Design, Simulation and Experimental Validation of a Robust Controller on Different Inverted Pendulum Systems

A Dissertation submitted
in partial fulfilment of the requirements
for the degree of

Master of Engineering
In
CAD/CAM Engineering

by
Ishan Chawla
Roll No.: 801584010

Under the guidance of
Dr. ASHISH SINGLA
Assistant Professor
Department of Mechanical Engineering
Thapar University, Patiala

DEPARTMENT OF MECHANICAL ENGINEERING
THAPAR UNIVERSITY, PATIALA

July, 2017

DECLARATION

I hereby declare that work done in this dissertation entitled, **" Design, Simulation and Experimental Validation of a Robust Controller on Different Inverted Pendulum Systems**" submitted towards partial fulfilment of requirement for award of **Master of Engineering degree** in **CAD/CAM Engineering** in **Department of Mechanical Engineering, Thapar University**, **Patiala**, is an authentic record of work carried out by me under the supervision and guidance of **Dr. Ashish Singla,** Assistant Professor of **Department of Mechanical Engineering**, **Thapar University**, **Patiala**.

This matter embodied in this report has not been submitted in part or full to any other university or institute for the award of any degree.

DATE: 7, July, 2017 **Ishan Chawla**

This is to certify that above declaration made by the student concerned is correct to the best of my knowledge and belief.

Dr. Ashish Singla
Assistant Professor
Department of Mechanical Engineering
Thapar University, Patiala

ACKNOWLEDGEMENT

Through this, I would like to acknowledge all the inspirational events and people who concerned to my journey from childhood to Master of Engineering research curriculum. It all started from my home, as my mother **Mrs. Dimple Chawla** is a remarkable science and mathematics teacher where as my father **Mr. Anil Kumar Chawla** is a passionate civil engineer. They are the prior reason behind my profound foundation of science and related subjects. Later in higher secondary, physics become my favourite subject. It was my love for physics that made me choose Mechanical as my discipline of the Bachelor course. Apart from academics, I became part of technical group named **Technical Educational and Development Association (TEDA)**, which was a turning point in my life. It promoted the desire of innovation in me. I have participated in many robotics projects with the collaboration of TEDA where I developed keen interest in Robotics and Control. Further, proceeding with my post graduate courses, I came to know that there is a faculty in the research domain of Robotics & Control Engineering who became my supervisor in research. I would like to express my reflective adoration and gratitude to my mentor, **Dr. Ashish Singla** for his amicable guidance, beneficial suggestions and empowering inspiration in the concerned work. It has been blissful experience to learn the professional and soft skills under his supervision which overall developed my personality. Furthermore, I would also like to acknowledge **Dr. Vikram Chopra** for guiding me with his new ideas and continuously motivating me to achieve my research goals.

I would also like to express my gratitude to **Mr. Jyotindra Narayan** for introducing me to the soft computing technique named ANFIS which helped me throughout this dissertation. Furthermore, I would like to thank **Mr. Sumit Mehta**, **Mr. Aakash Soni** and **Mr. Lalit Kumar** for being an excellent support in my bad times.

I would also like to thank **Miss Anmol Gupta** and **Miss Amandeep Sidhu** for supporting and motivating me whenever I felt demotivated. Being with them, makes me realize that I can do anything. At the end, i would like to express my sincere gratitude to all those who directly or indirectly helped me for the successful completion of this dissertation.

Ishan Chawla

ABSTRACT

From the last five decades, inverted pendulum systems have been considered as a benchmark problem in the control literature due to their inherit nature of instability, nonlinearity and underactuation. Moreover, inverted pendulum dynamics closely resembles many real-time systems. Due to its applications in teaching and research experiments, a wide variety of inverted pendulum (IP) systems exist. To enhance the wealth of this robotic benchmark, this dissertation focuses on modeling and control of different inverted pendulum systems such as *x*-*y*-*z*, *x*-*y*, *x*-*z*, *x* and rotary inverted pendulum system. Dynamic mathematical modeling of different inverted pendulum systems are performed in which each system is linearized about its unstable equilibrium point i.e. vertically upright pendulum position and a state-space representation of each system is formed. Conventional modern controllers such as pole placement and LQR controllers are designed and implemented in real-time on rotary inverted pendulum. Further, to encounter the robustness issue in LQR controller, a novel robust LQR based ANFIS controller is proposed and validated on different inverted pendulum systems. Comparative simulation as well as experimental analysis of proposed controller with conventional controllers is done to validate the robustness of the proposed controller to pendulum mass, in which the proposed controller gives better performance in achieving lesser overshoot and settling time along with better robustness properties. Later, swing up problem of rotary inverted pendulum system is achieved using a nonlinear energy controller.

Keywords: Inverted Pendulum; Robust Control; ANFIS; Stabilization; Modeling; Real-time; Swing-up; LQR Controller; Pole Placement Controller.

Contents

List of Figures

List of Tables

List of Abbreviations

IP	:	Inverted Pendulum
LQR	:	Linear Quadratic Regulator
ANFIS	:	Adaptive Neural Fuzzy Inference Systems
RLQR-ANFIS	:	Robust LQR based ANFIS
IFAC	:	International Federation of Automatic Control
CNS	:	Central Nervous System
VTOL	:	Vertical Take-off and Land
XIP	:	*x* Inverted Pendulum
XYIP	:	*x*-*y* Inverted Pendulum
XZIP	:	*x*-*z* Inverted Pendulum
LQG	:	Linear Quadratic Gaussian
SDRE	:	State Dependent Riccati Equation
PID	:	Proportional-Integral-Derivative
MIMO	:	Multiple Input Multiple Output
DOF	:	Degrees of Freedom
ISMC	:	Integral Sliding Mode Controller
KE	:	Kinetic Energy
PE	:	Potential Energy
RIP	:	Rotary Inverted Pendulum
CCW	:	Counter-Clockwise
PPC	:	Pole Placement Controller
MF	:	Membership Function
PSO	:	Particle Swarm Optimization
GA	:	Genetic Algorithm

List of Symbols

t	:	time
u	:	input to plant
K_P	:	proportional coefficient
K_i	:	integral coefficient
K_d	:	derivative coefficient
e	:	error
e_{ss}	:	steady state error
M_p	:	maximum percentage overshoot
ξ	:	damping factor
t_s	:	settling time
ω_d	:	damping frequency
ω_n	:	natural frequency
k	:	pole separation factor
$\boldsymbol{K}$	:	LQR controller gain
$\boldsymbol{Q}$	:	state weighing matrix
$\boldsymbol{R}$	:	control cost matrix
$\boldsymbol{A}$	:	system matrix
$\boldsymbol{B}$	:	input matrix
$\boldsymbol{C}$	:	output matrix
$\boldsymbol{D}$	:	feedthrough matrix
q	:	generalized coordinates
$\mathcal{L}$	:	Lagrange operator
R	:	Rayleigh's dissipation function
x	:	cart displacement in *x* direction
y	:	cart displacement in *y* direction
z	:	cart displacement in *z* direction
$\dot{x}$	:	cart velocity in *x* direction
$\dot{y}$	:	cart velocity in *y* direction
$\dot{z}$	:	cart velocity in *z* direction
$\ddot{x}$	:	cart acceleration in *x* direction
$\ddot{y}$	:	cart acceleration in *y* direction

$\ddot{z}$: cart acceleration in *z* direction

x_P : pendulum centre of mass displacement in *x* direction

y_P : pendulum centre of mass displacement in *y* direction

z_P : pendulum centre of mass displacement in *z* direction

$\dot{x}_P$: pendulum centre of mass velocity in *x* direction

$\dot{y}_P$: pendulum centre of mass velocity in *y* direction

$\dot{z}_P$: pendulum centre of mass velocity in *z* direction

l : distance between cart hinge point to the centre of mass of the pendulum

M : cart mass

m : pendulum mass

g : gravity

θ : pendulum angle from the vertical axis, as seen in *x*-*z* plane

$\dot{\theta}$: pendulum angular velocity from the vertical axis, as seen in *x*-*z* plane

$\ddot{\theta}$: pendulum angular acceleration from the vertical axis, as seen in *x*-*z* plane

ϕ : pendulum angle from the vertical axis, as seen in *x*-*z* plane

$\dot{\phi}$: pendulum angular velocity from the vertical axis, as seen in *x*-*z* plane

$\ddot{\phi}$: pendulum angular acceleration from the vertical axis, as seen in *x*-*z* plane

F_x : control force on the cart in *x* direction

F_y : control force on the cart in *y* directions

F_z : control force on the cart in *z* directions

S_θ : $\sin(\theta)$

C_θ : $\cos(\theta)$

S_ϕ : $\sin(\phi)$

C_ϕ : $\cos(\phi)$

m_p : mass of pendulum for rotary inverted pendulum

J_p : pendulum inertia

L_p : length of the pendulum

B_p : pendulum damping coefficient

J_r : rotary arm inertia

L_r : length of the rotary arm

B_r : viscous damping coefficient

J_m : motor inertia

R_m	:	terminal resistance
K_m	:	motor back emf constant
K_t	:	torque constant
L_m	:	rotor inductance
$\boldsymbol{x}$	:	state vector
$\boldsymbol{y}$	:	output vector
$\boldsymbol{M}$	:	controllability test matrix
$\boldsymbol{T}$	:	transformation matrix
J	:	cost function
O_i^1	:	output of node i for layer *1*
$\mu_{A_i}(x)$	:	membership value of input variable x for fuzzy membership function a_i.
c_i	:	centre position for subsequent membership function
a_i	:	corresponding half width
b_i	:	slopes at the distinguished crossover points (where MF value is 0.5)
w_i	:	firing strength
$\overline{w_i}$	:	normalized firing strength
$\boldsymbol{x_{des}}$	:	desired state vector
$\boldsymbol{e}$	:	error vector
$\boldsymbol{K^m}$	:	LQR gain for system with pendulum mass m
$\boldsymbol{A^m}$	:	system matrix for pendulum mass m
$\boldsymbol{B^m}$	:	control input matrix for pendulum mass m
$\boldsymbol{u_i^m}$	:	input vector for i^{th} actuator corresponding to pendulum mass m
e_α	:	error in state variable α
e_x	:	error in state variable x
$e_{\dot{x}}$	:	error in state variable $\dot{x}$
e_y	:	error in state variable y
$e_{\dot{y}}$	:	error in state variable $\dot{y}$
e_z	:	error in state variable z
$e_{\dot{z}}$	:	error in state variable $\dot{z}$
e_θ	:	error in state variable θ
$e_{\dot{\theta}}$	:	error in state variable $\dot{\theta}$
e_ϕ	:	error in state variable ϕ

$e_{\dot{\phi}}$	:	error in state variable $\dot{\phi}$
T_s	:	settling time of pendulum angle
α_{max}	:	maximum overshoot in pendulum angle of rotary inverted pendulum
θ_{max}	:	maximum overshoot in rotary arm angle of rotary inverted pendulum
x_{max}	:	maximum overshoot in cart displacement in x direction
y_{max}	:	maximum overshoot in cart displacement in y direction
z_{max}	:	maximum overshoot in cart displacement in z direction
$F_{x_{max}}$	:	maximum overshoot in control input in x direction
$F_{y_{max}}$	:	maximum overshoot in control input in y direction
$F_{z_{max}}$	:	maximum overshoot in control input in z direction
ϕ_{max}	:	maximum overshoot in ϕ
E	:	pendulum energy
E_r	:	reference pendulum energy
μ	:	tunable swing-up controller gain
u_{max}	:	maximum control input
$sat_{u_{max}}$	:	function that saturates the control signal at the maximum acceleration of the pendulum pivot, u_{max}
u_{stab}	:	controller input for stabilization control
u_{swing}	:	controller input for swing up control

List of Publications

[1] Ishan Chawla, Vikram Chopra and Ashish Singla, "**Robust LQR Based ANFIS Control of *x*, *x*-*y* and *x*-*z* Inverted Pendulums**", *Journal of Vibration and Control,* 2017. [**Paper Communicated**]

[2] Ishan Chawla, Vikram Chopra and Ashish Singla, "**Control of *x*-*y*-*z* Inverted Pendulum Using Robust LQR Based ANFIS Controller**", *Iranian Journal of Fuzzy Systems.* [**Paper Communicated**]

[3] Ishan Chawla and Ashish Singla, "**Real-Time Control of Rotary Inverted Pendulum using Robust LQR Based ANFIS Controller**", *International Journal of Nonlinear Sciences and Numerical Simulation.* [**Paper Communicated**]

[4] Ishan Chawla and Ashish Singla, "**System Identification of an Inverted Pendulum Using Adaptive Neural Fuzzy Inference System**", *3rd International and 18th National Conference on Machines and Mechanisms, 2017*. [**Paper Communicated**]

Chapter 1

Introduction

In our childhood or at some point in our lives almost all of us have attempted to balance a broomstick on the palm of our hand in a similar way as shown in Figure 1.1. The broomstick can be thought of as an inverted pendulum with the difference being that the broomstick is free to move in a three-dimensional space while a pendulum, in this study, is hinged to a cart and can only move in a linear plane as shown in Figure 1.2. Just like the broomstick, an inverted pendulum is a highly unstable system. Proper control force must be appropriately applied to keep the pendulum balanced and upright.

Figure 1.1: Balancing of stick [W1]

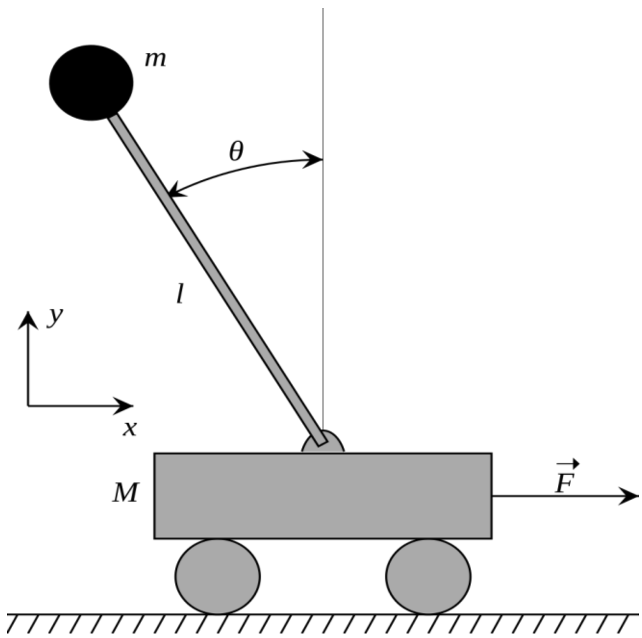

Figure 1.2: Linear Inverted Pendulum [W2]

In 1990, the International Federation of Automatic Control (IFAC) Theory Committee published a set of real world control problems to validate new and existing control methods, called benchmark problems [1]. Control of an inverted pendulum was one of them. Although it has simple structure it is one of the most difficult systems to control. This difficulty arises because of its underactuated nature where the numbers of actuators to control the system are less than the degrees of freedom of the system Apart from that, the system is highly nonlinear and unstable in nature thus making the system difficult to control.

The pendulum in this system has two equilibrium points – stable equilibrium and the unstable equilibrium point. The stable equilibrium point is when the pendulum is hanging vertically downward i.e. potential energy of the pendulum is minimum. The unstable equilibrium point is when the pendulum is at vertically upright position i.e. potential energy of the pendulum is maximum. Based on these equilibrium points, three types of control problems related to inverted pendulum systems are found in literature, given as:

1. **Swing-up control problem** – In this problem, pendulum is controlled to swing up from hanging stable equilibrium position to vertically upright unstable position. It is a highly nonlinear problem [2, 3].
2. **Stabilization control problem** – In this problem, pendulum is stabilized at vertically upright position which is an unstable position. It is a linear problem [4].
3. **Tracking control problem** – In this problem, cart or rotary arm is made to follow the desired trajectory while stabilizing the pendulum at vertically upright position. It can be a linear or nonlinear problem [5].

Inverted pendulum systems can be categorized into two major types – the rotary inverted pendulum system, and the linear inverted pendulum systems. The rotary inverted pendulum consists of rotary arm on which pendulum is hinged while the linear inverted pendulum systems consist of cart on which pendulum is hinged. The cart can move in *x*-, *y*-, and *z*-direction depending on the type of linear inverted pendulum system. The major difference between the two systems is that the linear inverted pendulum systems generally have a finite track length that needs to be taken into account while the rotary inverted pendulum is generally free to rotate completely without any restrictions.

1.1 Applications

Since the shape of inverted pendulum resemble many different real world systems, hence the inverted pendulum has numerous applications. Also based on inverted pendulum stabilization principle, many robotics control strategies are emerged. Some of the potential applications of inverted pendulum are given below:

- **Control of underactuated systems:** Underactuated mechanical systems are the systems that have less control inputs than the degree of freedom. Inverted pendulum is a typical example of underactuated system and thus, the controllers designed for inverted pendulum can be used for other underactuated systems [6].
- **Mobile wheeled inverted pendulums:** Based on inverted pendulum stabilization principle, many different mobile wheeled systems are designed and implemented.

Segways [7], unicycle, hoverboard [8] are some of the popular commercially available systems.

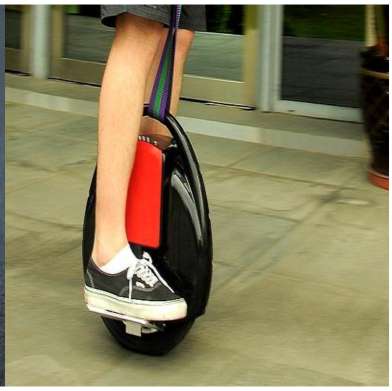

Figure 1.3: Segways [W3]   Figure 1.4: Hoverboard [W4]   Figure 1.5: Unicycle [W5]

- **Stabilization of ships and rockets:** The dynamics of inverted pendulum resembles rockets and ships and thus, stabilization controller for inverted pendulum can be used to control ships and rockets [9].

Figure 1.6: Rockets [10]

Figure 1.7: Ships [10]

- **Design of earthquake resistant building:** Dynamics of building during an earthquake resembles the dynamics of inverted pendulum [11]. Thus, earthquake resistant buildings can be designed accordingly.

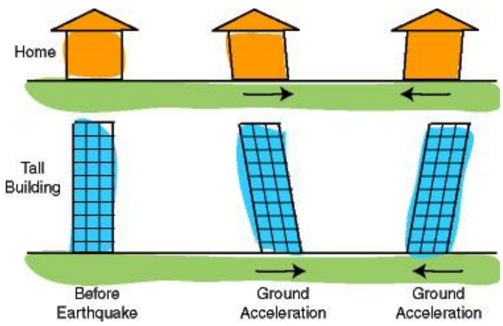

Figure 1.8: Design of earthquake resistant buildings [10]

- **Adequate model of human standing still:** The dynamics of the inverted pendulum can be considered similar to an adequate model of human standing still [12]. The ability of the human brain to maintain stability while standing straight is of great significance for the daily activities of people. It's amazing how our central nervous system (CNS) registers the pose and changes in the pose of the human body, and activates muscles in order to maintain balance.

- **Dynamic walking**: Human walking can be considered as combination of simple pendulum and inverted pendulum. Hence to obtain dynamic walking in humanoid robots, it is important to study inverted pendulum [13].

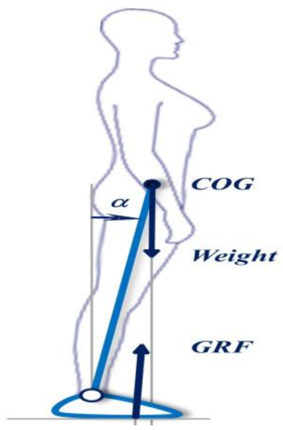

Figure 1.9: Human standing still [10]

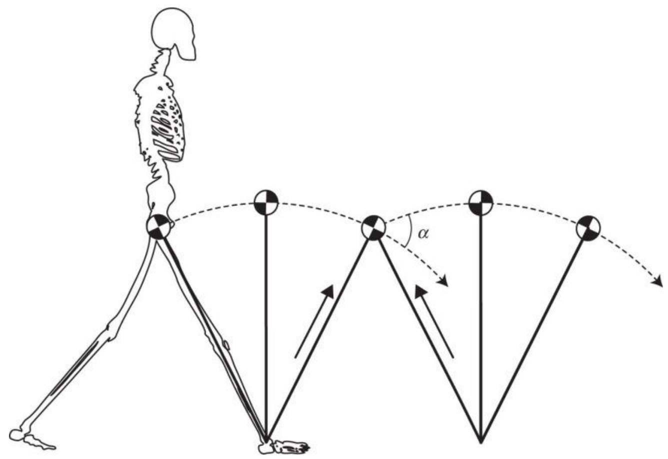

Figure 1.10: Dynamic walking [W6]

- **Simulation of Dynamics of robotic arm:** The inverted pendulum problem resembles the control systems that exist in robotic arms. The dynamics of inverted pendulum simulates the dynamics of robotic arm in the condition when the centre of pressure lies below the centre of gravity for the arm so that the system is also unstable. Robotic arm behaves very much like inverted pendulum under this condition.

1.2 Scope of the Thesis

From the last few decades, inverted pendulum is considered as the most popular milestone in the control field [1]. Due to its inherit nonlinearity, instability and underactuation, it is frequently considered as a benchmark problem to verify the effectiveness of emerging control strategies and techniques. Moreover, the dynamics of inverted pendulum resembles many real-time systems such as underactuated robotic systems [6], rockets [9], vertical take-off and land (VTOL) aircrafts [14], segway [7], hoverboard [8], humanoid robots [13], buildings [11] etc. Due to its applications in teaching and research experiments, a wide variety of inverted pendulum (IP) systems exist, such as *x* inverted pendulum (commonly called as linear inverted pendulum) [15], *x*-*y* inverted pendulum (commonly called as spherical inverted pendulum) [16], *x*-*z* inverted pendulum [17], acrobat [18], pendubot [19], inertia wheel pendulum [20] and rotary inverted pendulum [21] etc.

The *x* inverted pendulum (XIP) is the most popular inverted pendulum, where cart can only move in *x* direction to control the pendulum pivoted by a pin joint to the cart. It is a single input multiple output (SIMO) system with one degree of under actuation. Even though, the structure of XIP is simple, its control is still a challenging task. The *x*-*y* inverted pendulum (XYIP) is a classification of spherical inverted pendulum, where cart moves in the *x*-*y* horizontal plane to control the pendulum pivoted by a universal joint to the cart. The XYIP is highly nonlinear in nature due to which control of XYIP is a complex task. However, XYIP represents a more realistic model of many practical systems such as rockets, buildings, humanoid robots which marks its importance in control literature. The *x*-*z* inverted pendulum (XZIP) is a generalized case of XIP, which moves in both horizontal and vertical planes to control the pendulum. Since XZIP can also move in vertical, it represents a realistic model of practical systems such as planer vertical take-off and land aircrafts [22], rocket, buildings during earthquake.

The *x*-*y*-*z* inverted pendulum can be seen as a human trying to balance a stick on his palm, where his hand acts as a cart which can move in 3-dimensional space while the stick acts as a pendulum. The system has 5 degrees of freedom with 3 control inputs which acts on the cart, while the pendulum is indirectly controlled. It is an underactuated system with 2 degrees of underactuation. It can also be considered as a combination of *x*-*y* and *x*-*z* inverted pendulums which makes it a more challenging system as the pendulum can fall off in lateral direction too (as in *x*-*y* inverted pendulum) along with that cart can also move in vertical plane against gravity (as in *x*-*z* inverted pendulum). Thus, the dynamics of the system is highly complex and nonlinear, which makes the mathematical modeling and control of this system a challenging task.

The rotary inverted pendulum is one of the conventional inverted pendulum systems. It consists of a pendulum hinged to a rotary arm. The rotary arm is actuated with an actuator while the pendulum is indirectly actuated with rotary arm, thus, making it an underactuated system. The system was first introduced by Furuta [21] and hence it is also called as Furuta pendulum. This system has an advantage of infinite track length unlike linear inverted pendulum systems.

In literature, three kinds of control problems exist, i.e., swing up control [3]; stabilization control [4] and tracking control [5], out of these stabilization and tracking control are extensively studied due to their application in many practical systems. Many different control strategies are discussed for stabilization and tracking control of inverted pendulum such as linear quadratic regulator (LQR) controller [23], pole placement controller [24], PID controller [25], linear quadratic gaussian (LQG) controller [26], state dependent riccati equation (SDRE) based controller [27], sliding mode controller [28] and fuzzy controller [29] etc. The most robust control strategies are being followed by sliding mode controller and fuzzy controller while the most optimal strategy being LQR controller. The sliding mode controller has a problem of discontinuity and chattering whereas fuzzy controllers are difficult to tune and requires expert knowledge of system to tune it. Also for a multi variable system problem of rule explosion exists in fuzzy controllers. To resolve these problems in fuzzy controller, recently, the use of LQR based adaptive neural fuzzy inference system (ANFIS) has been suggested [30] but robustness of the controller is not addressed in the work.

