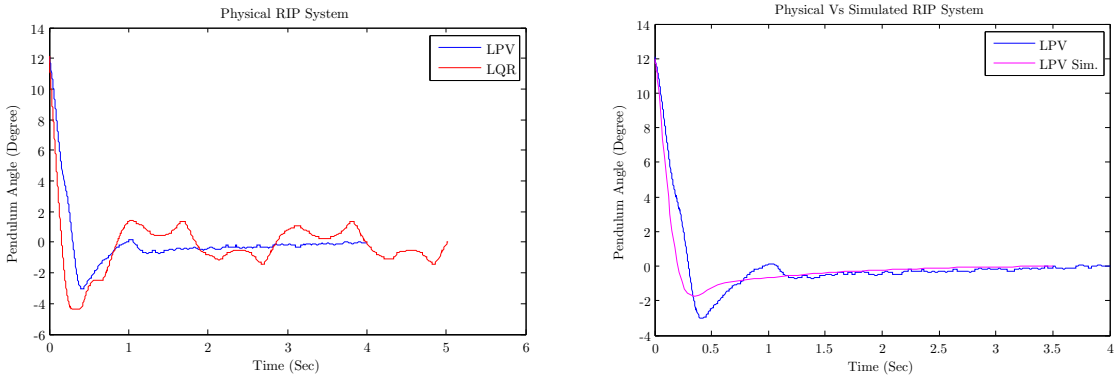

Figure 6.16. LPV and LQR controller results for the pendulum angle α.

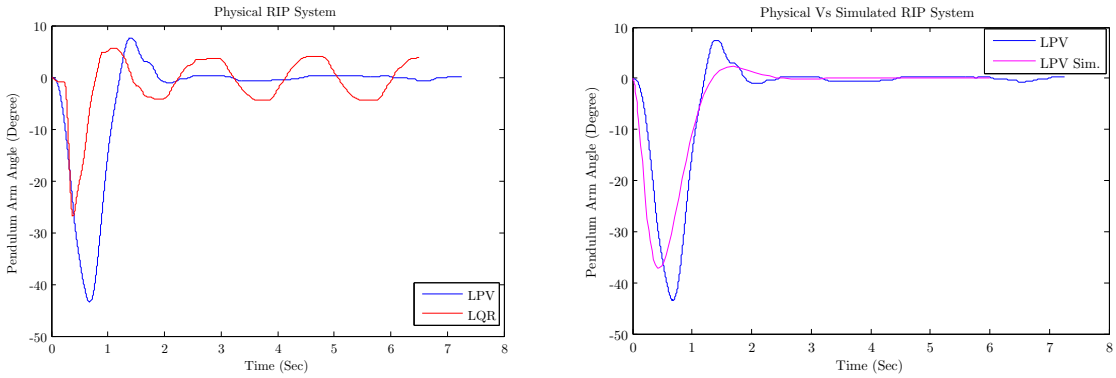

Figure 6.17. LPV and LQR controller results for the arm angle θ.

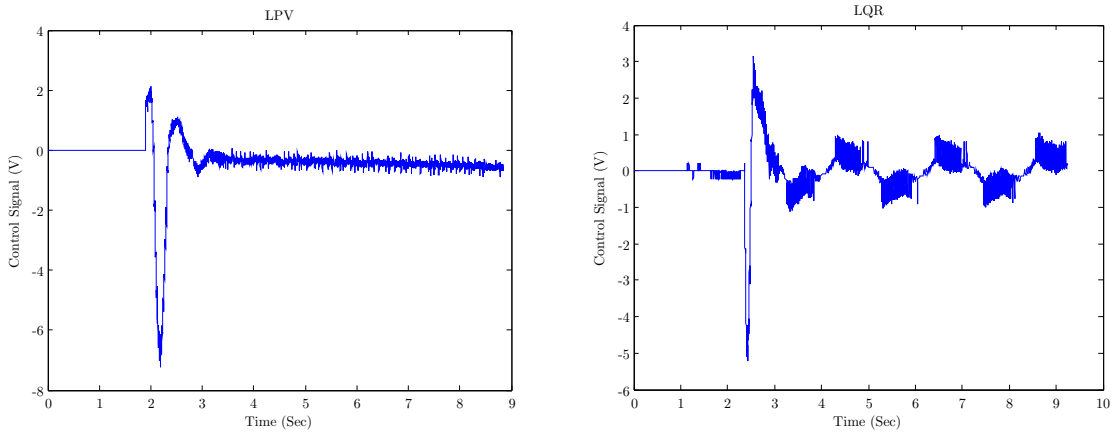

Figure 6.18. LPV and LQR control signals.

The designed LPV controller yields better results in simulation conditions 1, 2 and 3 about settling time and percent overshoot as compared to the LQR controller. However, the peak value of the control signal of the LPV controller is more than the peak value of control signal of the LQR controller in the simulation conditions 1, 2 and 3. Additionally, the LPV controller demonstrates a good performance in the experiment. In the experimental conditions 1,2; both the amount of control signal and the transient response characteristics of the LPV controller yields better results.