The contributions of this dissertation are:

1. Accurate mathematical modeling of different inverted pendulum systems i.e. *x*-*y*-*z*, *x*-*y*, *x*-*z*, *x* and rotary have been done using Euler-Lagrange approach.
2. Fine tuning of conventional modern controllers such as pole placement controller and LQR controller for rotary inverted pendulum
3. Design of a robust LQR based ANFIS (RLQR-ANFIS) controller which is robust to parametric uncertainties. The proposed controller has been implemented on different inverted pendulum systems i.e. *x*-*y*-*z*, *x*-*y*, *x*-*z*, *x* and rotary inverted pendulum to verify its effectiveness on different complex systems. The robustness of the controlled is validated by varying the pendulum mass and then analysing its performance parameters. The results obtained are compared with conventional controllers like LQR and PID controller, where the percentage reduction in performance parameters have been analysed for different cases.
4. A swing-up controller based on energy method is designed and tuned so as to obtain the swing up of pendulum. Later, a hybrid control algorithm is designed which a

switch from swing-up controller to conventional stabilization controller (LQR controller) as the pendulum angle reaches in the described range.

1.3 Objectives of the Thesis

Based on the literature review as expressed in chapter 2, following objectives are set in this dissertation:

1. Formation of generalized (XYZIP) mathematical model of linear inverted pendulum systems.
2. Mathematical modeling of rotary inverted pendulum system along with its validation with real-time system.
3. Real-time stabilization and swing-up control of RIP system using conventional controller.
4. Robust Controller design and its validation on different inverted pendulum system.

1.4 Organization of the Thesis

The present dissertation is organized as follows – Chapter 2 represents the detailed literature survey on the control of inverted pendulum systems. The varieties of controllers such as classical, modern, intelligent and robust controllers are discussed with the perspective of inverted pendulum. The conclusion from the literature survey is summarized at the end of this chapter.

Chapter 3 deals with the dynamic mathematical modeling of inverted pendulum systems. Firstly, basic theory of different mathematical modeling techniques is discussed. Later, one of these techniques i.e. Euler-Lagrangian approach is used to derive dynamic mathematical model of different inverted pendulum systems such as *x*, *x*-*y*, *x*-*z*, *x*-*y*-*z* and rotary inverted pendulum. Each system is linearized about its unstable equilibrium point i.e. vertically upright pendulum position and a state-space representation of the system is formed. Apart from that, the mathematical model of rotary inverted pendulum is compared with real-time rotary inverted pendulum system and comparative results are presented.

In Chapter 4, two of the modern control techniques i.e. pole placement control and LQR control, for stabilization of inverted pendulum about unstable point is discussed. Basic theories of these controllers are discussed followed by their design approach for rotary inverted pendulum. Experimental and simulation results of rotary inverted pendulum mounted with each controller are obtained and comparative analysis of performance is carried out in this chapter.

Chapter 5 discusses robust control of inverted pendulum systems. A novel robust LQR based ANFIS (RLQR-ANFIS) controller is discussed in this chapter. At first, the design procedure of RLQR-ANFIS is discussed in detail, later, the proposed controller is implemented on different inverted pendulum systems i.e. as x, x-y, x-z, x-y-z and rotary inverted pendulum. To verify controller robustness, pendulum mass is varied within a bounded range and performance parameters of response are analysed. The proposed controller is further compared with LQR and PID controller to analyse its robustness with respect to conventional controllers.

Chapter 6 discusses the swing-up problem of inverted pendulum system. Energy controller is used to tackle this nonlinear problem. The controller is designed and implemented on real-time rotary inverted pendulum system. Further, the controller is combined with conventional stabilization controller i.e. LQR controller to obtain a hybrid control scheme, which is also implemented on real-time rotary inverted pendulum system.

Finally in Chapter 7, Closing remarks are presented. In that, contributions of this dissertation along with conclusion from the work are presented. Further, future directions of the current work are also outlined.

Chapter 2

Literature Review

2.1 Introduction

In this chapter, literature review of different control strategies for inverted pendulum is discussed. Since 1950, the inverted pendulum system is the benchmark for teaching and studying different types of control strategies [31]. Because of its wide range of application and popularity in control systems many types of controllers are available in literature to address inverted pendulum system. Boubaker [32] have briefly discussed the variety of controllers available for cart pendulum systems in a survey article. However, most of the controllers available in published literature have been tested in simulation environment only, not demonstrated on real-time experimental setups. The simulation results are often found to be different when compared with real time experimental results. Moreover, most of the simulations ignore the physical restrictions like constrained cart length, effect of friction and addition dynamics like actuators, gears, timing pulley etc. In this review, many different kinds of controller are discussed. Later, a conclusion from the literature review is presented at the end of this chapter.

2.2 Bang–Bang Controller

Bang–Bang controller also called on–off controller is a feedback controller that switches abruptly between two states to control a certain process/plant. They are often used to control a plant which can take a binary input, for example a furnace which can be either completely off or completely on.

Pros:

- Easy to tune
- Easy to code
- Robust to parametric uncertainties of control system [33]
- Bang–Bang control frequently used in minimum time problems. For example, if it is desired to stop a car time at a certain position sufficiently far ahead of the car in the shortest possible, the solution is to drive at maximum acceleration until the

unique switching point, and then apply brakes at maximum force to come to rest exactly at the desired position

Cons:

- Requires Complex Calculations
- Inaccurate
- Consumes lot of effort or energy
- Requires knowledge of the system

Miori *et al.* [33] used bang-bang controller to control linear inverted pendulum in real time and obtained that controller is robust to parametric uncertainties of control system. Later, Furuta *et al.* [34] applied bang-bang controller for swing up of real time rotary inverted pendulum.

2.3 Fuzzy Controller

The term "fuzzy" refers to the fact that the logic involved can deal with concepts that cannot be expressed as the "true" or "false" but rather as "partially true".

Pros:

- Simple Structure.
- Fast computation speed.
- Very popular methods for both the swing-up and stabilization of inverted pendulum.

Cons:

- They lack the specification of stability conditions.
- Difficult to tune.
- Rule explosion for multiple input multiple output (MIMO) systems.

Fuzzy logic was first proposed by Lotfi *et al.* [35]. Yubazaki *et al.* [36] used fuzzy controller for swing up the pendulum and then stabilize the whole system in 2.0 seconds.

2.4 Proportion Integral Derivative (PID) Control

A proportional integral derivative (PID) controller is a closed loop feedback controller. It continuously calculates an error value i.e. the difference between a desired value and a measured process variable.

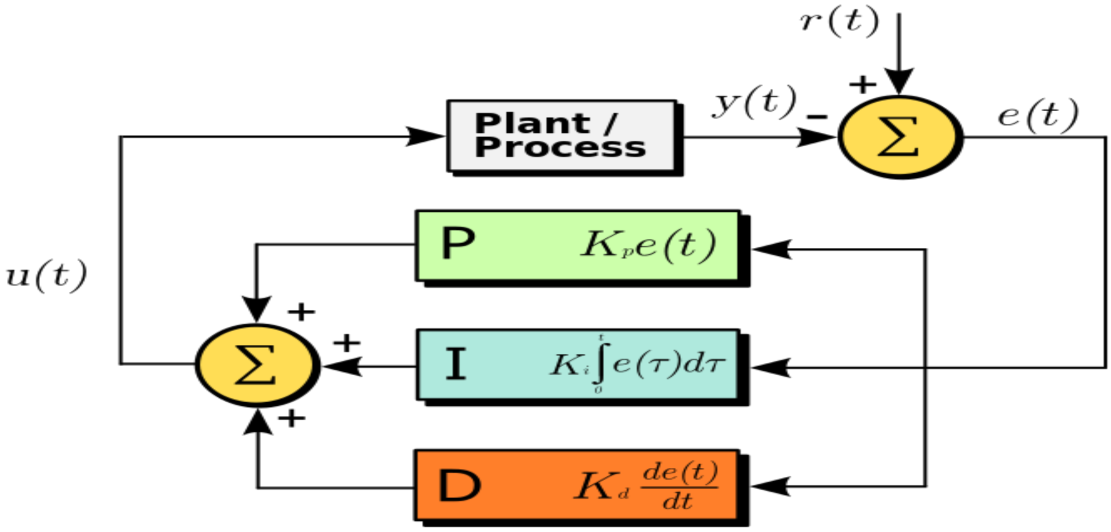

Figure 2.1: PID controller [W7]

The controller attempts to minimize the error over time by adjustment of a control variable to a new value determined by a weighted sum.

$$u(t) = K_P e(t) + K_i \int_0^t e(\tau)d\tau + K_d \frac{de(t)}{dt} \tag{2.1}$$

Where, K_p, K_i and K_d represents the coefficients for the proportional, integral, and derivative terms, respectively (sometimes denoted P, I, and D). PID Controller is a combination of three controllers

1. Proportional (P) Controller
2. Integral (I) Controller
3. Derivative(D) Controller

Each controller is briefly discussed below:

Proportional (P) Controller

- Output of controller is proportional to error value.
- P accounts for present values of the error.
- Fast response.
- Cannot counter static loading.
- Steady state error cannot be brought to zero using this controller alone.
- Steady state error decreases as K_p increases.

- e_{ss} (Steady state error for 2nd degree system) $= \frac{2\delta}{w_n K_p}$ (2.2)
- Hence for e_{ss} to be zero K_p will have to be infinity which is practically not possible as K_p represents a gain which could physically be a motor coefficient and no motor can give infinite torque.

Integral Controller

- Controller output is proportional to integration of error until the time t .Hence we can conclude that controller output is proportional to area below error till time *t*.
- I take care of the past values of the error. For example, if the error is not sufficiently large, it will accumulate over time and the controller will respond by applying a stronger action.
- Steady state error reduces with time.
- Slow response.
- Takes care of static loading.

Derivative Controller

- D accounts for possible future values of the error, based on its current rate of change.
- Controller output is proportional to rate of change of error.
- Seldom used alone.
- No effect on steady state error.
- Anticipatory in nature
- Cannot handle static loading.

Hence when these 3 controllers are combined, a PID controller is formed.

Mahadi *et al.* [37] used PD controller to control the angle of a linear inverted pendulum system. Later Pasqule *et al.* [38] used double PID controller to control linear inverted pendulum, where first PID controls the pendulum angle while the second PID controls the cart position. The block diagram of controller used by author is illustrated in Figure 2.2.

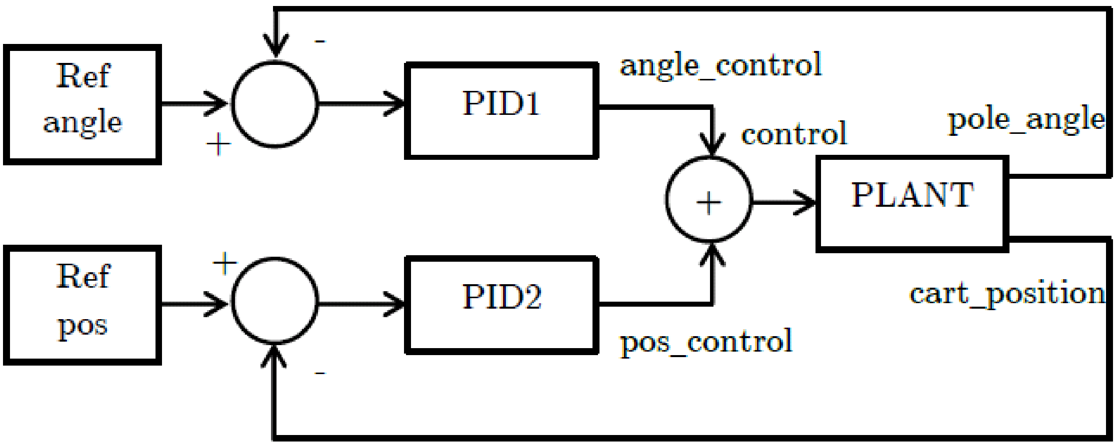

Figure 2.2: Complete 2-PID control system [13]

Author used the following method for tuning PID:

1. Initially, PID for position control was disabled and PID for pendulum anfle stabilization was tuned.
2. Once stability of pendulum angle is achieved, PID for position control is tuned with light parameters so as to preserve the previously achieved pendulum angle stabilization.

The author concluded that the main limitation of the plant relies in:

- Friction between the cart and the rail which remains considerable even after applying appropriate lubrication.
- Asymmetric traction offered by the belt in the two directions that caused some extra friction.
- Poor precision of motor and its underutilization.

Ghosh *et al.* [39] used gains from LQR Controller to tune a double PID controller for inverted pendulum stabilization. Since LQR is a PD controller an Integral element is added to the controller to encounter steady state error of the system with the advantage of easy tuning of LQR controller. To tune a PID controller he added a pole in negative s plane far away from LQR controller poles to get a fifth degree characteristic polynomial equation and compared it with polynomial equation of double PID controlled inverted pendulum and thus obtained all the PID controller gains in terms of one of the gain input by author. The results obtained by simulation and experimental testing were better than LQR as the controller has added advantage of integral gain in it.

Recently, a self-tuning PID controller is applied on the cart pendulum system with the help of Lyapunov function by Chang *et al.* [40], but only simulation was performed while verification of controller on actual experimental setup has not been performed.

2.5 Pole Placement Controller

Pole placement method places the closed loop poles system at the desired location by designing an appropriate gain of state feedback controller. The desired closed loop poles must be kept in negative *s* plane to ensure stability of the system, so the major task while designing a pole placement controller is to decide the desired closed loop poles location so as to ensure desired performance and stability of system.

Many researchers have tried to link the desired poles location with performance parameters such as settling time and maximum overshoot. Nivedita [41] used pole placement method for stabilization of cart inverted pendulum. Author input settling time and maximum overshoot values to define 2 of the desired poles and other 2 poles are chosen far away in the left side of *s* plane so that they should not dominate the response.

Since the Overshoot $M_p = e^{-\pi\xi / \sqrt{(1-\xi^2}}$ (2.3)

Damping frequency $\omega_d = \omega_n\sqrt{(1-\xi^2}$ (2.4)

For the desired closed loop system characteristics equation given by:

$$(s^2+2\xi\omega_n s+\omega_n^2)(s^2+as+b) = 0 \quad (2.5)$$

In the above equation the pair (ξ, ω_n) are chosen so that the poles are the dominant poles according to the required performance and the pair (a, b) are chosen far away in the left plane so that they should not dominant the response.

Author concluded that the proposed controller specified excellent performance. In addition, the system has a good robustness; it can overcome any external disturbances on cart or on pendulum rod.

Roshdy *et al.* [42] used pole placement method to stabilize real inverted pendulum using a pole separation factor. Author defined separation factor k between the dominant poles and the other poles. Author defined the pair (a, b) = ($k \times \xi$, $(k \times \omega_n)^2$), where k is separation factor between dominant poles and other poles. In order to obtain a small settling time and low overshoot (ξ, ω_n) are chosen as (0.858, 4.66). The separation factor of 2 is chosen by simulation which follows that (a, b) = (1.716, 86.8624).

Author concluded that the experimental results showed that the proposed controller specified excellent performance with 1.5 seconds settling time and low overshoot. In addition, the system has a good robustness; it can overcome any external disturbances on cart or on pendulum rod and stabilize within 2 seconds.

2.6 LQR Controller

Linear Quadratic Regulator is the optimal controller which aims at minimizing a cost function given in (2.4). The gain $\boldsymbol{K}$ of the controller is calculated using algebraic riccatti equation and depends on state weighing matrix $\boldsymbol{Q}$, control cost matrix $\boldsymbol{R}$, system matrix $\boldsymbol{A}$ and the input matrix $\boldsymbol{B}$. The state weighing matrix $\boldsymbol{Q}$ and control cost matrix $\boldsymbol{R}$ are tuning parameters while the system matrix $\boldsymbol{A}$ and the input matrix $\boldsymbol{B}$ depends on system to be controlled. Akhtaruzzaman and Shafie [43] used pole placement, 2-DOF PID and LQR Controller for swing up and stabilization of rotary inverted pendulum. Author concluded that:

- 2-DOF PID is not robust in practice.
- LQR is more robust than pole placement controller in swinging up of pendulum.
- LQR controller is also faster than pole placement controller.
- Overall, LQR is more consistent.

Recently, Trimpe *et al.* [44] proposed a self-tuning LQR approach using stochastic optimization, but only simulations are performed, while real-time implementation on experimental setup is not done.

2.7 Linear Quadratic Gaussian (LQG) Controller

Linear Quadratic Gaussian (LQG) Controller combines LQR with a kalman Filter which improves disturbance rejection of the system. Eide *et al.* [26] implemented this controller for balancing inverted pendulum mobile robot, however, results show that LQR give better results when compared to the LQG approach.

2.8 State-Dependent Riccati Equation (SDRE) Based Controller

This controller is a modification of LQR Controller. It can be considered as a nonlinear counterpart of LQR, where the matrices $\boldsymbol{A(x)}$, $\boldsymbol{B(x)}$, $\boldsymbol{Q(x)}$, $\boldsymbol{R(x)}$, are not constant and are the functions of states. SDRE control has an advantage of design flexibility of tuning the state weightage and input weightage matrices $\boldsymbol{Q}(\boldsymbol{x})$ and $\boldsymbol{R}(\boldsymbol{x})$ respectively, as functions of states. The SDRE control has been implemented on inverted pendulum by [27, 45] but only simulation results were displayed. Later, Dang and Lewis [46] successfully implemented

SDRE controller on real-time inverted pendulum for both the swing up as well as the stabilization. The major drawback of this controller is that it requires the solution of complicated state-dependent riccati equations at each time step which makes it computationally intense.

2.9 Energy Controller

Energy controller is the most popular controller for swing up problem of inverted pendulum, which is nonlinear in nature. Energy controller was first proposed by Astrom and Furuta [47] for the swing up of inverted pendulum. They used pendulum energy to design the controller where the motive of the controller is to maximize the energy of the pendulum, which happens at vertically upright pendulum position. The control input was designed based on Lyapunov theory where the Lyapunov function was taken as square of energy of pendulum. The controller performs well but the finite track length of cart was not taken into account which leads to cart hitting the limit switch while swinging up and thus ceasing the process.

Lozano *et al.* [48] proposed a Lyapunov function which considers the finite track length of cart for swing up the pendulum. The function is taken as sum of squares of mechanical energy, cart position and cart velocity which led to four parameters in control law which is difficult to tune. Chatterjee *et al.* [49] used a potential well concept to solve the finite cart track length problem. The control law is constructed in such a way that the cart experiences a repulsive force when it reaches the boundaries. This method is effective but the major limitation in their control law is to design five coefficients for the potential well by trial and error, which is a cumbersome job.

Yang *et al.* [50] addressed the finite track length problem by defining the control law, which is given by sum of square potential energy and weighted square of cart velocity using LaSalle’s theorem. As compared to previous papers Yang *et al.* [27] used two parameters only, which make the tuning process easy. However in their work the stabilization condition was not discussed. Many other researchers [51, 52] used Lyapunov based energy controller for swing up problem of inverted pendulum.

2.10 ANFIS based Controller

ANFIS is a soft computing technique that is used to tune a fuzzy inference system to follow a stipulated input-output data with minimum error tolerance. To do this, it used a hybrid of two optimization algorithms i.e. gradient decent algorithm and least square estimate. It is basically a knowledge based controller. In a recent approach [30], the usage of LQR based ANFIS has proposed to solve the problem of rule explosion and difficulty in tuning of a fuzzy controller. The author's used the fusion function to decrease the number of rules in fuzzy controller. Although the use of fusion functions solves the problems in fuzzy logic controller but it does no major improvement in performance when compared to parent LQR controller.

2.11 Sliding Mode Controller

Sliding mode control scheme is a well-known robust control scheme for dynamic uncertain systems. However, sliding mode control suffers from the dangerous chattering effect which prevents them from being extensively used in practice. Also, the performance of sliding mode control depends heavily on the sliding surface. If the sliding surface is not designed properly, it may lead to adverse effects. In [53], a modified Van der Pol oscillator is implemented for both the swing-up and the stabilization of the pendulum however, there were some performance issues due to fast switching in the controller input causing chattering. Recently, an aggressive sliding mode control law is presented for both the swing-up as well as stabilization of real-time rotary inverted pendulum [54].

2.12 Integral Sliding Mode Controller (ISMC)

Integral sliding mode is an improvement to sliding mode control. Unlike sliding mode controller, ISMC eliminates reachability phase. In designing an ISMC, it is considered that there exists a nominal plant, for which state feedback controller has already been properly designed to ensure asymptotic stability as well as predefined performance of the closed-loop system. To maintain the nominal performance along with insensitivity to external disturbances a discontinuous controller is added to the nominal state feedback controller. Recently, Bhavsar and Kumar [55] used ISMC to control linear inverted pendulum robust to input disturbance but only simulation results are presented and no experimentation validation is presented.

2.13 Conclusions from Literature Review

Through the literature review of the works reported by many authors on control strategies for inverted pendulum, following observations are drawn.

- Most of the authors neglect the effect of a friction, handling delays, constrained track length and additional dynamics of actuator, timing pulley, gears, etc. while modeling of the inverted pendulum system which makes the model simulation to behave different from real inverted pendulum thus reducing the performance of controllers.

- There is no any generalized (*x*-*y*-*z* Inverted pendulum) mathematical modeling of linear inverted pendulum systems such as *x*, *x*-*y* and *x*-*z* inverted pendulum system.
- Since the modeling cannot be expected to consider all the realistic assumptions so there is a need of robust controllers which are insensitive to parameter uncertainties and inaccurate plant modeling. Sliding model control has a problem of chattering and discontinuity therefore there is a need of alternative robust control scheme.
- Due to its wide applications in rockets, ships, robots, etc. there is a need of adaptive controller which adapts itself with change in environment conditions or modeling parameters. Few researchers have designed adaptive controller for inverted pendulum however, the performance obtained from controller is not good enough.
- In order to address the swing-up problem of the inverted pendulum system, it has been reviewed the energy controller is the most efficient than other controllers. The major problem by using energy controller is to swing up the pendulum in the restricted track length. Many authors have successfully proposed controller to swing-up the pendulum within the track by using different algorithms with energy technique. However, these methods are reported cumbersome to use or to implement on real time systems.
- There is a need of smooth hybrid controller for swing up and stabilization of inverted pendulum. Normally the controller designed for swing up and stabilization are different and a switch operation is performed between them to bring the pendulum to vertically upright position from stable position which leads to high control input and stressed on the setup. Although some researchers have designed a hybrid controller based on energy method but the performance of controller is not good enough.
- Practical implementation of self-tuning PID controller presented in [40] and self-tuning LQR controller presented in [19] is not done on experimental setup.
- Experimental validation of integral sliding mode controller on real-time inverted pendulum system has not been done although simulation results had been presented for linear inverted pendulum.

Chapter 3

Dynamic Modeling of Inverted Pendulum Systems

3.1 Introduction

The first step in design, analysis and control of any system is accurate dynamic mathematical modeling of the system. This chapter comprises of mathematical modeling of different kinds of inverted pendulum systems. Before that, basics of different types of modeling techniques are discussed. One of these techniques is further used to obtain the nonlinear mathematical model of different inverted pendulum systems. Further, the nonlinear models are linearized around the point of interests (equilibrium points) and state-space representation is presented, which is further used to design controllers discussed in next chapter.

3.2 Mathematical Modeling Techniques

Generally, two types of modeling techniques are used to obtain the mathematical model of the system– Newton-Euler and Euler-Langrange approach. One is based on Newton's second law while the other is based on energy of the system. Basic structure of each modeling technique is discussed below.

3.2.1 Newton-Euler Approach

In Newton-Euler method, the system is modeled on the basis of forces acting on it. It is based on Newton's second law of motion, in which dynamics of the system is described in terms of force and momentum [56]. It states that "*The acceleration of an object as produced by a net force is directly proportional to the magnitude of the net force, in the same direction as the net force, and inversely proportional to the mass of the object*". It involves all the forces and moments acting on the individual system links. The equations obtained from the Newton-Euler method include the constraint forces i.e. external forces, gravitational force, coriolis force, damping force, etc. acting between adjacent links. Thus, arithmetic operations are required to eliminate these terms and obtain relations between the torques and the resultant motion in terms of displacements.

3.2.2 Euler-Lagrangian Approach

In this method, the system is described in the terms of work and energy using kinematics (generalized coordinates). The kinetic and potential energies of the system are used to formulate the equation of motion of the system [57]. In deriving the differential equation of the system, it needs to define generalized coordinates, the Lagrangian and to state Hamilton's principle. The resultant equations provide a closed-form expression in terms of torques and displacements.

The Lagrange equation is:

$$\mathcal{L}(\boldsymbol{q}, \dot{q}, t) = KE(\boldsymbol{q}, \dot{\boldsymbol{q}}, t) - PE(\boldsymbol{q}, t) \tag{3.1}$$

In which, $\mathcal{L}$ is the Lagrangian operator, $\boldsymbol{q}$ is the vector of generalized coordinates, *KE* denotes the kinetic energy and *PE* denotes potential energy of the system.

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{q}_i}\right) - \frac{\partial \mathcal{L}}{\partial q_i} + \frac{\partial R}{\partial \dot{q}_i} = f_i \quad \forall \;\; i = 1, 2, \dots, n \tag{3.2}$$

where f_i is the i^{th} generalized force acting on the system and *R* is Rayleigh's Dissipation function.

3.3 Dynamic Modeling of Linear Inverted Pendulum Systems

The schematic diagrams of different types of linear inverted pendulum systems i.e. *x*, *x-z*, *x-y* and *x-y-z* are shown in Figure 3.1.

Here,

l	=	0.3	=	Distance between cart hinge point to the centre of mass of the pendulum in meters
M	=	1	=	Cart mass in kgs
m	=	0.1	=	Pendulum mass in kgs
g	=	9.8	=	Acceleration due to gravity in m/sec^2
θ	=	-	=	Pendulum angle from the vertical axis, as seen in *x-z* plane
ϕ	=	-	=	Pendulum angle from the vertical axis, as seen in *y-z* plane
F_x, F_y, F_z	=	-	=	Control forces on the cart in *x*, *y*, and *z* directions respectively

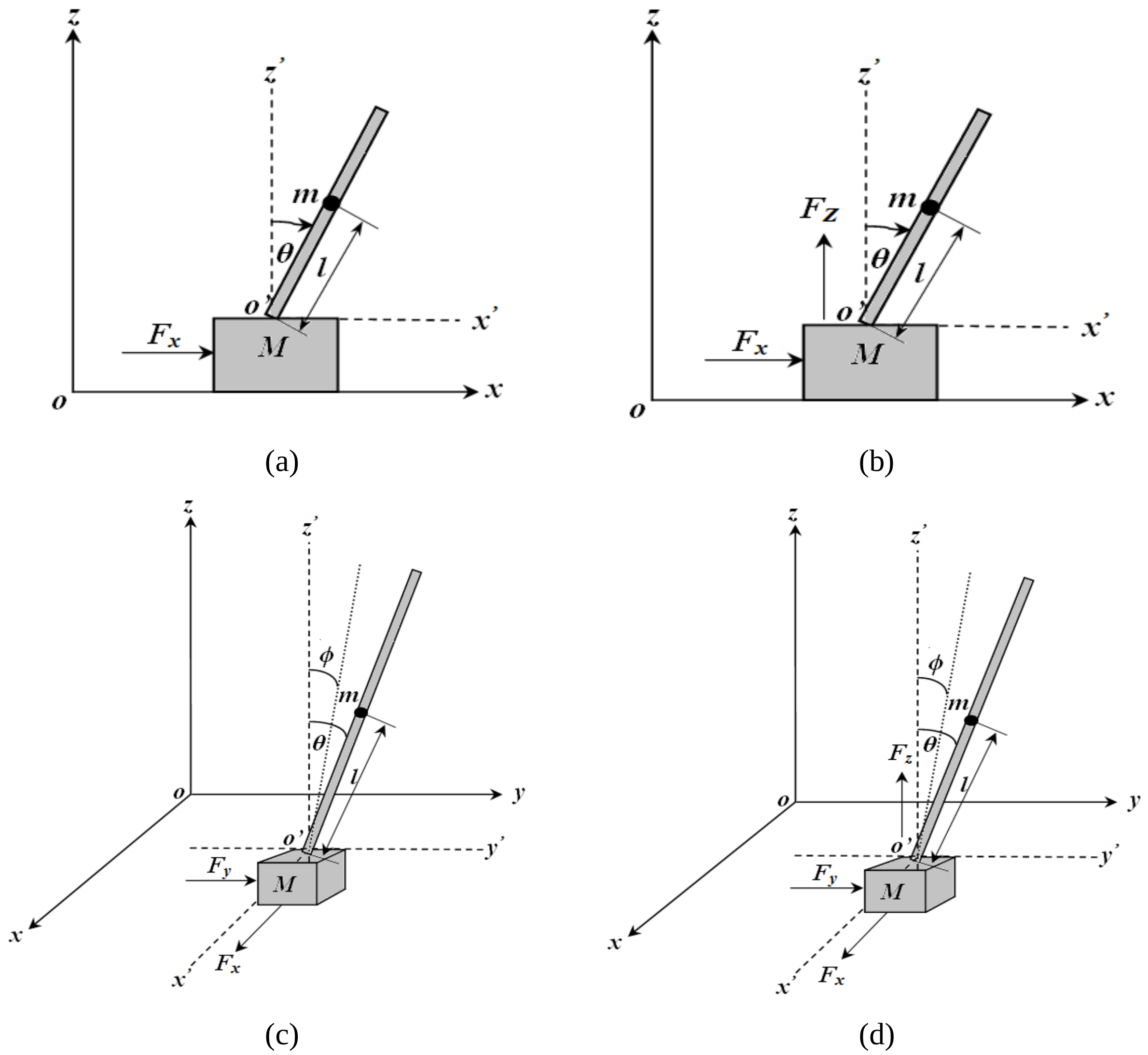

Figure 3.1: Schematic diagram of inverted pendulums – (a) *x* (b) *x-z* (c) *x-y* (d) *x-y-z*

3.3.1 Dynamic Modeling of *x-y-z* Inverted Pendulum

To derive the mathematical model of system, Euler Lagrange's approach is used with following assumptions:

1. Effect of friction is neglected.
2. Since radius of pendulum is small, moment of inertia of pendulum is neglected.

Thus, using Euler Lagrange's approach, the kinetic energy (*KE*) and the potential energy (*PE*) of the system can be represented as:

$$KE = \frac{1}{2}M(\dot{x}^2 + \dot{y}^2 + \dot{z}^2) + \frac{1}{2}m(\dot{x}_P^2 + \dot{y}_P^2 + \dot{z}_P^2) \tag{3.3}$$

And

$$PE = Mgz + mgz_P \tag{3.4}$$

Here, $(\dot{x}, \dot{y}, \dot{z})$ are the velocites of the cart in the *x*-*y*-*z* coordinate system. Similarly (x_P, y_P, z_P) and $(\dot{x}_P, \dot{y}_P, \dot{z}_P)$ are the position and velocity of the pendulum centre of mass in the *x*-*y*-*z* coordinate system.

where, $x_P = x + l\ cos\phi\ sin\theta, y_P = y + l\ cos\theta\ sin\phi$ and $z_P = z + l\ cos\theta\ cos\phi$

The system Lagrangian ($\mathcal{L}$) of *x*-*y*-*z* inverted pendulum is defined as:

$$\mathcal{L} = KE - PE \tag{3.5}$$

$$\mathcal{L} = \frac{1}{2}M(\dot{x}^2 + \dot{y}^2 + \dot{z}^2) + \frac{1}{2}m(\dot{x}_P^2 + \dot{y}_P^2 + \dot{z}_P^2) - (Mgz + mgz_P) \tag{3.6}$$

The state equations of *x*-*y*-*z* inverted pendulum have been derived using Euler Lagrange's equations and are given below:

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{x}}\right) - \frac{\partial \mathcal{L}}{\partial x} = F_x \tag{3.7}$$

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{y}}\right) - \frac{\partial \mathcal{L}}{\partial y} = F_y \tag{3.8}$$

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{z}}\right) - \frac{\partial \mathcal{L}}{\partial z} = F_z \tag{3.9}$$

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\theta}}\right) - \frac{\partial \mathcal{L}}{\partial \theta} = 0 \tag{3.10}$$

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{\phi}}\right) - \frac{\partial \mathcal{L}}{\partial \phi} = 0 \tag{3.11}$$

Using (3.7−3.11), the equations of motion for *x*-*y*-*z* inverted pendulum can be written as:

$$(M+m)\ddot{x} + ml\cos\theta\cos\phi\ddot{\theta} - 2ml\sin\phi\cos\theta\dot{\theta}\dot{\phi} - ml\sin\theta\sin\phi\ddot{\phi} - ml\cos\phi\sin\theta(\dot{\theta}^2 + \dot{\phi}^2) = F_x \tag{3.12}$$

$$(M+m)\ddot{y} + ml\cos\theta\cos\phi\ddot{\phi} - 2ml\sin\theta\cos\phi\dot{\theta}\dot{\phi} - ml\sin\phi\sin\theta\ddot{\theta} - ml\sin\phi\cos\theta(\dot{\theta}^2 + \dot{\phi}^2) = F_y \tag{3.13}$$

$$(M+m)\ddot{z} - ml\cos\theta\sin\phi\ddot{\phi} + 2ml\sin\theta\sin\phi\dot{\theta}\dot{\phi} - ml\cos\phi\sin\theta\ddot{\theta} - ml\cos\phi\cos\theta(\dot{\theta}^2 + \dot{\phi}^2) = F_z \tag{3.14}$$

$$\ddot{\theta}(ml^2\cos^2\phi + ml^2\sin^2\phi\sin^2\theta) + \dot{\theta}\dot{\phi}(-2ml^2\cos^2\theta\sin\phi\cos\phi)$$
$$+\dot{\theta}^2(ml^2\sin^2\phi\sin\theta\cos\theta) + \dot{\phi}^2(ml^2\sin^2\phi\sin\theta\cos\theta) - (ml^2\sin\phi\sin\theta\cos\phi\cos\theta)\ddot{\phi}$$
$$-(ml\sin\theta\sin\phi)\ddot{y} + (ml\cos\theta\cos\phi)\ddot{x} - (ml\cos\phi\sin\theta)\ddot{z} - mgl\sin\theta\cos\phi = 0 \tag{3.15}$$

$$\ddot{\phi}(ml^2\cos^2\theta + ml^2\sin^2\theta\sin^2\phi) + \dot{\theta}\dot{\phi}(-2ml^2\cos^2\phi\sin\theta\cos\theta)$$
$$+\dot{\phi}^2(ml^2\sin^2\theta\sin\phi\cos\phi) + \dot{\theta}^2(ml^2\sin^2\theta\sin\phi\cos\phi) - (ml^2\sin\phi\sin\theta\cos\phi\cos\theta)\ddot{\theta}$$
$$-(ml\sin\theta\sin\phi)\ddot{x} + (ml\cos\theta\cos\phi)\ddot{y} - (ml\cos\theta\sin\phi)\ddot{z} - mgl\cos\theta\sin\phi = 0 \tag{3.16}$$

On linearizing the equation of motion about unstable equilibrium point ($\theta = 0,\ \phi = 0$) and representing in state-space form

$$\begin{bmatrix}\dot{x}\\ \ddot{x}\\ \dot{y}\\ \ddot{y}\\ \dot{z}\\ \ddot{z}\\ \dot{\theta}\\ \ddot{\theta}\\ \dot{\phi}\\ \ddot{\phi}\end{bmatrix} = \begin{bmatrix} 0&1&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&\frac{-mg}{M}&0&0&0\\ 0&0&0&1&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&\frac{-mg}{M}&0\\ 0&0&0&0&0&1&0&0&0&0\\ 0&0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&1&0&0\\ 0&0&0&0&0&0&\frac{(M+m)g}{Ml}&0&0&0\\ 0&0&0&0&0&0&0&0&0&1\\ 0&0&0&0&0&0&0&0&\frac{(M+m)g}{Ml}&0 \end{bmatrix}\begin{bmatrix}x\\ \dot{x}\\ y\\ \dot{y}\\ z\\ \dot{z}\\ \theta\\ \dot{\theta}\\ \phi\\ \dot{\phi}\end{bmatrix} + \begin{bmatrix} 0&0&0\\ \frac{1}{M}&0&0\\ 0&0&0\\ 0&\frac{1}{M}&0\\ 0&0&0\\ 0&0&\frac{1}{M+m}\\ 0&0&0\\ -\frac{1}{Ml}&0&0\\ 0&0&0\\ 0&-\frac{1}{Ml}&0 \end{bmatrix}\begin{bmatrix}F_x\\ F_y\\ F_z-g(M+m)\end{bmatrix} \tag{3.17}$$

Further, incorporating the parameters, (3.17) yields:

$$\begin{bmatrix}\dot{x}\\ \ddot{x}\\ \dot{y}\\ \ddot{y}\\ \dot{z}\\ \ddot{z}\\ \dot{\theta}\\ \ddot{\theta}\\ \dot{\phi}\\ \ddot{\phi}\end{bmatrix} = \begin{bmatrix} 0&1&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&-0.98&0&0&0\\ 0&0&0&1&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&0&-0.98&0\\ 0&0&0&0&0&1&0&0&0&0\\ 0&0&0&0&0&0&0&0&0&0\\ 0&0&0&0&0&0&0&1&0&0\\ 0&0&0&0&0&0&35.93&0&0&0\\ 0&0&0&0&0&0&0&0&0&1\\ 0&0&0&0&0&0&0&0&35.93&0 \end{bmatrix}\begin{bmatrix}x\\ \dot{x}\\ y\\ \dot{y}\\ z\\ \dot{z}\\ \theta\\ \dot{\theta}\\ \phi\\ \dot{\phi}\end{bmatrix} + \begin{bmatrix} 0&0&0\\ 1&0&0\\ 0&0&0\\ 0&1&0\\ 0&0&0\\ 0&0&0.91\\ 0&0&0\\ -3.33&0&0\\ 0&0&0\\ 0&-3.33&0 \end{bmatrix}\begin{bmatrix}F_x\\ F_y\\ F_z-10.78\end{bmatrix} \tag{3.18}$$

The *x-y-z* inverted pendulum is the generalized case of linear inverted pendulum. The model of *x*, *x-y* and *x-z* inverted pendulum can easily be derived from *x-y-z* inverted pendulum by putting some of the state variables zero.

3.3.2 Dynamic Modeling of *x-y* Inverted Pendulum

The *x-y* inverted pendulum system will only transverse in *x*- and *y*-direction and not in *z*-direction. Thus the component $\dot{z}$ will be zero along with F_z .Thus, the evaluation of the system can be similarly written as in *x-y-z* inverted pendulum, using the following state equations, which are derived using Lagrangian approach as given below.

$$(M+m)\ddot{x}+mlC_\theta C_\phi\ddot{\theta}-2mlS_\phi C_\theta\dot{\theta}\dot{\phi}-mlS_\theta S_\phi\ddot{\phi}-mlC_\phi S_\theta(\dot{\theta}^2+\dot{\phi}^2)=F_x \tag{3.19}$$

$$(M+m)\ddot{y}+mlC_\theta C_\phi\ddot{\phi}-2mlS_\theta C_\phi\dot{\theta}\dot{\phi}-mlS_\phi S_\theta\ddot{\theta}-mlS_\phi C_\theta(\dot{\theta}^2+\dot{\phi}^2)=F_y \tag{3.20}$$

$$\begin{aligned}&\ddot{\theta}(ml^2C_\phi^2+ml^2S_\phi^2S_\theta^2)+\dot{\theta}\dot{\phi}(-2ml^2C_\theta^2S_\phi C_\phi)+\dot{\theta}^2(ml^2S_\phi^2S_\theta C_\theta)+\dot{\phi}^2(ml^2S_\phi^2S_\theta C_\theta)\\&-(ml^2S_\phi S_\theta C_\phi C_\theta)\ddot{\phi}-(mlS_\theta S_\phi)\ddot{y}+(mlC_\theta C_\phi)\ddot{x}-mglS_\theta C_\phi=0\end{aligned} \tag{3.21}$$

$$\begin{aligned}&\ddot{\phi}(ml^2C_\theta^2+ml^2S_\theta^2S_\phi^2)+\dot{\theta}\dot{\phi}(-2ml^2C_\phi^2S_\theta C_\theta)+\dot{\phi}^2(ml^2S_\theta^2S_\phi C_\phi)+\dot{\theta}^2(ml^2S_\theta^2S_\phi C_\phi)\\&-(ml^2S_\phi S_\theta C_\phi C_\theta)\ddot{\theta}-(mlS_\theta S_\phi)\ddot{x}+(mlC_\theta C_\phi)\ddot{y}-mglC_\theta S_\phi=0\end{aligned} \tag{3.22}$$

where, $S_\theta=\sin(\theta), C_\theta=\cos(\theta), S_\phi=\sin(\phi)\ and\ C_\phi=\cos(\phi)$

The state-space representation of *x-y* inverted pendulum is calculated as:

$$\begin{bmatrix}\dot{x}\\\ddot{x}\\\dot{y}\\\ddot{y}\\\dot{\theta}\\\ddot{\theta}\\\dot{\phi}\\\ddot{\phi}\end{bmatrix}=\begin{bmatrix}0&1&0&0&0&0&0&0\\0&0&0&0&\frac{-mg}{M}&0&0&0\\0&0&0&1&0&0&0&0\\0&0&0&0&0&0&\frac{-mg}{M}&0\\0&0&0&0&0&1&0&0\\0&0&0&0&\frac{(M+m)g}{Ml}&0&0&0\\0&0&0&0&0&0&0&1\\0&0&0&0&0&0&\frac{(M+m)g}{Ml}&0\end{bmatrix}\begin{bmatrix}x\\\dot{x}\\y\\\dot{y}\\\theta\\\dot{\theta}\\\phi\\\dot{\phi}\end{bmatrix}+\begin{bmatrix}0&0\\1/M&0\\0&0\\0&1/M\\0&0\\-1/Ml&0\\0&0\\0&-1/Ml\end{bmatrix}\begin{bmatrix}F_x\\F_y\end{bmatrix} \tag{3.23}$$

Plugging in the value of different parameters, (3.23) can be written as:

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{y} \\ \ddot{y} \\ \dot{\theta} \\ \ddot{\theta} \\ \dot{\phi} \\ \ddot{\phi} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -0.98 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -0.98 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 35.93 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 35.93 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ y \\ \dot{y} \\ \theta \\ \dot{\theta} \\ \phi \\ \dot{\phi} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 1 \\ 0 & 0 \\ -3.33 & 0 \\ 0 & 0 \\ 0 & -3.33 \end{bmatrix} \begin{bmatrix} F_X \\ F_y \end{bmatrix} \tag{3.24}$$

3.3.3 Dynamic Modeling of *x*-*z* Inverted Pendulum

The *x*-*z* inverted pendulum system will only transverse in *x*- and *z*-direction and not in *y*-direction. Thus the component $\dot{y}$ will be zero along with F_y. Also pendulum angular deflection in *y*-direction i.e. ϕ will be zero. Thus, Lagrange's equations of motion [25] for *x*-*z* inverted pendulum are given as:

$$(M+m)\ddot{x} + ml\cos\theta\ddot{\theta} - ml\sin\theta\dot{\theta}^2 = F_x \tag{3.25}$$

$$(M+m)\ddot{z} - ml\sin\theta\ddot{\theta} - ml\cos\theta\dot{\theta}^2 + (M+m)g = F_z \tag{3.26}$$

$$l\ddot{\theta} + \cos\theta\ddot{x} - \sin\theta\ddot{z} - g\sin\theta = 0 \tag{3.27}$$

On linearizing, the equation of motion and representing it in state-space form:

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{z} \\ \ddot{z} \\ \dot{\theta} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & \frac{-m}{M}g & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & \frac{g(M+m)}{Ml} & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ z \\ \dot{z} \\ \theta \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 1/M & 0 \\ 0 & 0 \\ 0 & 1/(M+m) \\ 0 & 0 \\ -1/Ml & 0 \end{bmatrix} \begin{bmatrix} F_x \\ F_z - g(M+m) \end{bmatrix} \tag{3.28}$$

Further, incorporating the parameters in (3.28), yields:

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{z} \\ \ddot{z} \\ \dot{\theta} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & -0.98 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 35.93 & 0 \end{bmatrix} \begin{bmatrix} x \\ \dot{x} \\ z \\ \dot{z} \\ \theta \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 1 & 0 \\ 0 & 0 \\ 0 & 0.91 \\ 0 & 0 \\ -3.33 & 0 \end{bmatrix} \begin{bmatrix} F_x \\ F_z - 10.78 \end{bmatrix} \tag{3.29}$$

3.3.4 Dynamic Modeling of *x* inverted pendulum

The *x* inverted pendulum is the most simplest and popular case of linear inverted pendulum. In this, the cart only transverse in *x*-direction while the pendulum is hinged to a cart and is only allowed to oscillate around *y*-axis. Thus, using Lagrange approach, the equations of motion [25] of the *x* inverted pendulum can be written as:

$$(M+m)\ddot{x} + ml\cos\theta\ddot{\theta} - ml\sin\theta\dot{\theta}^2 = F_x \tag{3.30}$$

$$\cos\theta\,\ddot{x} + l\ddot{\theta} - g\sin\theta = 0 \tag{3.31}$$

After linearizing, the system can be represented in state-space form as written below:

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\theta} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{-m}{M}g & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & \frac{(M+m)}{Ml}g & 0 \end{bmatrix} \begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\theta} \\ \ddot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ 1/M \\ 0 \\ -1/Ml \end{bmatrix} F_x \tag{3.32}$$

Incorporating the parameter values, (3.32) can be written as:

$$\begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\theta} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & -0.98 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 35.93 & 0 \end{bmatrix} \begin{bmatrix} \dot{x} \\ \ddot{x} \\ \dot{\theta} \\ \ddot{\theta} \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \\ 0 \\ -3.33 \end{bmatrix} F_x \tag{3.33}$$

3.4 Dynamic Modeling of Rotary Inverted Pendulum System

The schematic diagram of a rotary inverted pendulum as well as QUBE-Servo platform is shown in Figures 3.2-3.3. It consists of a rotary arm on which a pendulum is pivoted. The rotary arm is directly controlled by motor while pendulum is indirectly controlled, thus, making it an underactuated system.

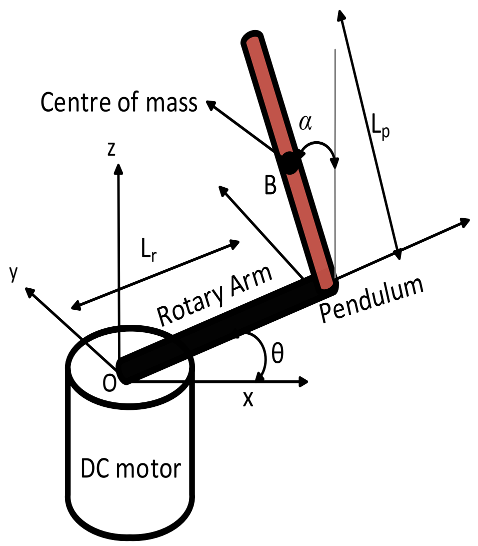

Figure 3.2: Schematic diagram of RIP system [24]

Figure 3.3: QUBE-Servo platform

Here, L_r represents length of rotary arm, L_p represents length of the pendulum, θ represents rotary arm angle which is taken positive in counter-clockwise (CCW) direction and α represents pendulum angle from vertically upright pendulum position and is taken positive in CCW as seen from outside of system. The servo should turn in the CCW direction when the control voltage is positive, i.e. $V_m > 0$.The generalized coordinates of the system are θ and α. The parameters of rotary inverted pendulum are given in Table 3.1.