7. CONCLUSION

In this thesis, an LPV controller design is proposed for the Rotary Inverted Pendulum system, starting from the nonlinear equations. The LPV controller scheme is designed and illustrated both in the nonlinear simulation and the physical system environments. The LFR Toolbox [15] also made easier to build an LFT representation of the RIP model and minimized the model with its model reduction feature. For the existence of the controller, stability and performance LMI conditions of the desired closed loop system are developed and solved by IQC Synthesis Toolbox [14]. The closed control loop is asymptotically stable in the Lyapunov sense and guarantees the stability. Combining LFR Toolbox with IQC Synthesis Toolbox facilitated the controller design process. It is also observed from the simulation and the experimental results that the synthesized LPV controller gives better results in different operation conditions regarding the performance.

For the improvement of the results, the selected weighting functions can be more optimized. Hence, the LPV controller may produce smoother control signal and present better transient response. Also, increasing the model accuracy may improve the performance of the designed LPV controller especially in the experimental environment.

APPENDIX A: CONSTANTS IN MATHEMATICAL MODEL OF RIP

Constant Parameter	**Value**
$B_{eq}\,(Ns/m)$	4.0e-3
η_g	0.9
η_m	0.69
$g\,(m/s^2)$	9.8
$h\,(m)$	0.215
$J_{eq}\,(kg\,m^2)$	3.584e-3
$J_m\,(kg\,m^2)$	3.87e-7
K_g	70
$K_m\,(V\,s/rad)$	7.67e-3
$K_t\,(N\,m/A)$	7.67e-3
$L\,(m)$	0.1675
$m\,(kg)$	0.128
$r\,(m)$	0.215
$R_m\,(ohm)$	2.6

REFERENCES

1. *SRV02-Series Rotary Pendulum User Manual*, Revision 01, Quanser Inc., Canada.

2. Ling, K. V., P. Falugi, J.M. Maciejowski and L. Chisci, "Robust Predictive Control of the Furuta Pendulum", *Proc. of 15th IFAC World Congress on Automatic Control*, Barcelona, Spain, 2002.

3. Basturk, H. I., *Quasi-LPV Modeling And Control of Twin Rotor Multiple Input Multiple Output System*, MSc. Thesis, Boğaziçi University, 2008.

4. Yubai, K., K. Okuhara, and J. Hirai, "Gain-Scheduling Control of a Rotary Inverted Pendulum by Weight Optimization and H_∞ Loop Shaping Procedure", *Electrical Engineering in Japan*, Vol. 163, No. 2, 2008.

5. Boyd, S.P., L. Vandenberghe, *Convex Optimization*, Cambridge University Press, New York, USA, 2009.

6. Boyd, S.P. *et. al.*, *Linear Matrix Inequalities in System and Control Theory*, SIAM, New Jersey, USA, 1987.

7. Zhou, K., J. C. Doyle, and K. Glover, *Robust and Optimal Control*, Prentice-Hall, New Jersey, USA, 2002.

8. Dullerud, G. E. and F. Paganini, *A Course in Robust Control Theory*, Springer Verlag, New York, USA, 2000.

9. Juan, Z., C. Jie, and D. Lingxun, "Research on Control of Rotary Inverted Pendulum via Polytope Techniques", *2007 IEEE International Conference on Control and Automation*, pp. 2885-2889, 2007.

10. Roman, M., E. Bobasu, and D. Sendrescu, "Modelling of the Rotary Inverted Pendulum System", *2008 IEEE International Conference on Automation, Quality and Testing, Robotics*, pp. 141-146, 2008.

11. Bamieh, B. and L. Giarré, "Identification of Linear Parameter Varying Models", *International Journal of Robust and Nonlinear Control*, Vol. 12, No. 9, pp. 841-853, 2002.

12. Horn, R. A. *Matrix Analysis*, Cambridge University Press, New York, USA, 1990.

13. Scherer, C. W. and S. Wiland, *Lecture Notes: Linear Matrix Inequalities in Control*, `http://www.dcsc.tudelft.nl/cscherer/lmi/notes05.pdf`.

14. Delft University of Technology, *LPVMAD The IQC Synthesis Tool*, 2007.

15. Magni, J. F., *Technical Report: User Manual of the Linear Fractional Representation Toolbox*, Version 2.0, `http://www.ee.pucrs.br/gacs/new/disciplinas/ppgee/crobusto /referencias/lfr/lfrt_manual_v20.pdf`, 2006.

16. Scherer, C. W., "LPV Control and Full Block Multipliers", *Automatica*, Vol. 37, No. 3, pp. 361-375, 2001.

17. Diao, X., *Rotary Inverted Pendulum*, Course Project, New Mexico State University, 2006.

18. Kajita, S., M. Morisawa, K. Miura, S. Nakaoka, K. Harada, K. Kaneko, F. Kanehiro, K. Yokoi, "Biped Walking Stabilization Based on Linear Inverted Pendulum Tracking", *2010 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pp. 4489-4496, 2010.

19. The list of external solvers for the low-level numerical solution of optimization problems, `http://users.isy.liu.se/johanl/yalmip/pmwiki.php?n=Solvers.Solvers`.

20. Gahinet, P., A. Nemirovski, A. J. Laub, M. Chilali, *MATLAB, Robust Control Toolbox (LMILAB)*, Version r2011a, `http://www.mathworks.com/help/toolbox/robust/ug/f9-2283.html`, 1995

21. Chen, C. K., C. J. Lin, and L. C. Yao, "Input-State Linearization of a Rotary Inverted Pendulum", *Asian Journal of Control*, Vol. 6, No. 1, pp. 130-135, 2004.