Table 3.1: Parameters of RIP System

Symbol	**Description**	**Units**	**Values**
m_p	Mass of pendulum	kg	0.024
J_p	Pendulum inertia	kgm^2	3.3×10^{-5}
L_p	Length of the pendulum	m	0.129
B_p	Pendulum damping coefficient	Nms/rad	0.0005
J_r	Rotary arm inertia	kgm^2	5.7×10^{-5}
L_r	Length of the rotary arm	m	0.085
B_r	Viscous damping coefficient	Nms/rad	0.0015

The kinematics of the system is given by:

$$\boldsymbol{X}_r = \begin{bmatrix} x_r \\ y_r \end{bmatrix} = \begin{bmatrix} L_r \cos\theta \\ L_r \sin\theta \end{bmatrix} \tag{3.34}$$

$$\dot{\boldsymbol{X}}_{\boldsymbol{r}} = \begin{bmatrix} \dot{x}_r \\ \dot{y}_r \end{bmatrix} = \begin{bmatrix} -L_r\dot{\theta}\sin\theta \\ L_r\,\dot{\theta}\cos\theta \end{bmatrix} \tag{3.35}$$

$$\boldsymbol{X}_{\boldsymbol{p}} = \begin{bmatrix} x_p \\ y_p \\ z_p \end{bmatrix} = \begin{bmatrix} L_r\cos\theta + \frac{L_p}{2}\sin\alpha\sin\theta \\ L_r sin\theta - \frac{L_p}{2} sin\alpha\cos\theta \\ \frac{L_p}{2}\cos\alpha \end{bmatrix} \tag{3.36}$$

$$\dot{\boldsymbol{X}}_{\boldsymbol{p}} = \begin{bmatrix} \dot{x}_p \\ \dot{y}_p \\ \dot{z}_p \end{bmatrix} = \begin{bmatrix} -L_r\dot{\theta}\sin\theta + \frac{L_p}{2}\cos\alpha\sin\theta\,\dot{\alpha} + \frac{L_p}{2}\sin\alpha\cos\theta\,\dot{\theta} \\ L_r\,\dot{\theta}\cos\theta - \frac{L_p}{2}\cos\alpha\cos\theta\,\dot{\alpha} + \frac{L_p}{2}\sin\alpha\sin\theta\dot{\theta} \\ -\frac{L_p}{2}\sin\alpha\,\dot{\alpha} \end{bmatrix} \tag{3.37}$$

Total kinetic energy (*KE*) is given by summation of kinetic energy of pendulum and rotary arm, which is given as:

$$KE = KE_{arm} + KE_{Pend} \tag{3.38}$$

$$= \frac{1}{2}\left(m_pL_r^2 + J_r + m_p\frac{L_p^2}{4}\sin^2\alpha\,)\,\dot{\theta}^2 + \frac{1}{2}(m_p\frac{L_p^2}{4} + J_p\right)\dot{\alpha}^2 - \frac{1}{2}m_pL_rL_p\dot{\theta}\dot{\alpha}\cos\alpha \tag{3.39}$$

In this system, only pendulum possesses potential energy. The total potential energy (*PE*) of the system is given as:

$$PE = PE_{arm} + PE_{Pend} \tag{3.40}$$

$$= 0 + (-\,m_pg(\frac{L_p}{2} - \frac{L_p}{2}\cos\alpha) \tag{3.41}$$

Lagrangian $\mathcal{L}$ of the system is given by subtraction of total kinetic energy and total potential energy of the system, given as:

$$\mathcal{L} = KE - PE \tag{3.42}$$

$$= \frac{1}{2}\left(m_pL_r^2 + J_r + m_p\frac{L_p^2}{4}\sin^2\alpha\,)\,\dot{\theta}^2 + \frac{1}{2}(m_p\frac{L_p^2}{4} + J_p\right)\dot{\alpha}^2 + \tag{3.43}$$

$$\frac{1}{2}m_pL_rL_p\dot{\theta}\dot{\alpha}\cos\alpha + m_pg(\frac{L_p}{2} - \frac{L_p}{2}\cos\alpha)$$

To drive damping force of the system Rayleigh's dissipation function is used, which is given as:

$$R = \frac{1}{2}B_r\dot{\theta}^2 + \frac{1}{2}B_p\dot{\alpha}^2 \tag{3.44}$$

Where, B_r is equivalent viscous damping coefficient for rotary arm and B_p is equivalent viscous damping coefficient for pendulum. Due to two generalized coordinates, there are two equation of motion of the system. Therefore, Lagrange's equations of the system are given as:

$$\frac{d}{dt}\left(\frac{\partial\mathcal{L}}{\partial\dot{\theta}}\right) - \frac{\partial\mathcal{L}}{\partial\theta} + \frac{\partial R}{\partial\dot{\theta}} = \tau \tag{3.45}$$

$$\frac{d}{dt}\left(\frac{\partial\mathcal{L}}{\partial\dot{\alpha}}\right) - \frac{\partial\mathcal{L}}{\partial\alpha} + \frac{\partial R}{\partial\dot{\alpha}} = 0 \tag{3.46}$$

Further, on solving and linearizing the equations about the operating point (α = 0), the equations of motion obtained as follows:

$$(m_pL_r^2 + J_r)\ddot{\theta} - \frac{1}{2}m_pL_pL_r\ddot{\alpha} + B_r\dot{\theta} = \tau \tag{3.47}$$

$$-\frac{1}{2}m_pL_pL_r\ddot{\theta} + \left(J_p + \frac{1}{4}m_pL_p^2\right)\ddot{\alpha} - \frac{m_pL_pg\alpha}{2} + B_p\dot{\alpha} = 0 \tag{3.48}$$

Solving the acceleration term yields:

$$\ddot{\theta} = \frac{1}{J_T}\left(-\left(J_p + \frac{1}{4}m_pL_p^2\right)B_r\dot{\theta} - \frac{1}{2}m_pL_pL_rB_p\dot{\alpha} + \frac{1}{4}m_p^2L_p^2L_rg\alpha + \left(J_p + \frac{1}{4}m_pL_p^2\right)\tau\right) \tag{3.49}$$

And

$$\ddot{\alpha} = \frac{1}{J_T}\left(\frac{1}{2}m_pL_pL_rB_r\,\dot{\theta} - (J_r + m_pL_r^2)B_p\dot{\alpha} + \frac{1}{2}m_pL_pg(J_r + m_pL_r^2)\alpha - \frac{1}{2}m_pL_pL_r\tau\right) \tag{3.50}$$

$$J_T = J_pm_pL_r^2 + J_rJ_p + \frac{1}{4}J_rm_pL_p^2 \tag{3.51}$$

In the above equations, τ represents the torque generated by a direct-drive rotary servo system. The dynamics of the actuator is described by the following equation:

$$\tau = K_t\left(\frac{V_m - K_m\dot{\theta}}{R_m}\right) \tag{3.52}$$

Where, the actuator parameters are described in Table 3.2.

Table 3.2: Parameters of actuator

Symbol	**Description**	**Units**	**Value**
J_m	Motor inertia	kgm^2	4 x 10^{-6}
R_m	Terminal resistance	Ω	8.4
K_m	Motor back EMF constant	Vs/rad	0.042
K_t	Torque constant	Nm/A	0.042
L_m	Rotor inductance	mH	0.85

State-Space Representation

The behavior of a system at any given time for any set of quantities is called state of the system. The quantities which determine the state are called state variables and the hypothetical space spanned by these state variables is called the state-space. The state variables are mathematically equated together within the state-space is called the state-space representation. The matrix differential equation can be represented in a state-space form as

$$\dot{\boldsymbol{x}} = \boldsymbol{Ax} + \boldsymbol{Bu} \tag{3.53}$$

$$\boldsymbol{y} = \boldsymbol{Cx} + \boldsymbol{Du} \tag{3.54}$$

where $\boldsymbol{x} = [\theta\ \alpha\ \dot{\theta}\ \dot{\alpha}]$**,** is called as the state vector, $\boldsymbol{u}$ is known as the input vector and $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$ and $\boldsymbol{D}$ are called as state-weighing coefficient matrices. The derived mathematical model of the rotary inverted pendulum system can be represented in state-space form as:

$$\begin{bmatrix}\dot{\theta}\\ \dot{\alpha}\\ \ddot{\theta}\\ \ddot{\alpha}\end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\\ 0 & \left(\frac{m_p^2 L_p^2 L_r g}{4J_t}\right) & \frac{-D_r}{J_t}\left(J_p + \frac{m_p L_p^2}{4}\right) - \frac{K_t^2}{R_m}\left(\frac{K_t}{J_t R_m}\left(J_p + \frac{1}{4}m_p L_p^2\right)\right) & \left(\frac{-m_p L_r D_p L_p}{2J_t}\right)\\ 0 & \frac{m_p g L_p}{2J_t}\left(J_r + m_p L_r^2\right) & -\frac{m_p L_p L_r D_r}{2J_t} - \frac{K_t^2}{R_m}\left(\frac{K_t}{2J_t R_m} m_p L_p L_r\right) & \frac{-D_p}{J_t}\left(J_r + m_p L_r^2\right)\end{bmatrix}\begin{bmatrix}\theta\\ \alpha\\ \dot{\theta}\\ \dot{\alpha}\end{bmatrix} + \begin{bmatrix}0\\ 0\\ \frac{K_t}{J_t R_m}\left(J_p + \frac{1}{4}m_p L_p^2\right)\\ \frac{K_t}{J_t R_m}\left(\frac{1}{2}m_p L_p L_r\right)\end{bmatrix}[V_m] \tag{3.55}$$

Incorporating the values of parameters given in Table 3.1 and Table 3.2, the state-space representation can be expressed as:

$$\begin{bmatrix}\dot{\theta}\\ \dot{\alpha}\\ \ddot{\theta}\\ \ddot{\alpha}\end{bmatrix} = \begin{bmatrix} 0 & 0 & 1 & 0\\ 0 & 0 & 0 & 1\\ 0 & 149.2751 & -2.0886 & 0\\ 0 & 261.6091 & -2.0643 & 0\end{bmatrix}\begin{bmatrix}\theta\\ \alpha\\ \dot{\theta}\\ \dot{\alpha}\end{bmatrix} + \begin{bmatrix}0\\ 0\\ 49.7275\\ 49.1493\end{bmatrix}[V_m] \tag{3.56}$$

Comparison with Real Model

The above mathematical model of rotary inverted pendulum is compared with real rotary inverted pendulum using Matlab Simulink using the Simulink model as shown in Figure 3.4.

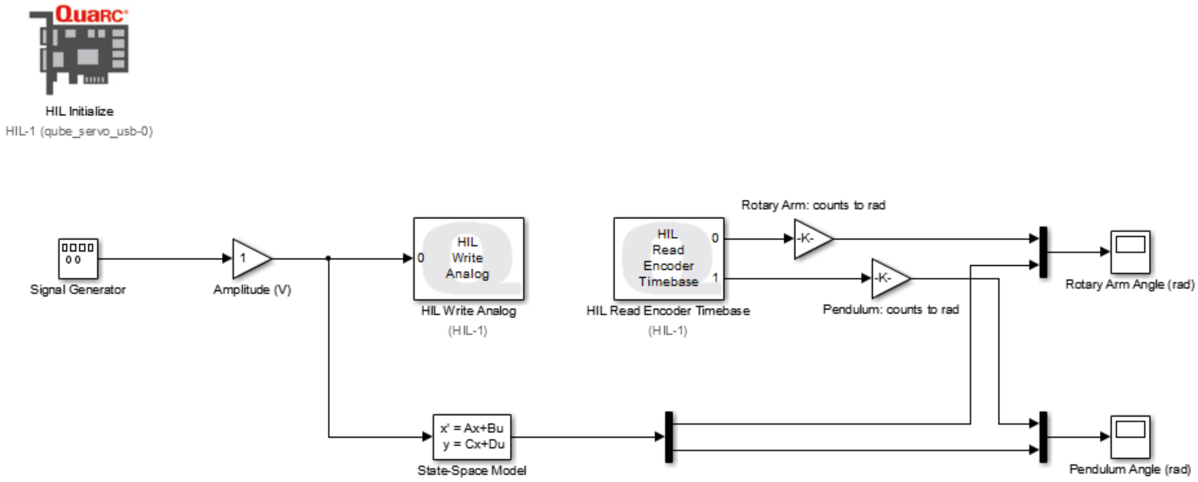

Figure 3.4: Simulink block diagram for comparison with real model

From the response illustrated in Figure 3.5, it is concluded that the pendulum response of the simulated model represents the actual pendulum response of the system accurately because in the simulated response which is blue in color closely resembles the measured response which is red in color while the model of arm response displays the characteristics of the measured arm response, but an offset is observed due to un-modeled dynamics such as disturbance introduced by the encoder cable and due to certain uncertainties.

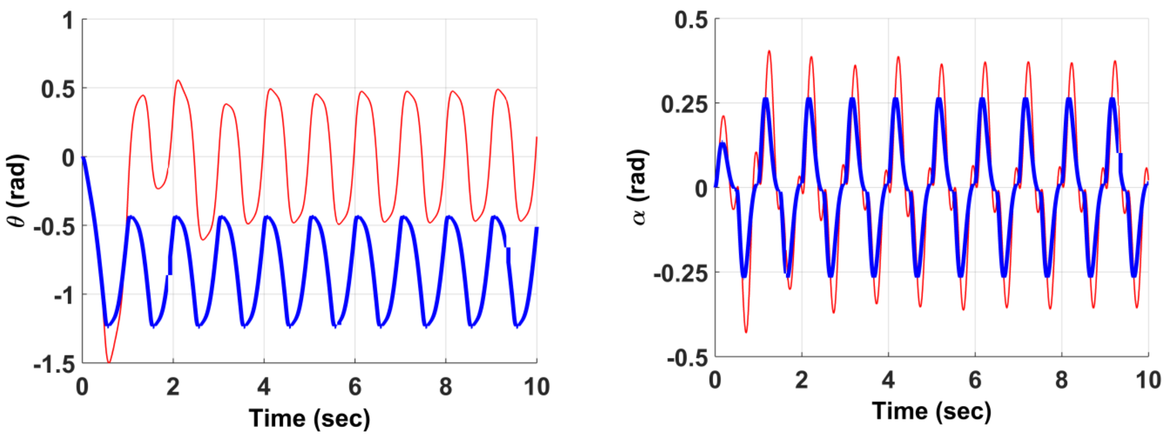

Figure 3.5: Comparison of simulation response with real-time model (red line: Simulation response and blue line: Real-time response

3.5 Summary

This chapter represented the dynamic modeling of different inverted pendulum systems i.e. rotary inverted pendulum system and linear inverted pendulum system. The Euler-Lagrangian method has been used to obtain dynamic model. The derived mathematical model of rotary inverted pendulum has been compared with actual experimental setup of rotary inverted pendulum. In linear inverted pendulum, mathematical modeling of *x*, *x-z*, *x-y* and *x-y-z* inverted pendulum systems were discussed. Each model has been linearized around the unstable pendulum position i.e. vertically upright position, which was further used to obtain state-space representation of the model. The obtained state-space representation is further used in designing controller for the system, which is discussed in the next chapter.

Chapter 4

Stabilization Controller Design

4.1 Introduction

Stabilization control means controlling the system to a desired position, when it the system in brought manually to the linear range of the desired point. In this thesis, the unstable equilibrium (upright) position of the pendulum is considered as the desired point of interest. This control problem is the most popular one due to its high applications in real-world systems. Control of rockets, ships, buildings, and segway etc. is based on stabilization principle of inverted pendulum. This chapter discusses the different modern controller design for stabilization problem of rotary inverted pendulum system. These controllers are linear controllers and are tuned according to linearized state-space model of system. The prime objective of the stabilization controller is stability while the secondary objective being optimality and robustness. To achieve these objectives, different stabilization controllers are discussed below.

4.2 Controllability of the System

Before designing the controller, the first and foremost requirement is to ensure the controllability of the system. Controllability is defined as the property of a system when it is possible to take the system from any initial state to any final state in a finite time by means of the input vector. Controllability of any system can be checked easily from the state-space representation of the system.

The theorem for controllability state that, "*A linear, time-invariant system described by the matrix state-equation,* $\dot{\boldsymbol{x}} = \boldsymbol{A}\boldsymbol{x} + \boldsymbol{B}\boldsymbol{u}$ *is controllable if and only if the controllability test matrix is of rank n, the order of the system*". The controllability of the system can also be checked using in-built Matlab function *ctrb* ($\boldsymbol{A}$, $\boldsymbol{B}$), which is based on the above theorem. The controllability test matrix is given by:

$$\boldsymbol{M} = [\boldsymbol{B}; \boldsymbol{AB};\ \boldsymbol{A}^2\boldsymbol{B};\ldots;\ \boldsymbol{A}^{n-1}\boldsymbol{B}] \tag{4.1}$$

Using the system matrices of rotary inverted pendulum, we obtained controllability test matrix as:

$$M = \begin{bmatrix} 0 & 50 & -104 & 7554 \\ 0 & 49 & -103 & 13072 \\ 50 & -104 & 7554 & -31100 \\ 49 & -103 & 13072 & -424467 \end{bmatrix}$$

The obtained controllability matrix is a full rank matrix. Hence the system is controllable at vertically upright position so certain controllers will be applied to control the system in next section.

4.3 Pole Placement Controller

Pole placement controller works on a principle of placing the closed loop poles of the system to the desired place by designing an appropriate state feedback controller gain. The locations of desired closed loop poles are chosen in order to satisfy system transient and steady state performance requirement. Pole placement design consists of following steps:

1) Verify the controllability of the system.

2) Using characteristic polynomial of matrix $\boldsymbol{A}$ as given below:

$$|sI - \boldsymbol{A}| = s^n + a_1 s^{n-1} + \ldots\ldots\ldots\ldots a_{n-1}s + a_{n.} \tag{4.2}$$

Calculate values of $a_1, a_2, a_3, \ldots, a_n$

3) Determine matrix $\boldsymbol{T}$ that transform the state-space equations to controllable canonical form:

$$\boldsymbol{T} = \boldsymbol{MW} \tag{4.3}$$

In which $\boldsymbol{M}$ is the controllability matrix given as:

$$\boldsymbol{M} = [\boldsymbol{B}{:}\, \boldsymbol{AB}{:}\ldots\ldots{:}\boldsymbol{A}^{n-1}\boldsymbol{B}] \tag{4.4}$$

And,

$$\boldsymbol{W} = \begin{bmatrix} a_{n-1} & a_{n-2} & \ldots & a_1 & 1 \\ a_{n-2} & a_{n-3} & 1 & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ a_1 & 1 & \ldots & 0 & 0 \\ 1 & 0 & \ldots & 0 & 0 \end{bmatrix} \tag{4.5}$$

4) Formulate the desired polynomial using the desired closed loop poles $\mu_1, \mu_2, \ldots, \mu_n$.

$$(s - \mu_1)\,(s - \mu_2)\ldots\ldots(s - \mu_n) = s^n + \alpha_1 s^{n-1} + \ldots\ldots\ldots\ldots \alpha_{n-1}s + \alpha_{n.} \tag{4.6}$$

And calculate the values of $\alpha_1, \alpha_2, \ldots, \alpha_n$.

5) State feedback gain matrix $\boldsymbol{K_{PPC}}$ is calculated by following equation:

$$\boldsymbol{K_{PPC}} = [\alpha_n - a_n{:}\, \alpha_{n-1} - a_{n-1}{:}\ldots\ldots\ldots\ldots{:}\, \alpha_2 - a_2{:}\, \alpha_1 - a_1]\boldsymbol{T}^{-1} \tag{4.7}$$

The state feedback gain matrix $\boldsymbol{K_{PPC}}$ can also be obtained using Ackermann's Formula [58]. Alternatively, a pre-defined Matlab function *place* can also be used.

Results and Discussions

Since the system is controllable as proved in previous section, poles location are determined which are found by finding eigen values of matrix ***A*** which came out to be [0, 15.627, -16.809, -0.9070] . Since one the pole is positive so the system is unstable. To make the system stable we will place the poles of the system to desired location by using pole placement controller. The locations of two of the desired poles is chosen by defining settling time and maximum overshoot of the system [16], while the other two poles are kept far away in left hand plan to reduce their effect on performance of the system. So the dominating poles were that obtained from maximum overshoot and settling time. By using the relations given below:

$$\text{Maximum Overshoot } M_P = e^{-\pi\xi / \sqrt{(1-\xi^2}} \tag{4.8}$$

$$\text{Settling time } t_s = 4/\ \xi\ \omega_n \tag{4.9}$$

For the desired closed loop system characteristics equation given by:

$$(s^2+2\xi\omega_n s+\omega_n^2)(s^2+as+b) = 0 \tag{4.10}$$

In the above equation the pair (ξ, ω_n) are chosen so that the poles are the dominant poles according to the required performance and the pair (a, b) are chosen far away in the left plane so that they should not dominate the response. In order to obtain a small settling time and low overshoot, the desired poles are chosen as ***P*** = [-2.0000+1.9058i, -2.0000-1.9058i, -30, -40]. To place the system poles at desired locations, controller gain is calculated as $\boldsymbol{K_{PPC}}$ = [-1.6146, 37.2239, -0.9824, 2.457]. To cross verify gain matrix, eigenvalues of ($\boldsymbol{A}$-$\boldsymbol{BK_{PPC}}$) can be found which will give the desired poles. The gain matrix for pole placement controller can be easily calculated using Matlab using the command $\boldsymbol{K_{PPC}}$ = place(***A***, ***B***, ***P***). After calculating gain $\boldsymbol{K_{PPC}}$, it is fed to Matlab Simulink block diagram as illustrated in Figure 4.1 to simulate the controller response.

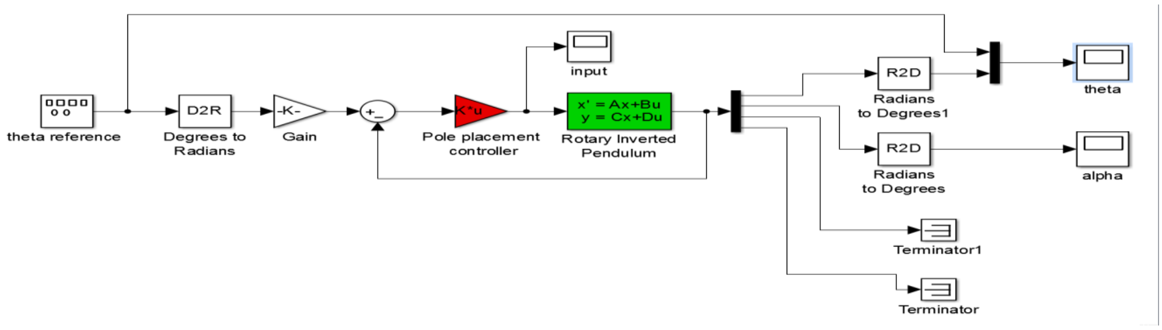

Figure 4.1: Simulink block diagram for pole placement controller

Here the controller is tested for its tracking ability where, theta reference is a square wave of amplitude 30 degree and frequency 0.125 Hz. On simulating the responses as illustrated in Figure 4.2 and Figure 4.3, it is observed that the tracking performance of designed pole placement controller is good with an overshoot of 5 degrees and settling time of 2.5 seconds for θ and maximum overshoot of 3 degrees and settling time of 2 seconds for α.

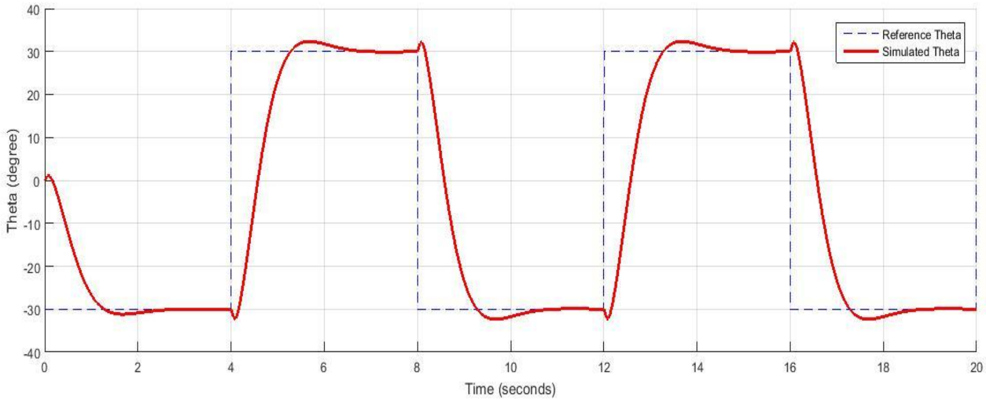

Figure 4.2: Simulation response of theta for pole placement controller

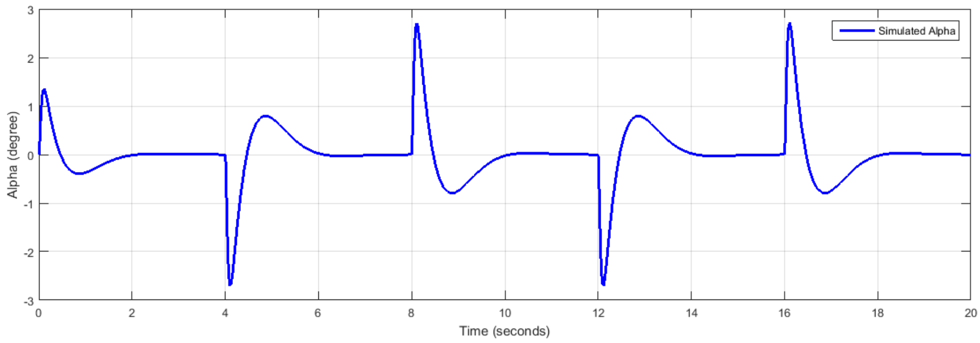

Figure 4.3: Simulation response of alpha for pole placement controller

The controller is further tested on real rotary inverted pendulum using Matlab Simulink and responses are collected as illustrated in Figure 4.4 and Figure 4.5, which shows results are in close proximity with simulation results.

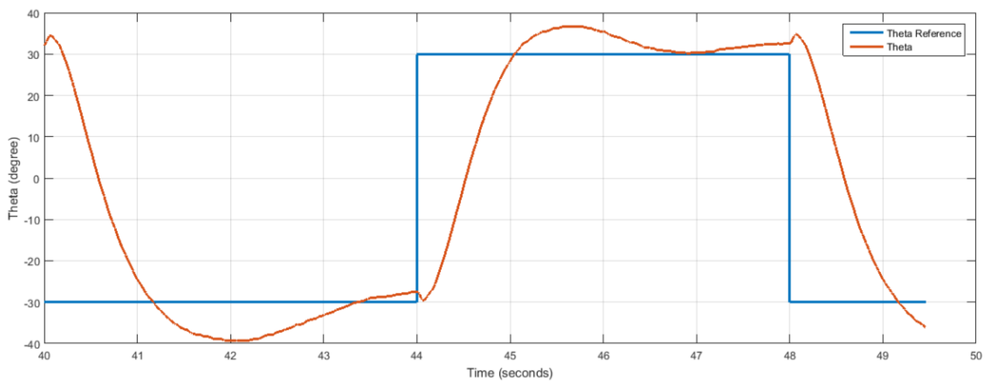

Figure 4.4: Experimental response of theta for pole placement controller

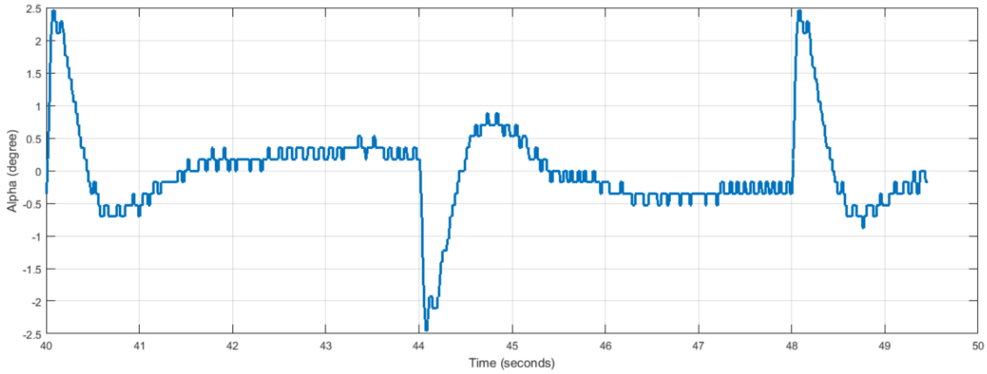

Figure 4.5: Experimental response of alpha for pole placement controller

4.4 LQR Controller

Linear Quadratic Regulator is the optimal control that is used to minimize the cost associated with generating control inputs. The cost function consists of two matrices, the state weighting matrix, $\boldsymbol{Q}$ and control cost matrix, $\boldsymbol{R}$. For a linearized plant, the objective of LQR controller is to minimize the integration of sum of these two matrices from initial to final time by using a feedback regulator. For this system with $\boldsymbol{u}$ input and $\boldsymbol{x}$ variable vector, the feedback regulator is given as:

$$\boldsymbol{u} = -\boldsymbol{K}\boldsymbol{x} \tag{4.11}$$

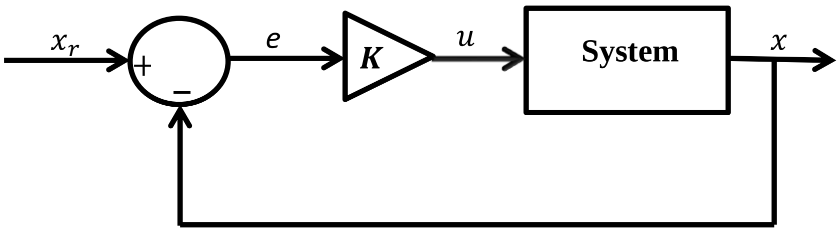

Figure 4.6: LQR Controller

And cost function i.e. the objective is given as:

$$J = \int_0^\infty (\boldsymbol{x}^T\boldsymbol{Q}\boldsymbol{x} + \boldsymbol{u}^T\boldsymbol{R}\boldsymbol{u})dt \tag{4.12}$$

Where, $\boldsymbol{K}$ is the final optimal state feedback control matrix, which is given as

$$\boldsymbol{K} = \boldsymbol{R}^{-1}\boldsymbol{B}^T\boldsymbol{P} \tag{4.13}$$

Here, $\boldsymbol{P}$ is the following algebra Symmetric positive definite solution of the equation:

$$\boldsymbol{PA} + \boldsymbol{A}^T\boldsymbol{P} - \boldsymbol{PBR}^{-1}\boldsymbol{B}^T\boldsymbol{P} + \boldsymbol{Q} = \boldsymbol{0} \tag{4.14}$$

It can be seen, gain $\boldsymbol{K}$ depends on $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{Q}$, and $\boldsymbol{R}$ matrix, $\boldsymbol{A}$ and $\boldsymbol{B}$, are determined by the structure and parameters of the system respectively, so really depends on the weight matrix $\boldsymbol{Q}$ and $\boldsymbol{R}$. Select $\boldsymbol{Q}$ and $\boldsymbol{R}$ according to desired performance criterion, the matrix $\boldsymbol{K}$ can be easily obtained by Matlab command *lqr*.

$$\boldsymbol{K} = \mathrm{lqr}(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{Q}, \boldsymbol{R}) \tag{4.15}$$

Results and Discussions

The controller given is modeled and implemented on rotary inverted pendulum using Matlab Simulink using the block diagram as shown in Figure 4.7.

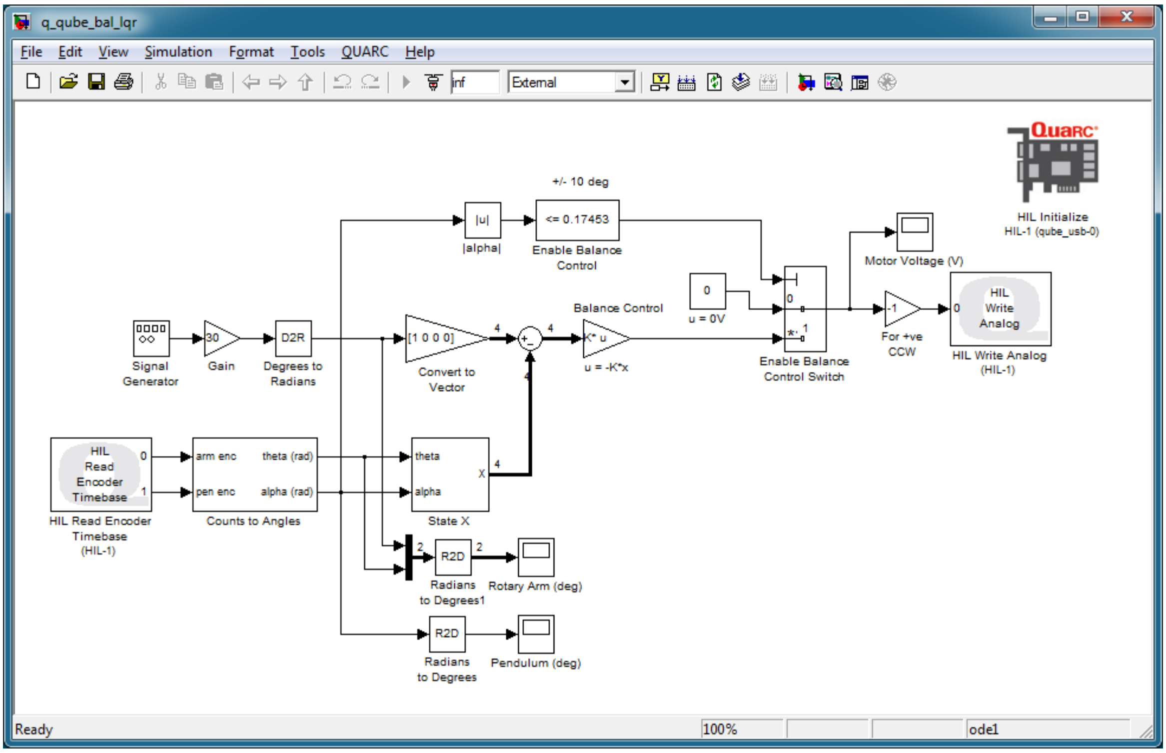

Figure 4.7: Simulink Block Diagram for LQR controller

In this model, the LQR controller is tracking a θ_{ref} signal of square type of amplitude 30 degrees and frequency of 0.125 Hz. To track the given signal, a controller is optimized by changing different values of $\boldsymbol{Q}$ (state weighing matrix) and comparing the results.

CASE 1: When $\boldsymbol{Q} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ and $\boldsymbol{R}$=1

LQR gains are calculated using Matlab command $\boldsymbol{K} = lqr(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{Q}, \boldsymbol{R})$ and it is obtained

$$\boldsymbol{K} = [-1.00, 34.24, -1.23, 3.08]$$

From response obtained as illustrated in Figure 4.8, it was observed that this controller has poor tracking performance with an error range of ± 30 degree and a steady state error of ± 10 degrees in θ while the performance for pendulum angle α is relatively good with an error range of ±2 degrees. Also while experimentation a lot of noise is observed due to high rate of change of pendulum angle.

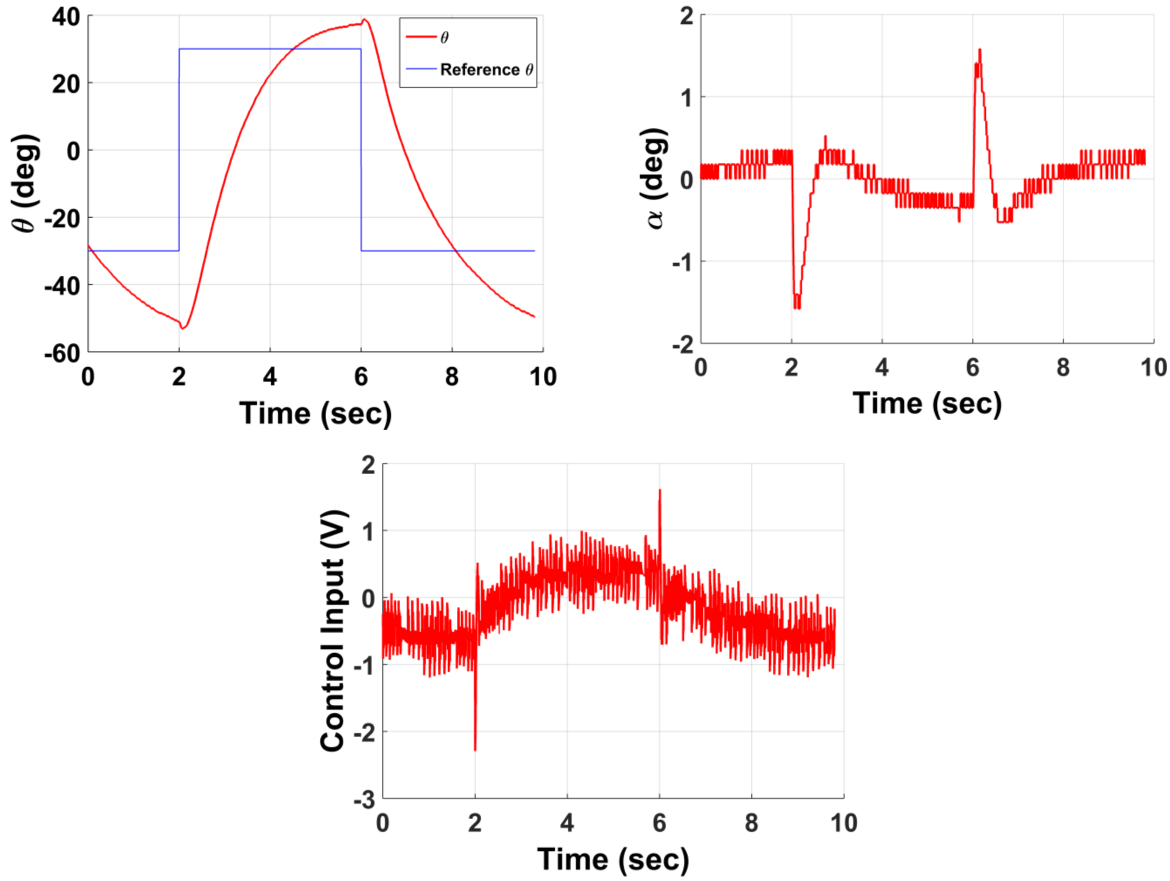

Figure 4.8: Response for LQR controller with $\boldsymbol{Q}$ = *diag*[1, 1, 1, 1]

CASE 2: When $\boldsymbol{Q} = \begin{bmatrix} 5 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$ and $\boldsymbol{R}$=1

LQR gains are calculated using Matlab command $\boldsymbol{K} = lqr(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{Q}, \boldsymbol{R})$ thus obtaining $\boldsymbol{K}$ as given:

$$\boldsymbol{K} = [-2.24, 37.62, -1.50, 3.38]$$

From the response obtained as illustrated in Figure 4.9, it was observed that this controller has better tracking performance with an error range of ± 7 degrees in θ and a relative decrease in steady state error while the performance for pendulum angle α is has been adversely affected with an error range of ± 4 degrees as more weightage was given to θ in $\boldsymbol{Q}$ matrix. Also while experimentation a lot of noise is observed due to high rate of change of pendulum angle.

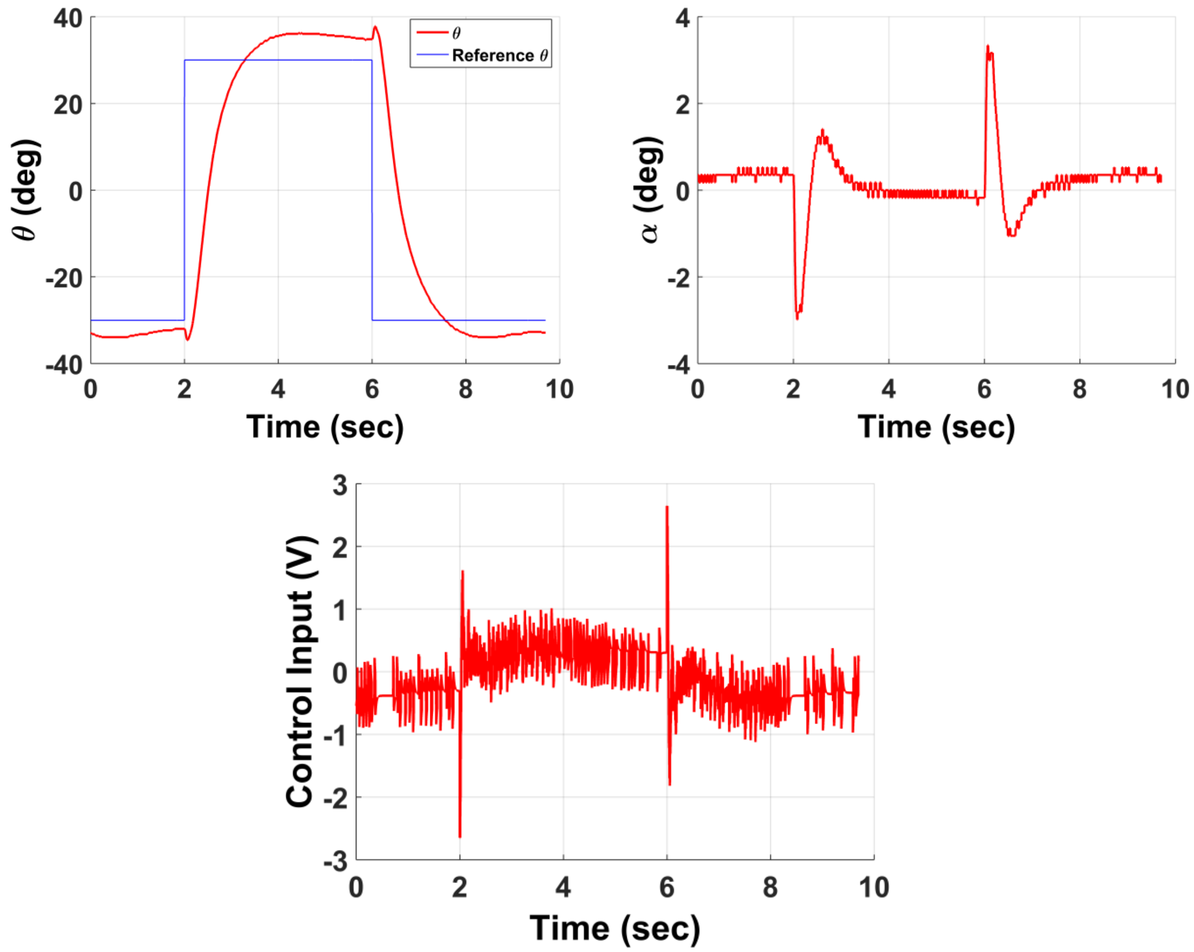

Figure 4.9: Response for LQR controller with $\boldsymbol{Q}$ = *diag*[5, 1, 1, 10]

CASE 3: When $\boldsymbol{Q} = \begin{bmatrix} 5 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 20 \end{bmatrix}$ and $\boldsymbol{R}$=1

LQR gains are calculated using Matlab command $\boldsymbol{K} = lqr(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{Q}, \boldsymbol{R})$. Thus $\boldsymbol{K}$ can be given as:

$$\boldsymbol{K} = [-2.24, 53.86, -1.70, 6.53]$$

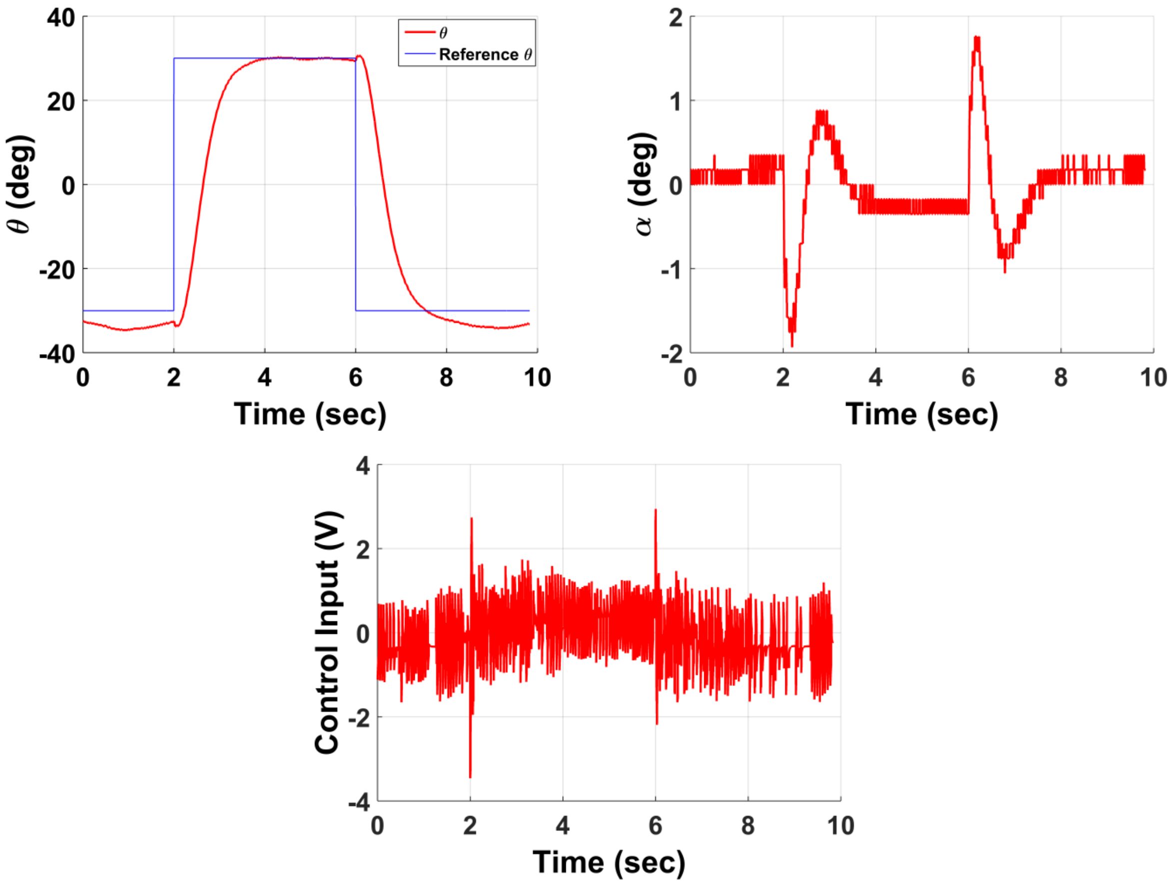

Figure 4.10: Response for LQR controller with $\boldsymbol{Q} = diag[5, 1, 1, 20]$

From the response obtained as illustrated in Figure 4.10, it was observed that in this controller, there has been a drastic decrease in vibrations of pendulum i.e. $\dot{\alpha}$ and hence leads to reduced noise while experimentation.

4.5 Comparison between Pole Placement Controller and LQR Controller

Comparative analysis between pole placement controller and LQR is done for the experimental setup and responses are obtained as illustrated in Figures 4.12-4.14.

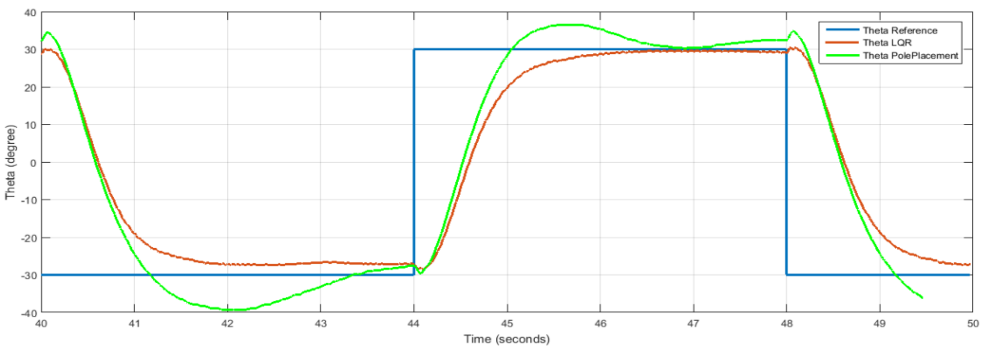

Figure 4.11: Response for theta for LQR and pole placement controller

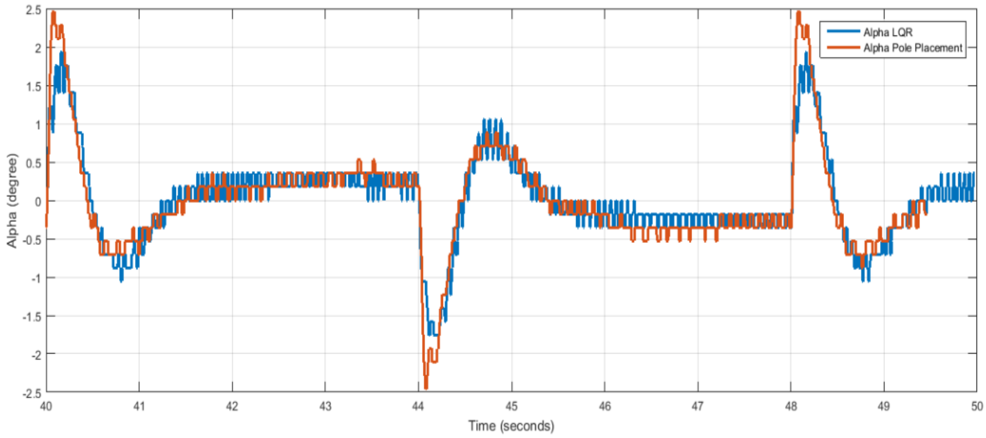

Figure 4.12: Response for alpha for LQR and pole placement controller

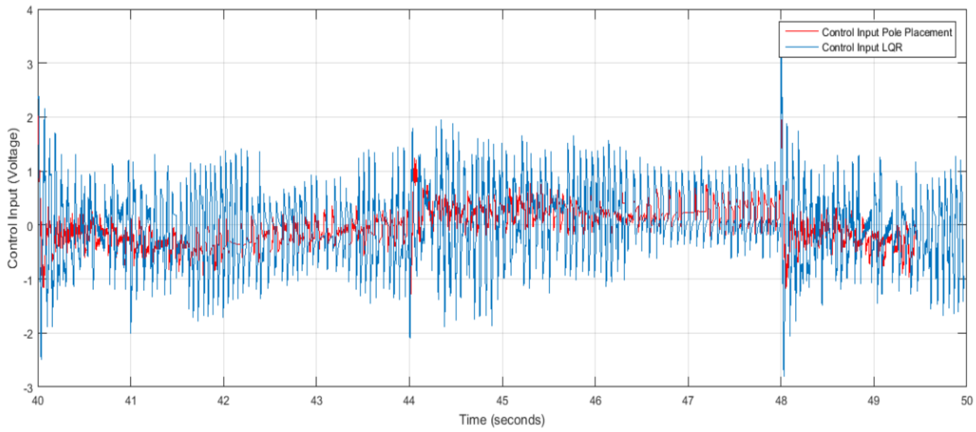

Figure 4.13: Response for control input for LQR and pole placement controller

Response obtained from comparison analysis shows that LQR performance is better for tracking of reference theta and alpha than pole placement controller. Overshoot and settling time for pole placement controller is more than LQR controller. Maximum overshoot obtained for theta for LQR controller is zero while that of pole placement controller is 10 degrees. Settling time for theta for LQR controller is 2.3 seconds while it is 2.8 seconds for pole placement controller. Maximum overshoot obtained for alpha for LQR controller is 2 degrees while that of pole placement controller is 2.5 degrees. However, control input for LQR is more than pole placement controller which can be reduced by increasing value of ***R*** i.e. control cost matrix.

4.6 Summary

This chapter presents controllability method, which has been used to check the controllability of the linearized model of rotary inverted pendulum systems. To address the stabilization problem of inverted pendulum, basic theory of two different modern controllers i.e. pole placement controller and LQR controller had been discussed in this chapter. These controllers were further tuned and implemented on rotary inverted pendulum. Further, work on robust controllers for stabilization of inverted pendulum system is presenter in Chapter 5.

Chapter 5

Robust Control

5.1 Introduction

Robust control is a method to design a controller when uncertainties are expected in the dynamic model. The controllers discussed in Chapter 4 are unable to handle parameter uncertainties. Robust controllers are designed to perform properly within a bounded range of uncertain parameters or disturbances. Robust methods aim to achieve robust performance along with stability in the presence of bounded modeling errors. In this chapter, a novel robust LQR based ANFIS controller is presented to deal with uncertainties in system parameters. This controller is a hybrid of LQR and fuzzy controller. It uses a soft computing technique named ANFIS to tune a fuzzy controller based on LQR controller. This controller is further implemented on different inverted pendulum systems to validate its effectiveness to parametric uncertainties. The steps to design robust LQR based ANFIS controller along with its implementation is discussed below.

5.2 Robust LQR based ANFIS Controller

ANFIS is a hybrid of adaptive networks and fuzzy logics. Jang [59], in 1993, presented ANFIS, which act as a platform to construct a fuzzy inference system to replicate a specified input output data set with minimum error tolerance. To optimize the parameters of fuzzy inference system, it utilises the combination of *least squares estimate* and *gradient descent method*, by maintaining a compromise between performance computational intricacies and learning performance. It inherits the Takagi and Sugeno fuzzy inference system [60] where input variables are fuzzy and the output variable is a non-fuzzy equation of input variable for example.

If *acceleration of vehicle* is *low*, then $\boldsymbol{f}(t) = k \times (acceleration\ of\ vehicle)^2$

where, *low* in premise portion of the rule denotes a linguistic fuzzy label with a suitable membership function aided by the value of *acceleration of vehicle* while the consequent portion of the rule is illustrated by an equation in accordance with input variable, *acceleration of vehicle*, of non-fuzzy behaviour.

ANFIS Architecture

ANFIS architecture naturally comprises of 5 layers. Two inputs *x* and *y* and one output *z* with 5 layers is assumed with the basis of fuzzy inference system to formulate the essential ANFIS architecture, as shown in Figure 5.1. In architecture, there are 2 kinds of nodes, a square node which gives adaptive reflection and having parameters and a circular node which signifies the characteristics of fixed node and having no parameter.

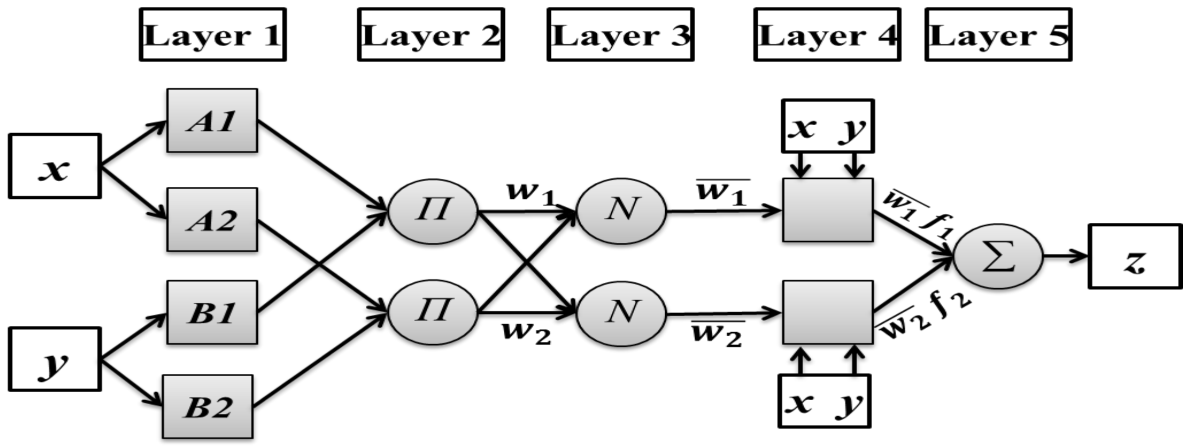

Figure 5.1: ANFIS architecture

Layer 1

This layer performs the function of converting crisp value of input variable into a linguistic fuzzy variable of sustainable membership function which is the output of this layer and illustrated as:

$$O_i^1 = \mu_{A_i}(x) \tag{5.1}$$

where, O_i^1 resembles the output of node *i* for layer *1* and $\mu_{A_i}(x)$ signifies the membership value of input variable *x* for fuzzy membership function A_i. The number and type of membership function (MF) is user defined for each input variable. Usually, generalized bell-shaped membership function is selected as it can efficiently replicate the real scenario of probability distribution function, given as

$$\mu_{A_i}(x) = \frac{1}{1 + \left[\left(\frac{x - c_i}{a_i}\right)^2\right]^{b_i}} \tag{5.2}$$

where, c_i denotes centre position for subsequent membership function; a_i is the corresponding half width; and b_i (with a_i) manage the slopes at the distinguished crossover points (where MF value is 0.5) as depicted in Figure 5.2. However, another continuous and piecewise differentiable membership function, for example, trapezoidal or triangular-shaped membership functions can also be considered.

For each input variable, optimized selections of number of membership function are done by analysing the required input-output data or by trial and error as there is no any other way to attain the desired operational efficiency with the minimisation of membership function. This layer constitutes premise parameters which are routinely arranged in a manner such that the operating domain of each input variable is supported by uniformly spaced MF's.

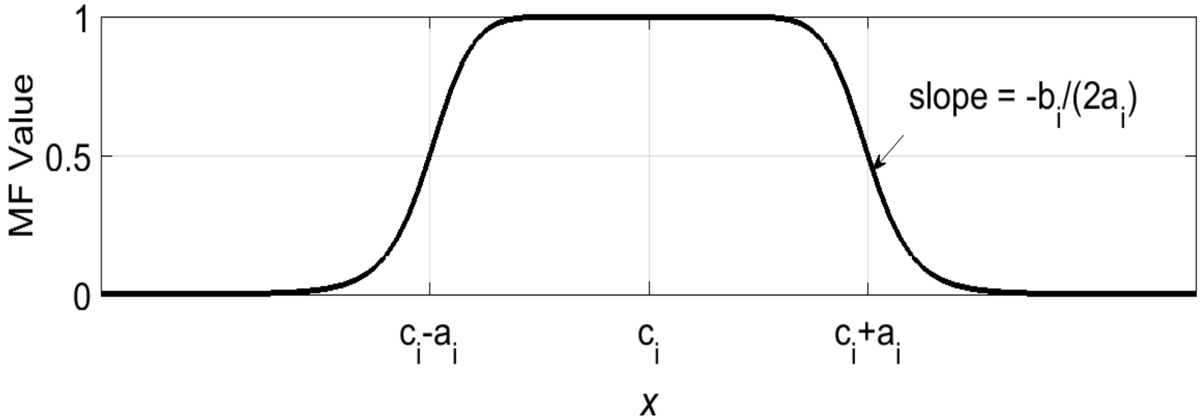

Figure 5.2: Bell shaped membership function

Layer 2

In this layer, firing strength is evaluated for each rule by multiplying all the arriving signals related to every circular node and denoted as Π. The firing strength w_i is as follows

$$O_i^2 = w_i = \mu_{A_i}(x) \times \mu_{B_i}(y), \qquad i = 1,2. \tag{5.3}$$

Layer 3

This layer estimated the normalized firing strength for each node by finding the ratio of the firing strength of rule i to the sum of all rule's firing strength. The normalized firing strength $\overline{w_i}$ is formulated as

$$O_i^3 = \overline{w_i} = \frac{w_i}{w_1 + w_2}, i = 1,2. \tag{5.4}$$

Layer 4

This layer comprises of all the square nodes with their corresponding output as

$$O_i^1 = \overline{w_i} f_i = \overline{w_i}(p_i x + q_i y + r_i) \tag{5.5}$$

Where, $\overline{w_i}$ denotes the normalized firing strength of the rule and p_i, q_i and r_i are the concerning consequent parameters for each rule.

Layer 5

In this layer, all nodes are in the circular shape having label $\sum$ which computes the complete output by summing up of all the arriving signals from previous layer i.e. layer 4, and expressed as

$$O_1^5 = \sum_i \overline{w_i} f_i = \frac{\sum_i w_i f_i}{\sum_i w_i} \tag{5.6}$$

Design of Robust LQR based ANFIS Controller

In this section, the design aspect of robust LQR based ANFIS is explained where basic procedure for the same is categorised in three steps as shown in Figure 5.3. Each step is elaborated in detail as follows:

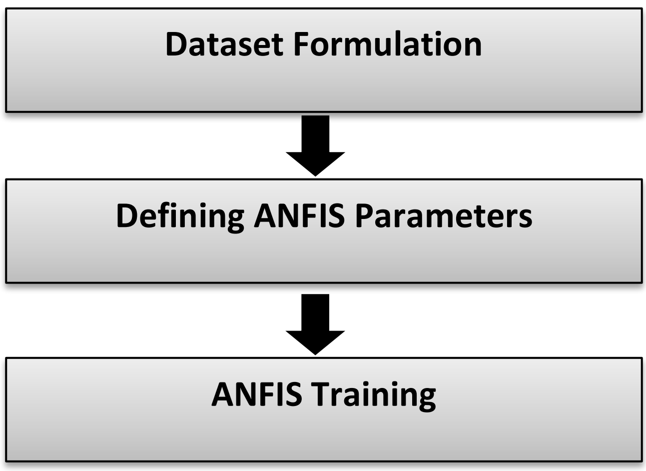

Figure 5.3: Procedure to design robust LQR based ANFIS controller

A. Dataset Formulation

In this step, formulation of dataset, given to the ANFIS, is done in the form of multiple inputs and single output column vector. To formulate a dataset, the optimal LQR controller is designed, which depend on the system parameters. The minimization of the cost function, given in (5.9), is considered as an optimization problem. The state-space representation of the requisite system for designing an LQR controller is shown below:

$$\dot{\boldsymbol{z}} = \boldsymbol{Ax} + \boldsymbol{Bu} \tag{5.7}$$

$$\boldsymbol{y} = \boldsymbol{Cx} + \boldsymbol{Du} \tag{5.8}$$

where, $\boldsymbol{x}$ signifies a state-vector, $\boldsymbol{u}$ denotes an input vector, $\boldsymbol{y}$ as an output vector and $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{C}$, $\boldsymbol{D}$ are called as the state-weighing coefficient matrices. The LQR cost function is formulated as

$$J = \frac{1}{2}\int_0^{\infty} (\boldsymbol{x^T Q x} + \ \boldsymbol{u^T R u})dt \tag{5.9}$$

where, $\boldsymbol{Q}$ is the state weighing matrix, $\boldsymbol{R}$ is the control cost matrix where both are having user defined property and employs a crucial role in the controller performance. The control input is expressed as

$$\boldsymbol{u} = \boldsymbol{Ke} = \boldsymbol{K}(\boldsymbol{x_{des}} - \boldsymbol{x}) \tag{5.10}$$

where,$\boldsymbol{e}$ represents an error vector, $\boldsymbol{x_{des}}$ is the desired state-vector and $\boldsymbol{K}$ is the final optimal state feedback control matrix, which is given as

$$\boldsymbol{K} = \ \boldsymbol{R^{-1} B^T P} \tag{5.11}$$

here, $\boldsymbol{P}$ is solution for the algebraic Riccati equation as given below:

$$\boldsymbol{PA} + \boldsymbol{A^T P} - \boldsymbol{PBR^{-1}B^T P} + \boldsymbol{Q} = \boldsymbol{0} \tag{5.12}$$

From equation (4.24), it is crystal clear that matrix $\boldsymbol{K}$ is governed by $\boldsymbol{A}$, $\boldsymbol{B}$, $\boldsymbol{Q}$, and $\boldsymbol{R}$ matrix, in which, $\boldsymbol{A}$ and $\boldsymbol{B}$ are dependent on configuration and parameters of the system. $\boldsymbol{Q}$ and $\boldsymbol{R}$ should be selected in accordance with estimated performance criterion and evaluation of matrix $\boldsymbol{K}$ is, however, done directly using a Matlab command *lqr*.

$$\boldsymbol{K} = lqr(\boldsymbol{A}, \boldsymbol{B}, \boldsymbol{Q}, \boldsymbol{R}) \tag{5.13}$$

After calculating LQR gain $\boldsymbol{K}$, a dataset is prepared for each control input and the operating range of $\boldsymbol{e}$ in the form:

$$\boldsymbol{Dataset_i} = [\boldsymbol{e^T}\ \boldsymbol{u_i}] \quad \forall \quad i = 1, 2, \dots, p \tag{5.14}$$

where, $\boldsymbol{e}$ is an error vector and $\boldsymbol{u_i}$ represents control input vector for i^{th} actuator and p is the number of control inputs. So the input for the controller will be error in state variables while the output is similar to LQR output for that error. So currently, with this dataset, ANFIS will try to optimize fuzzy inference system to follow LQR controller.

Dataset for Robust Control

To make the controller robust to parametric uncertainties, in our case, pendulum mass, the dataset is extended for a bounded set of pendulum mass for example m = (0.1, 0.45, 0.8, 1.15, 1.5) kg, where, for each mass, system matrices ($\boldsymbol{A^m}, \boldsymbol{B^m}$) are calculated. Further, LQR gain $\boldsymbol{K^m}$ is computed and thus $\boldsymbol{u_i^m}$ is calculated corresponding to error vector and is added to dataset. The extended robust dataset for each control input i is given as:

$$\boldsymbol{Robust\ dataset_i} = \begin{bmatrix} \boldsymbol{e^T} & \boldsymbol{u_i^{0.1}} \\ \boldsymbol{e^T} & \boldsymbol{u_i^{0.45}} \\ \vdots & \vdots \\ \boldsymbol{e^T} & \boldsymbol{u_i^{1.5}} \end{bmatrix} \tag{5.15}$$

where, $\boldsymbol{e} = (\boldsymbol{x_{des}} - \boldsymbol{x})$ represents an error vector and $\boldsymbol{u_i^{0.1}}, \boldsymbol{u_i^{0.45}}, \dots, \boldsymbol{u_i^{1.5}}$ denotes input vector for i^{th} actuator corresponding to pendulum mass of $0.1, 0.45, \dots, 1.5$ kg respectively.

B. Defining ANFIS Parameters

In this step, ANFIS parameters such as number and type of membership functions, error tolerance, number of epochs and learning method is specified as shown in Table 5.1

Table 5.1: ANFIS parameters

ANFIS Parameter	**Value**
Number of membership functions	2
Type of membership function	Bell-shaped
Error tolerance	Zero
Number of epochs	10
Learning method	Hybrid

C. ANFIS Training

In this step, a fuzzy inference system is trained by ANFIS by incorporating parameters given in Table 5.1 to fit in the robust dataset, where the last column of the dataset is automatically considered as output of fuzzy inference system while the other columns are considered as input of fuzzy inference system i.e., error vector is considered as input while the input vector is considered as the output of the of the fuzzy inference system as seen in (5.15). ANFIS training can be easily done in Matlab by using a pre-defined function *anfis*. Post training, a *.fis* file is created which act as a state feedback controller for the designed system.

5.3 Results and Discussions

5.3.1 Rotary Inverted Pendulum System

This section comprises of simulation and experimental results for rotary inverted pendulum. The control objective is to stabilize the pendulum at an unstable point i.e. the vertically upright position, while the rotary arm is to stabilize at exactly 30° in counter-clockwise direction. To verify the robustness of the proposed controller to parametric uncertainties, the mass of the pendulum is varied both in simulation (0.024, 0.124 and 0.224 kg) and experimentation (0.024, 0.034 and 0.044 kg) and the results obtained are compared with LQR controller.

The design and implementation of robust LQR based ANFIS controller on rotary inverted pendulum is done in following steps:

Step 1: LQR gain $\boldsymbol{K^m}$ is computed by incorporating tuning parameters $\boldsymbol{Q}$ = *diag*[5, 1, 1, 20] and $\boldsymbol{R}$ = 1, for a bounded set of pendulum masses m_p = (0.024, 0.074, 0.124, 0.174, 0.224) as follows:

$$\boldsymbol{K^{0.024}} = lqr(\boldsymbol{A^{0.024}}, \boldsymbol{B^{0.024}}, \boldsymbol{Q}, \boldsymbol{R}) \tag{5.16}$$
$$\boldsymbol{K^{0.024}} = [-2.2361 \quad 53.8740 \quad -1.7400 \quad 6.5269] \tag{5.17}$$
$$\boldsymbol{K^{0.074}} = lqr(\boldsymbol{A^{0.074}}, \boldsymbol{B^{0.074}}, \boldsymbol{Q}, \boldsymbol{R}) \tag{5.18}$$
$$\boldsymbol{K^{0.074}} = [-2.2361 \quad 65.3497 \quad -1.7754 \quad 6.8428] \tag{5.19}$$
$$\boldsymbol{K^{0.124}} = lqr(\boldsymbol{A^{0.124}}, \boldsymbol{B^{0.124}}, \boldsymbol{Q}, \boldsymbol{R}) \tag{5.20}$$
$$\boldsymbol{K^{0.124}} = [-2.2361 \quad 78.5012 \quad -1.8284 \quad 7.2572] \tag{5.21}$$
$$\boldsymbol{K^{0.174}} = lqr(\boldsymbol{A^{0.174}}, \boldsymbol{B^{0.174}}, \boldsymbol{Q}, \boldsymbol{R}) \tag{5.22}$$
$$\boldsymbol{K^{0.174}} = [-2.2361 \quad 92.9066 \quad -1.8930 \quad 7.7595] \tag{5.23}$$
$$\boldsymbol{K^{0.224}} = lqr(\boldsymbol{A^{0.224}}, \boldsymbol{B^{0.224}}, \boldsymbol{Q}, \boldsymbol{R}) \tag{5.24}$$
$$\boldsymbol{K^{0.224}} = [-2.2361 \quad 108.1907 \quad -1.9643 \quad 8.3342] \tag{5.25}$$

Step 2: Using $\boldsymbol{K}^m$, $\boldsymbol{u}_1^m$ is computed for the operating range (given in Table 5.2) in $\boldsymbol{e}$ for rotary inverted pendulum as follows:

$$\boldsymbol{u}_1^m = \boldsymbol{K}^m[\boldsymbol{e}_\theta \quad \boldsymbol{e}_\alpha \quad \boldsymbol{e}_{\dot{\theta}} \quad \boldsymbol{e}_{\dot{\alpha}}]^T \quad \forall \quad m = 0.024, 0.074, \ldots, 0.224 \tag{5.26}$$

Table 5.2: Operating range for rotary inverted pendulum

State variable	Operating range	Units
e_θ	-2 to 2	radian
$e_{\dot{\theta}}$	-1 to 1	radian/sec
e_α	-1 to 1	radian
$e_{\dot{\alpha}}$	-1 to 1	radian/sec

Step 3: A robust dataset of samples is created as given in equation (5.15), which is used for training by ANFIS using a set of Matlab command as shown in Figure 5.4.

```
numMFs = 2; % No. of Membership function for each input variable
mfType = 'gbellmf';% Type of Membership function
in_fis1 = genfis1(Robustdataset, numMFs, mfType);
%generates a Sugeno-type FIS structure used as initial conditions for anfis
anfis1 = anfis(Robustdataset, in_fis1, 10, [0,0,0,0]); % train ANFIS network
```

Figure 5.4: Matlab commands for ANFIS training

Step 4: After ANFIS training, a fuzzy inference system (anfis1.fis file) is generated in Matlab, which is the desired controller. This controller is further implemented on nonlinear mathematical model of rotary inverted pendulum and compared with LQR controller with nominal gain ($\boldsymbol{K} = \boldsymbol{K}^{0.024}$) using Matlab Simulink block diagram as shown in Figure 5.5. Further, simulation results are obtained for different pendulum masses as shown in Figures 5.6-5.7.

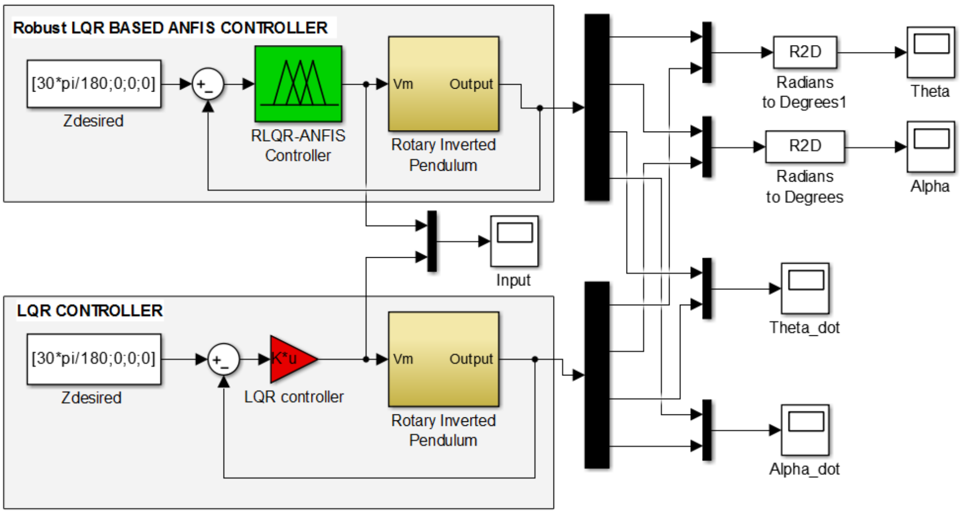

Figure 5.5: Simulink block diagram for comparison between RLQR-ANFIS controller and LQR controller

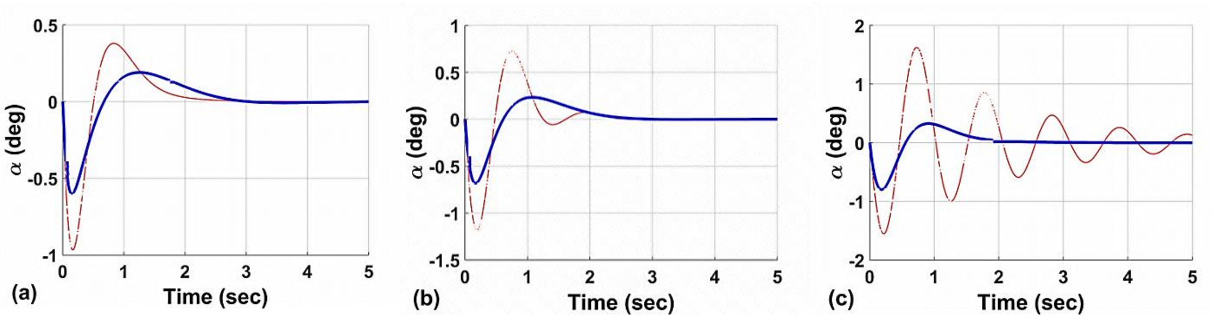

Figure 5.6: Simulation response of α for pendulum mass (m_p) in kg = a) 0.024; b) 0.124; c) 0.224 (solid blue line: RLQR-ANFIS controller and dashed red line: LQR controller)

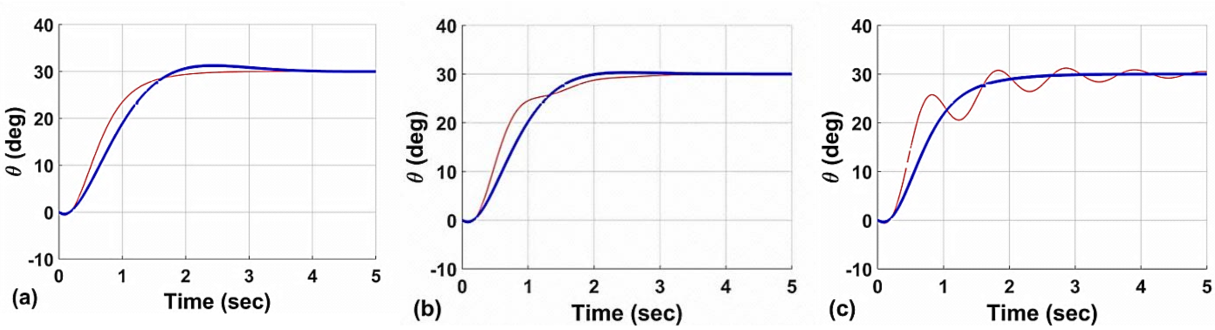

Figure 5.7: Simulation response of θ for pendulum mass (m_p) in kg = a) 0.024; b) 0.124; c) 0.224 (solid blue line: RLQR-ANFIS and dashed red line: LQR controller)

As seen from the simulation result obtained in Figures 5.6-5.7, it can be observed that although for nominal mass, some of the performance parameters has degraded but as the mass is increased proposed controller gives better performance than LQR controller, which shows that the proposed controller is robust to pendulum mass while LQR controller performance degrades as the mass is increased. The detailed information on performance parameters such as settling time of pendulum angle (T_s) and maximum overshoot in pendulum angle (α_{max}), and rotary arm angle (θ_{max}) is given in Table 5.3. Moreover, on further increasing the pendulum mass to 0.26 kg, the LQR controller was unable to stabilize the inverted pendulum (as the amplitude is increasing), while the RLQR-ANFIS controller have insignificant impact on the performance of the system as seen in Figure 5.8.

Table 5.3: Performance parameters comparison for different pendulum masses on rotary inverted pendulum

Pendulum Mass (m_p)	**Performance Parameters**	**Units**	**RLQR-ANFIS**	**LQR**	**Percentage Reduction**
0.024	α_{max}	degree	0.1903	0.3803	49.96
	θ_{max}	degree	31.2204	30	-4.07
	T_s	second	2.7717	2.1579	-28.44
0.124	α_{max}	degree	0.2331	0.7249	67.84
	θ_{max}	degree	30.3079	30	-1.03
	T_s	second	2.5409	2.3065	-10.16
0.224	α_{max}	degree	0.3248	1.6238	80.00
	θ_{max}	degree	30	31.2075	3.87
	T_s	second	2.3735	7.6411	68.94

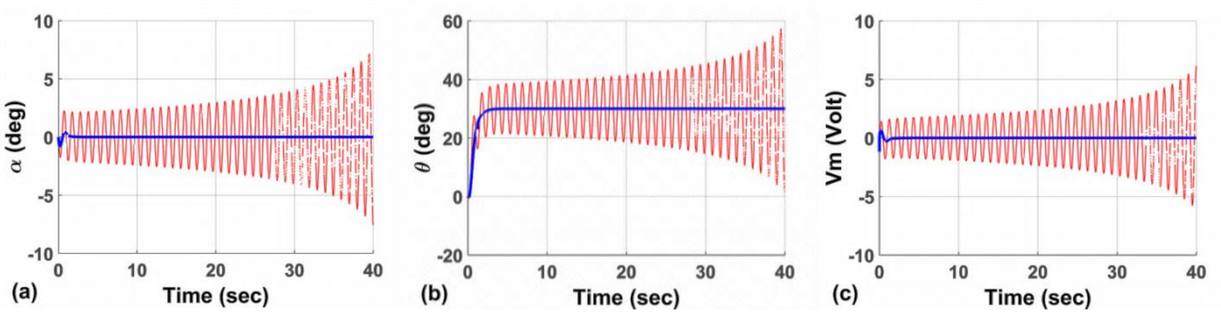

Figure 5.8: Simulation response of a) α b) θ c) input V_m for pendulum mass m_p = 0.26 kg (solid blue line: RLQR-ANFIS and red dashed line: LQR controller)

Step 6: Similarly, experimental results are obtained on experimental setup of rotary inverted pendulum as shown in Figures 5.9-5.10. Although, the variation in pendulum mass is limited to 0.044 kg due to experimental limitation of setup. It can be observed that similar trends are observed in experimentation as in simulation. Further comparison of simulation and experimental results are done as shown in Figure 5.11.

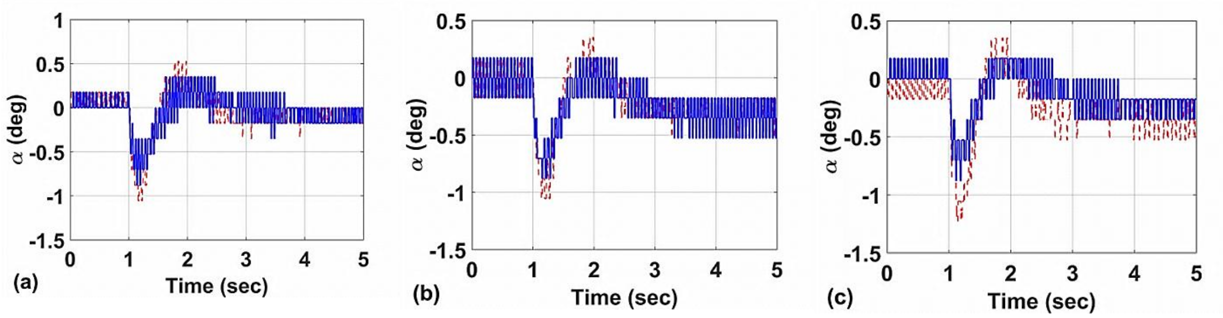

Figure 5.9: Experimental response of α for pendulum mass (m_p) in kg = a) 0.024; b) 0.034; c) 0.044 (solid blue line: RLQR-ANFIS controller and red dashed line: LQR controller)

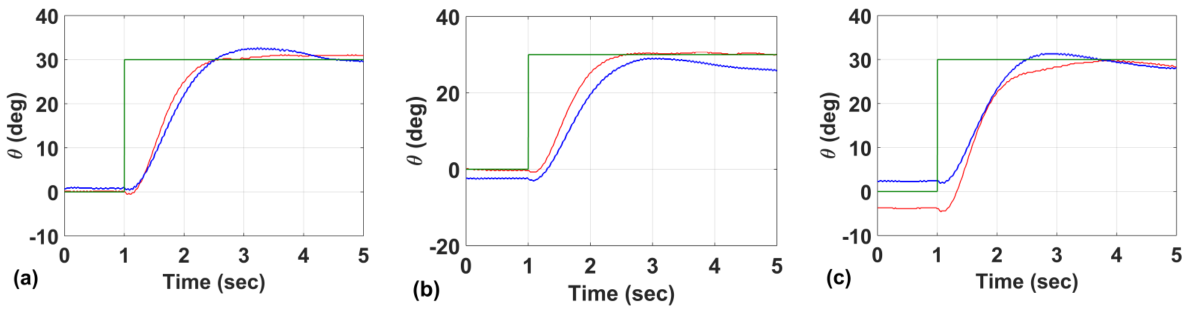

Figure 5.10: Experimental response of θ for mass (m_p) in kg = a) 0.024; b) 0.034; c) 0.044 (blue line: RLQR-ANFIS controller, red line: LQR controller and green line: desired θ)

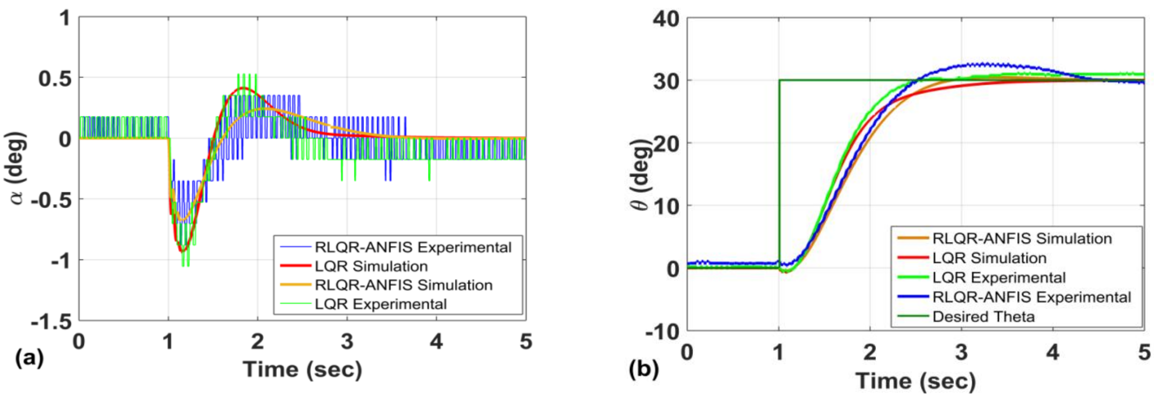

Figure 5.11: Simulation and experimental response of a) α b) θ for pendulum mass m_p = 0.024 kg

As seen in Figure 5.11, the experimental results are in close proximity to simulation results and hence we can rely on simulation results for higher pendulum mass variations and thus results can be concluded accordingly.

5.3.2 Linear Inverted Pendulum Systems

In this section, simulation results are presented for linear inverted pendulum systems. The control objective in all the simulations is to stabilize the pendulum at an unstable equilibrium position i.e. the upright position, while the cart is also supposed to move exactly by 1 m distance in *x*-, *y*- and *z*-direction depending upon the type of the inverted pendulum system. To induce parametric variations, the pendulum masses are varied within the bounded range i.e. (0.1–0.5) kg in the case of *x*, *x-y* and *x-z* inverted pendulum systems and (0.1–1.5) kg in the case of *x-y-z* inverted pendulum system. The results obtained using RLQR-ANFIS controller are compared with the conventional stabilization controllers i.e. PID or LQR controller for different pendulum masses.

A) *x* Inverted Pendulum: Using $\boldsymbol{Q} = diag[5, 1, 1, 10]$ and $\boldsymbol{R} = 1$, LQR gains are calculated for a bounded set of pendulum masses i.e, m = (0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5) kg and a robust dataset is prepared for an operating range of *XIP* as given in Table 5.4.A dataset of 5625 samples is generated, which is used for training by ANFIS. Thus, a fuzzy inference system with 16 (2^4) rules is obtained, which is implemented on the *XIP* and simulation results are obtained. Using the Simulunk block diagram, as shown in Figure 5.12, the results obtained are compared with conventional PID controller [25], for different masses as shown in Figures 5.13–5.15.

Table 5.4: Operating range of state variables for *x*, *x-y*, *x-z* inverted pendulum

Error in state variable	Operating range	Units	Pendulum type
e_x	-2 to 2	meter	*x*, *x-y*, *x-z*
$e_{\dot{x}}$	-1 to 1	meter/sec	
e_θ	-1 to 1	radian	
$e_{\dot{\theta}}$	-1 to 1	radian/sec	
e_z	-2 to 2	meter	*x-z*
$e_{\dot{z}}$	-1 to 1	meter/sec	
e_y	-2 to 2	meter	*x-y*
$e_{\dot{y}}$	-1 to 1	meter/sec	
e_ϕ	-1 to 1	radian	
$e_{\dot{\phi}}$	-1 to 1	radian/sec	

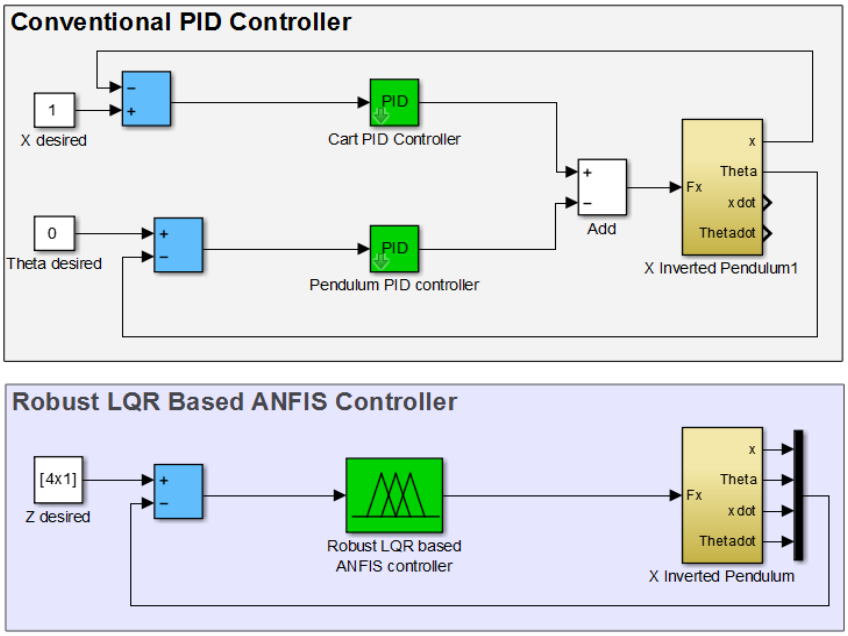

Figure 5.12: Simmulink block diagram for comparison between conventional PID controller and robust LQR rased ANFIS controller

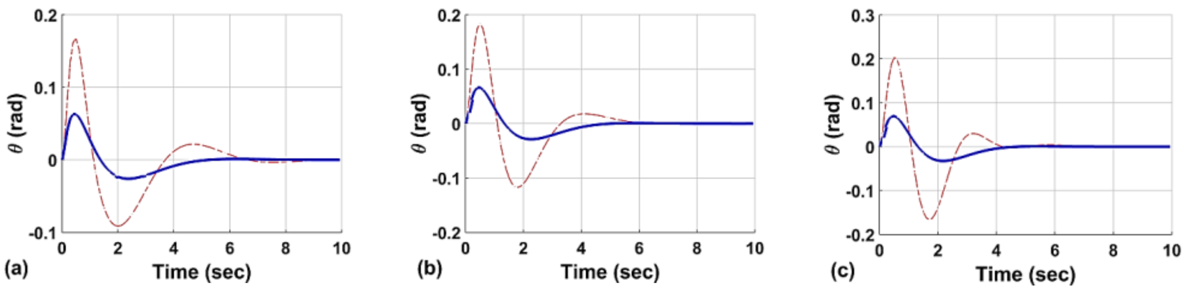

Figure 5.13: Response of θ for mass (m) in kg = a) 0.1; b) 0.3; c) 0.5(solid line: RLQR-ANFIS controller and dashed line: conventional PID controller)

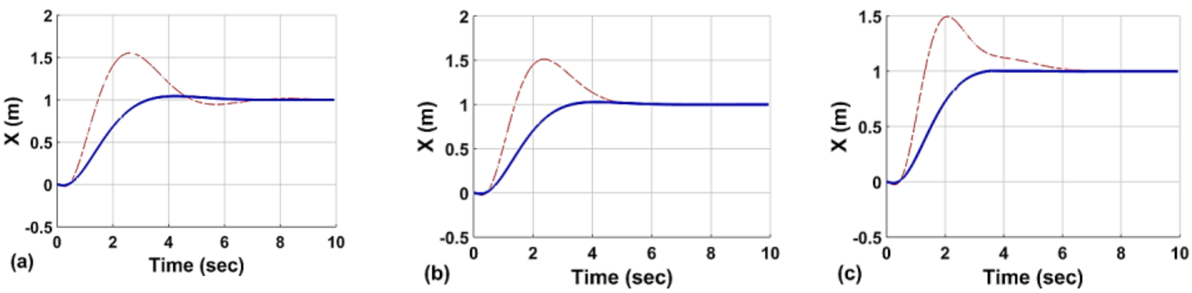

Figure 5.14: Response of x for mass (m) in kg = a) 0.1; b) 0.3; c) 0.5 (solid line: RLQR-ANFIS controller and dashed line: conventional PID controller)

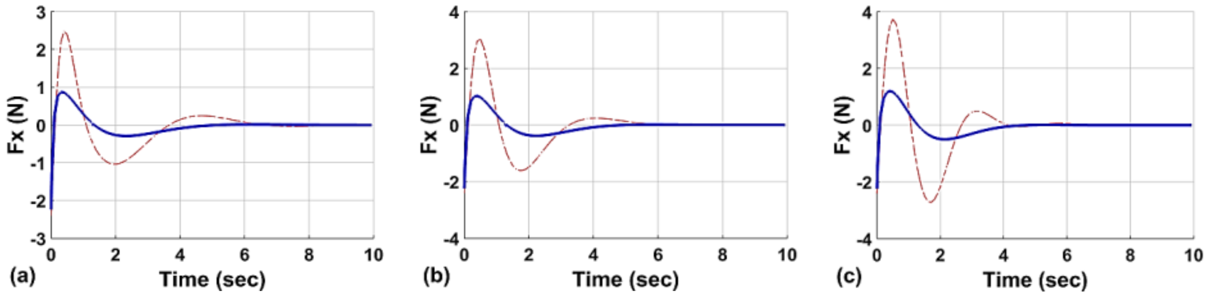

Figure 5.15: Response of F_x input for mass (m) in kg = a) 0.1; b) 0.3; c) 0.5 (solid line: RLQR-ANFIS controller and dashed line: conventional PID controller)

From the simulation results obtained in Figures 5.13–5.15, it is observed that with conventional PID controller for pendulum mass of 0.1 kg, the pendulum settling time (T_S) and the maximum overshoot in pendulum position (θ_{max}), cart position (x_{max}) and the input force ($F_{x_{max}}$) is found as 8.14 sec, 0.16 rad, 1.55 m and 2.48 N respectively, whereas using the proposed controller, the corresponding values are found as 4.90 sec, 0.06 rad, 1.04 m and 0.87 N. This shows that as compared with conventional PID controller, the proposed controller have shown a reduction of 39.71%, 62.23%, 32.86% and 65.03% in the corresponding parameters. Similar trends are observed for pendulum mass of 0.3 kg and 0.5 kg as given in Table 5.5.

Further, it can be concluded that RLQR-ANFIS controller outperformed the conventional PID controller for *XIP* and is also robust to variations in pendulum mass. Also, on further increasing the pendulum mass to 0.8 kg, it is observed that the conventional PID controller is unable to stabilize the inverted pendulum, while the RLQR-ANFIS controller has an insignificant impact on the performance of system as shown in Figure 5.16.

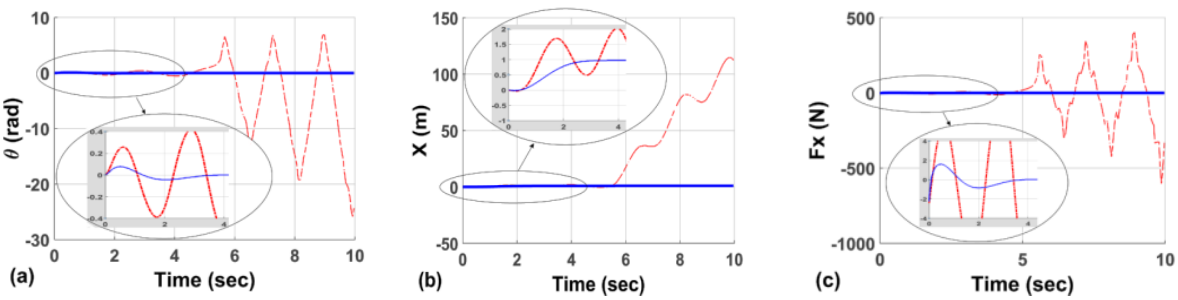

Figure 5.16: Response of a) pendulum angle θ ; b) x; c) input (F_x) for m = 0.8 kg (solid line: RLQR-ANFIS controller and dashed line: conventional PID controller)

Table 5.5: Performance parameters for different pendulum masses on *XIP*

Pendulum Mass (m)	**Performance Parameters**	**Units**	**RLQR-ANFIS**	**Conventional PID**	**Percentage Reduction**
0.1	θ_{max}	Radian	0.0630	0.1668	62.23 %
	x_{max}	Meter	1.0426	1.5528	32.86 %
	$F_{x_{max}}$	Newton	0.8703	2.4890	65.03 %
	T_s	Second	4.9091	8.1419	39.71 %
0.3	θ_{max}	Radian	0.0662	0.1841	64.04 %
	x_{max}	Meter	1.0278	1.5112	31.99 %
	$F_{x_{max}}$	Newton	1.0257	3.0148	65.98 %
	T_s	Second	4.6773	5.6320	16.95 %
0.5	θ_{max}	Radian	0.0697	0.2034	65.73 %
	x_{max}	Meter	1.0115	1.4964	32.40 %
	$F_{x_{max}}$	Newton	1.2036	3.7365	67.79 %
	T_s	Second	4.3971	5.9526	26.13 %

B) *x-z* Inverted Pendulum: As compared to *XIP*, the *XZIP* have one additional degree of freedom (DOF), as the cart can move in *x-z* plane.This 3-DOF system is controlled with two actuators in *x* and *z* direction, with one degree of underactuation. This system is more challenging to control as compared to the previous one, due to the impact of gravity as the cart moves in *z*-direction. To design a controller, LQR gains are computed for a bounded set of pendulum mass *m* = (0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5) kg considering LQR tuning parameters, $\boldsymbol{Q}$ = *diag*[5, 1, 5, 1, 1, 10] and $\boldsymbol{R}$ = *diag*[1, 1]. Further control input is calculated using (5.27) for an operating range of *XZIP* as given in Table 5.4.

$$\boldsymbol{u} = [\,\boldsymbol{u_1}\,\boldsymbol{u_2}]^T = \boldsymbol{Ke} \tag{5.27}$$

where,

$$\boldsymbol{u} = [\boldsymbol{u_1}\,\boldsymbol{u_2}]^T = [\boldsymbol{F_x}\;\;\boldsymbol{F_z} - g(M+m)]^T \tag{5.28}$$

Furthermore, two datasets, one of 5625 samples while the other of 225 samples are generated, which are used for training by ANFIS. Thus, two fuzzy inference system, one with 16 (2^4) rules for controlling x position and pendulum angle while the other with 4 (2^2) rules for controlling z position are obtained. As it can be seen in (5.28), the term $(-g(M+m))$ appears in $\boldsymbol{u_2}$, thus gravity compensation is provided in the controller by adding $g(M+m)$ term to the output of the lateral fuzzy inference system. Further, the obtained fuzzy inference systems are implemented on *XZIP* and simulation results are obtained. The results obtained are compared with conventional PID controller [25] for different pendulum masses as shown in Figures 5.17–5.21.

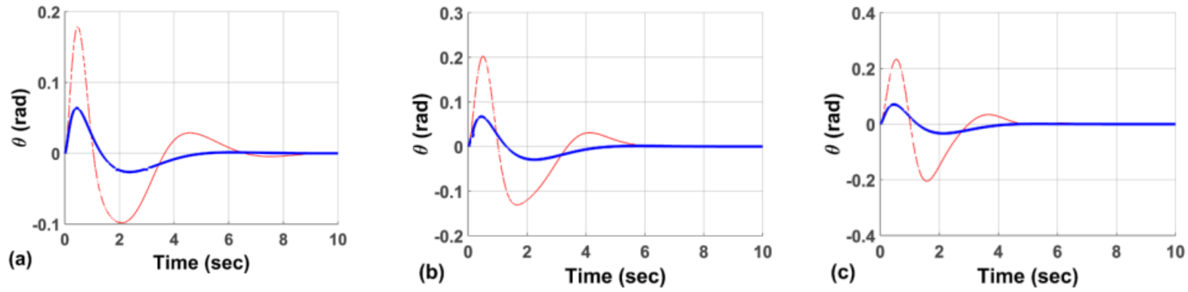

Figure 5.17: Response of θ for mass (m) in kg = a) 0.1; b) 0.3; c) 0.5 (solid line: RLQR-ANFIS controller and dashed line: conventional PID controller)

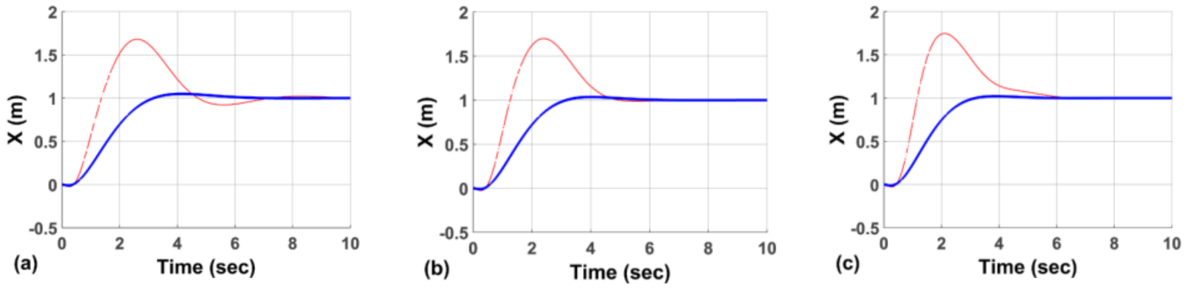

Figure 5.18: Response of x for mass (m) in kg = a) 0.1; b) 0.3; c) 0.5 (solid line: RLQR-ANFIS controller and dashed line: conventional PID controller)

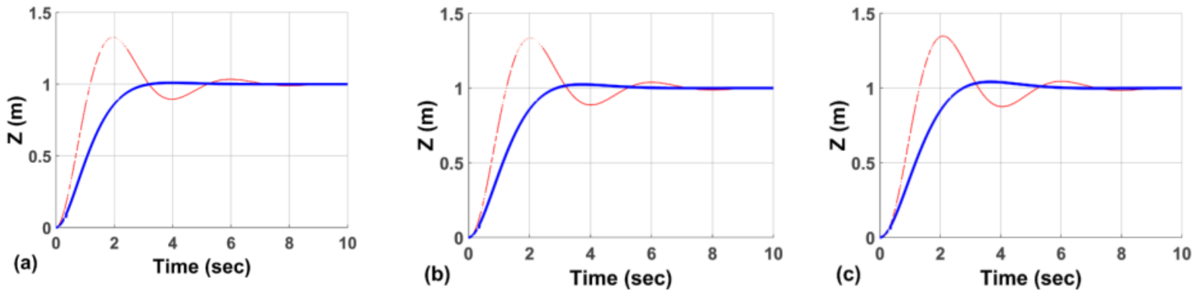

Figure 5.19: Response of z for mass (m) in kg = a) 0.1; b) 0.3; c) 0.5 (solid line: RLQR-ANFIS controller and dashed line: conventional PID controller)

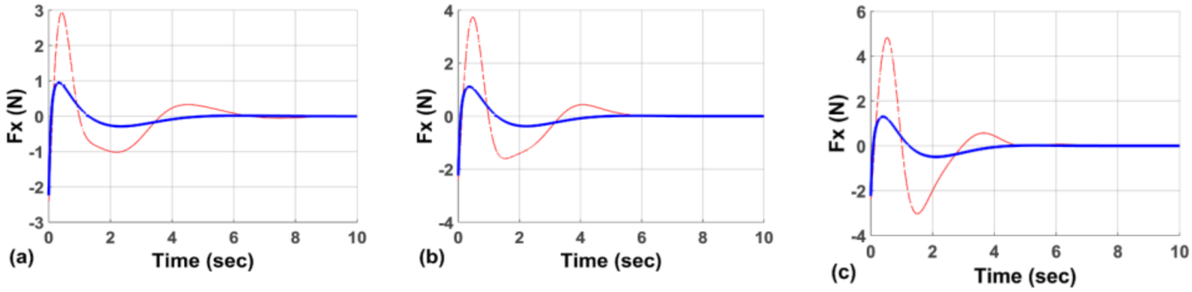

Figure 5.20: Response of input F_x for mass (m) in kg = a) 0.1; b) 0.3; c) 0.5 (solid line: RLQR-ANFIS controller and dashed line: conventional PID controller)

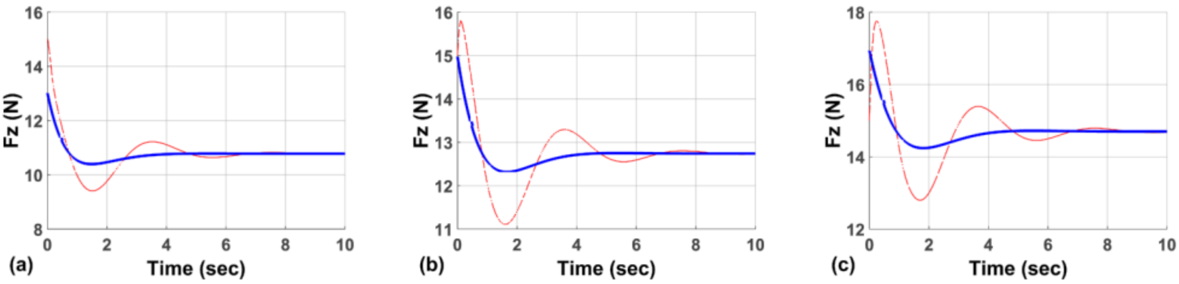

Figure 5.21: Response of input F_z for mass (m) in kg = a) 0.1; b) 0.3; c) 0.5 (solid line: RLQR-ANFIS controller and dashed line: conventional PID controller)

From the simulation results obtained in Figures 5.17–5.21, a significant reduction in performance parameters such as the pendulum settling time (T_s) and the maximum overshoot in pendulum position (θ_{max}), cart position in x (x_{max}) and z (z_{max}) direction and the input forces ($F_{x_{max}}$ and $F_{z_{max}}$) is observed for different pendulum masses as reported in Table 5.6. Thus, it can be concluded that controller performance and robustness is much better for RLQR-ANFIS controller than conventional PID controller.

***C)* x-y Inverted Pendulum:** As compared to x and x-z inverted pendulums, the *XYIP* is a highly non-linear system. It is a 4-DOF system, which is controlled with two actuators in x and y direction, with two degree of underactuation. This system is more challenging to control as compared to the previous cases, as the pendulum can also fall in the lateral direction.To control this challenging system, LQR gains are calculated using $\boldsymbol{Q} = diag[5, 1, 5, 1, 1, 10, 1, 10]$ and $\boldsymbol{R} = diag[1, 1]$, for a bounded set of pendulum mass m = (0.1, 0.15, 0.2, 0.25, 0.3, 0.35, 0.4, 0.45, 0.5) kg and for an operating range as specified in Table 5.4. Two robust datasets of 5625 samples each, are generated and are futher used for training by ANFIS. Thus, two fuzzy inference system, one for controlling x position and θ pendulum angle while the other for controlling y position and ϕ pendulum angle are obtained with 16 (2^4) rules in each. Further, the obtained fuzzy inference systems are implemented on *XYIP* and simulation results obtained are compared with conventional PID controller [25] for different pendulum masses, as shown in Figures 5.22–5.24.

Table 5.6: Performance parameters for different pendulum masses on *XZIP*

Pendulum Mass (m)	Performance Parameters	Units	RLQR-ANFIS	Conventional PID	Percentage Reduction
0.1	θ_{max}	Radian	0.0640	0.1789	64.23 %
	x_{max}	Meter	1.0481	1.6804	37.63 %
	z_{max}	Meter	1.0091	1.3272	23.97 %
	$F_{x_{max}}$	Newton	0.9499	2.9320	67.60 %
	$F_{z_{max}}$	Newton	13.0161	15.0021	13.24 %
	T_s	Second	6.4820	8.3829	22.68 %
0.3	θ_{max}	Radian	0.0672	0.2024	66.80 %
	x_{max}	Meter	1.0351	1.6991	39.08 %
	z_{max}	Meter	1.0240	1.3353	23.31 %
	$F_{x_{max}}$	Newton	1.1145	3.7431	70.23 %
	$F_{z_{max}}$	Newton	14.9761	15.8050	5.24 %
	T_s	Second	4.6198	5.6652	18.45 %
0.5	θ_{max}	Radian	0.0708	0.2323	69.52 %
	x_{max}	Meter	1.0206	1.7471	41.58 %
	z_{max}	Meter	1.0413	1.3481	22.76 %
	$F_{x_{max}}$	Newton	1.3016	4.8253	73.03 %
	$F_{z_{max}}$	Newton	16.9361	17.7528	4.60 %
	T_s	Second	4.3547	4.6364	6.08 %

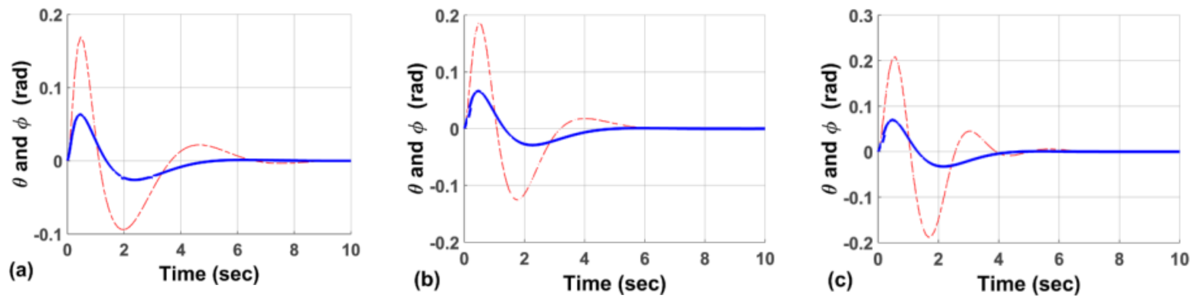

Figure 5.22: Response of θ and ϕ for mass (m) in kg = a) 0.1; b) 0.3; c) 0.5 (solid line: RLQR-ANFIS controller and dashed line: conventional PID controller)

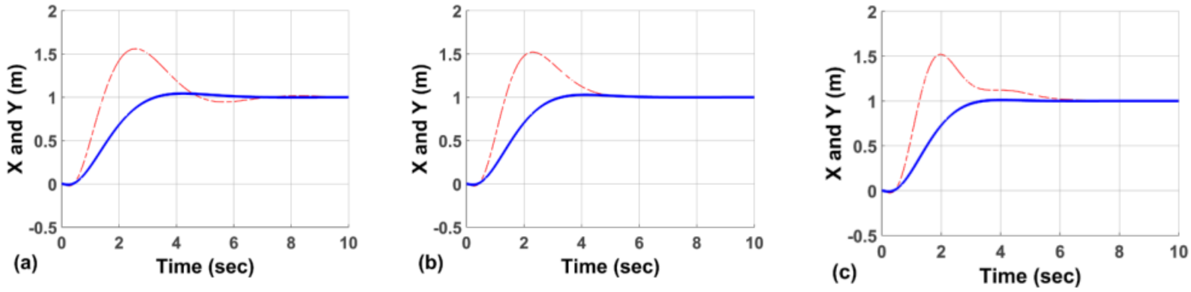

Figure 5.23: Response of x and y position for mass (m) in kg = a) 0.1; b) 0.3; c) 0.5 (solid line: RLQR-ANFIS controller and dashed line: conventional PID controller)

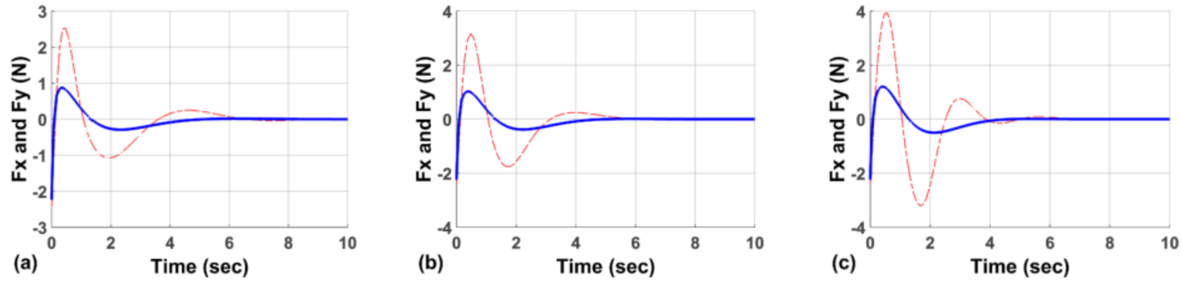

Figure 5.24: Response of input F_x and F_y for mass (m) in kg = a) 0.1; b) 0.3; c) 0.5 (solid line: RLQR-ANFIS controller and dashed line: conventional PID controller)

Table 5.7: Performance parameters for different pendulum masses on *XYIP*

Pendulum Mass (*m*)	**Performance Parameters**	**Units**	**RLQR-ANFIS**	**Conventional PID**	**Percentage Reduction**
0.1	θ_{max} / ϕ_{max}	radian	0.0635	0.1693	62.49 %
	x_{max} / y_{max}	meter	1.0424	1.5598	33.17 %
	$F_{x_{max}}$ / $F_{y_{max}}$	Newton	0.8735	2.5284	65.45 %
	T_s	second	4.9041	8.1026	39.47 %
0.3	θ_{max} / ϕ_{max}	radian	0.0666	0.1858	64.16 %
	x_{max} /y_{max}	meter	1.0276	1.5212	32.45 %
	$F_{x_{max}}$ /$F_{y_{max}}$	Newton	1.0326	3.1658	67.38 %
	T_s	second	4.6704	5.5518	15.88 %
0.5	θ_{max} / ϕ_{max}	radian	0.0699	0.2088	66.52 %
	x_{max} / y_{max}	meter	1.0113	1.5224	33.57 %
	$F_{x_{max}}$ /$F_{y_{max}}$	Newton	1.2015	3.9377	69.49 %
	T_s	second	4.3895	6.0905	27.93 %

From Figures 5.22–5.24, similar trends can be observed in *XYIP* as in *x*, and *x-z* cases. In this case also, the LQR based ANFIS controller has outperformed the conventional PID controller and achieved a significant reduction in performance parameters such as the pendulum settling time (T_s) and the maximum overshoot in pendulum position (θ_{max} and ϕ_{max}), cart position in x (x_{max}) and y (y_{max}) direction and the input forces ($F_{x_{max}}$ and $F_{y_{max}}$) for different pendulum masses as given in Table 5.7.

As reported in XIP, similar behavior is observed in XYIP and XZIP when the pendulum mass is increased to 0.65 kg and 0.8 kg respectively. That is, the conventional PID controller is unable to stabilize the pendulum, whereas the proposed controller give satisfactory results.

D) *x-y-z* Inverted Pendulum: The *x-y-z* inverted pendulum is the generalized case of linear inverted pendulum systems. It can be considered as combination of *x-y* and *x-z* inverted pendulum system, thus, the complexity of system is far more than any other linear inverted pendulum system. The robust LQR based ANFIS controller is designed and implemented on *x-y-z* inverted pendulum in following steps:

Step 1: LQR gain $\boldsymbol{K^m}$ is calculated using tuning parameters $\boldsymbol{Q}$ = *diag*[5, 1, 5, 1, 5, 1, 1, 20, 1, 20] and $\boldsymbol{R}$ = *diag*[1, 1, 1] for a bounded set of pendulum masses *m* = (0.1, 0.45, 0.8, 1.15, 1.5) as follows:

$$\boldsymbol{K^{0.1}} = lqr(\boldsymbol{A^{0.1}}, \boldsymbol{B^{0.1}}, \boldsymbol{Q}, \boldsymbol{R}) \tag{5.29}$$

$$\boldsymbol{K^{0.45}} = lqr\left(\boldsymbol{A^{0.45}}, \boldsymbol{B^{0.45}}, \boldsymbol{Q}, \boldsymbol{R}\right) \tag{5.30}$$

$$\boldsymbol{K^{0.8}} = lqr(\boldsymbol{A^{0.8}}, \boldsymbol{B^{0.8}}, \boldsymbol{Q}, \boldsymbol{R}) \tag{5.31}$$

$$\boldsymbol{K^{1.15}} = lqr\left(\boldsymbol{A^{1.15}}, \boldsymbol{B^{1.15}}, \boldsymbol{Q}, \boldsymbol{R}\right) \tag{5.32}$$

$$\boldsymbol{K^{1.5}} = lqr\left(\boldsymbol{A^{1.5}}, \boldsymbol{B^{1.5}}, \boldsymbol{Q}, \boldsymbol{R}\right) \tag{5.33}$$

Step 2: Further, $\boldsymbol{u^m}$ is calculated form $\boldsymbol{K^m}$ for the operating range in $\boldsymbol{e}$ (given in Table 5.8) for *x-y-z* inverted pendulum as follows:

$$\boldsymbol{u^m} = \boldsymbol{K^m}[\boldsymbol{e_x}\ \boldsymbol{e_{\dot{x}}}\ \boldsymbol{e_y}\ \boldsymbol{e_{\dot{y}}}\ \boldsymbol{e_z}\ \boldsymbol{e_{\dot{z}}}\ \boldsymbol{e_\theta}\ \boldsymbol{e_{\dot{\theta}}}\ \boldsymbol{e_\phi}\ \boldsymbol{e_{\dot{\phi}}}]^T \quad \forall \quad m = (0.1, 0.45, 0.8, 1.15, 1.5) \tag{5.34}$$

where,

$$\boldsymbol{u^m} = [\boldsymbol{u_1^m}\ \ \boldsymbol{u_2^m}\ \ \boldsymbol{u_3^m}]^T = [\boldsymbol{F_x^m}\ \ \boldsymbol{F_y^m}\ \ \boldsymbol{F_z^m} - g(M+m)]^T \tag{5.35}$$

Table 5.8: Operating range for *x-y-z* inverted pendulum

Error in state variable	**Operating range**	**Units**
e_x	-2 to 2	meter
$e_{\dot{x}}$	-1 to 1	meter/sec
e_y	-2 to 2	meter
$e_{\dot{y}}$	-1 to 1	meter/sec
e_z	-2 to 2	meter
$e_{\dot{z}}$	-1 to 1	meter/sec
e_θ	-1 to 1	radian
$e_{\dot{\theta}}$	-1 to 1	radian/sec
e_ϕ	-1 to 1	radian
$e_{\dot{\phi}}$	-1 to 1	radian/sec

Step 3: Three robust datasets of 9375, 9375 and 375 samples respectively are generated (one for each actuator) in the format given in (5.15). Three ANFIS trainings are done using these robust datasets by a set of Matlab command as given in Figure 5.25.

```
numMFs = 2; % No. of Membership function for each input variable
mfType = 'gbellmf';% Type of Membership function
in_fis1 = genfis1(Robustdataset1, numMFs, mfType);
%generates a Sugeno-type FIS structure used as initial conditions for anfis
anfis1 = anfis(Robustdataset1, in_fis1, 10, [0,0,0,0]); % train first ANFIS
in_fis2 = genfis1(Robustdataset2, numMFs, mfType);
%generates a Sugeno-type FIS structure used as initial conditions for anfis
anfis2 = anfis(Robustdataset2, in_fis2, 10, [0,0,0,0]); % train second ANFIS
in_fis3 = genfis1(Robustdataset3, numMFs, mfType);
%generates a Sugeno-type FIS structure used as initial conditions for anfis
anfis3 = anfis(Robustdataset3, in_fis3, 10, [0,0,0,0]); % train third ANFIS
```

Figure 5.25: Matlab commands for ANFIS training

Step 4: Three fuzzy inference systems (anfis1.fis, anfis2.fis, anfis3.fis) are generated in Matlab, after ANFIS training, which are our desired controller. Since *-g(M+m)* term exists in $\boldsymbol{u_3^m}$, therefore, gravity compensation of *g(M+m)* is added to the output of 3rd controller (that controls actuator along *z* axis). These controllers are further implemented on nonlinear model of *x-y-z* inverted pendulum and the results obtained are compared with LQR controller with nominal gain ($\boldsymbol{K} = \boldsymbol{K}^{0.1}$) as shown in Figures 5.27-5.31. The Matlab Simulink block diagram used to compare the controllers is shown in Figure 5.26.

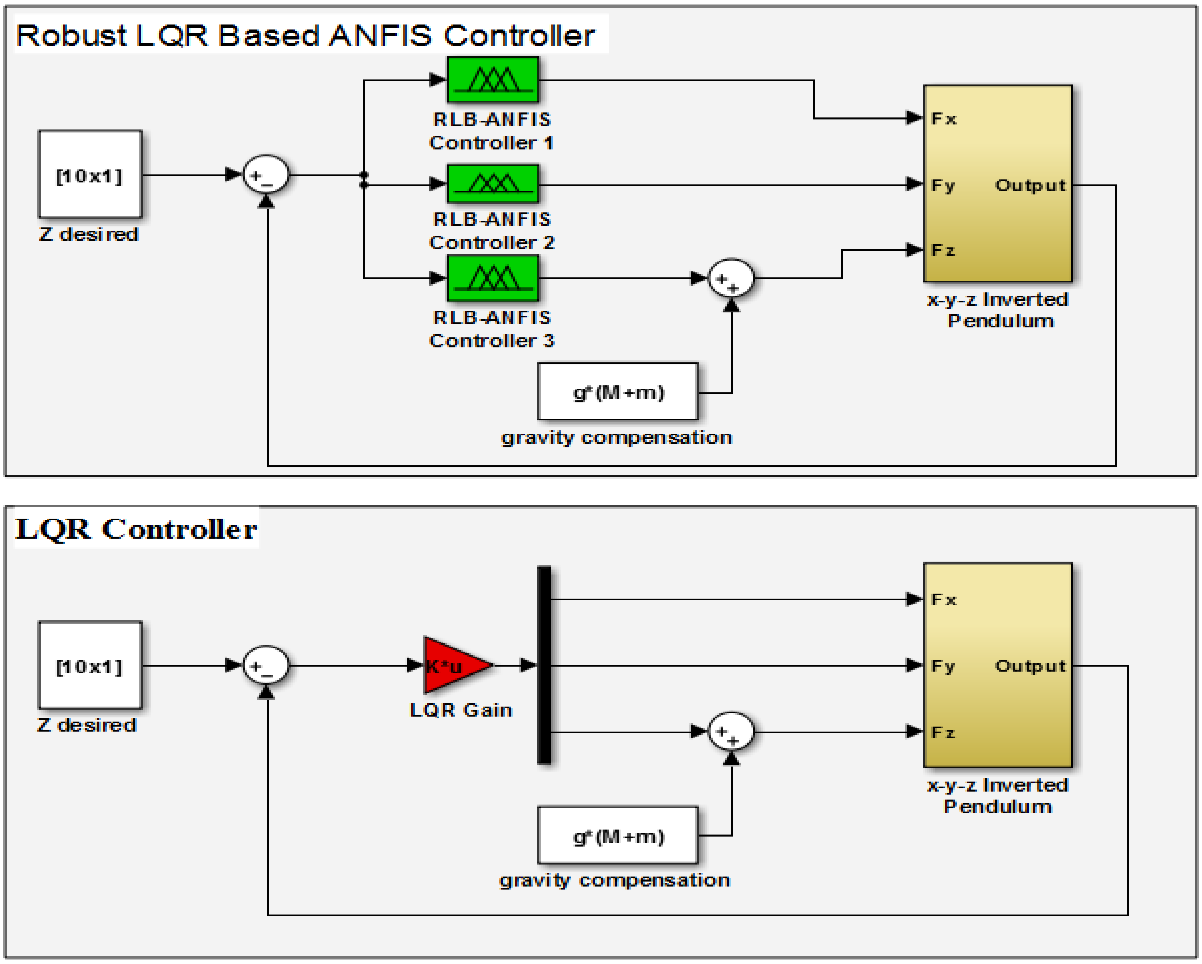

Figure 5.26: Simulink block diagram for comparison between RLQR-ANFIS controller and LQR controller

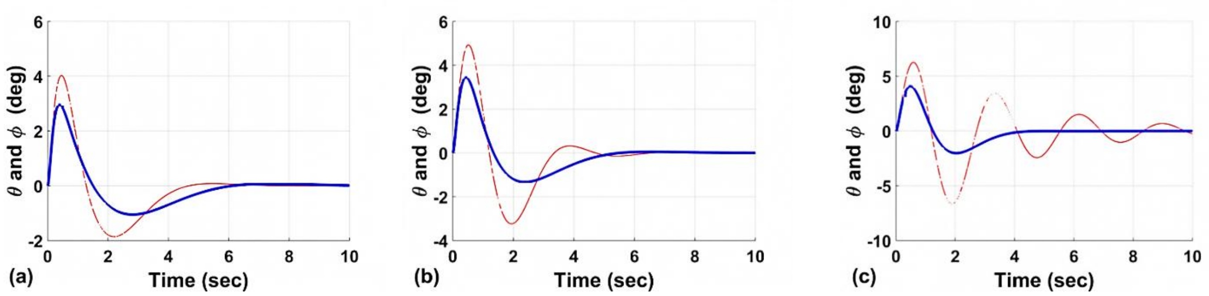

Figure 5.27: Response of θ and ϕ for pendulum mass (m) in kg = a) 0.1; b) 0.8; c) 1.5 (solid line: robust LQR based ANFIS controller and dashed line: LQR controller)

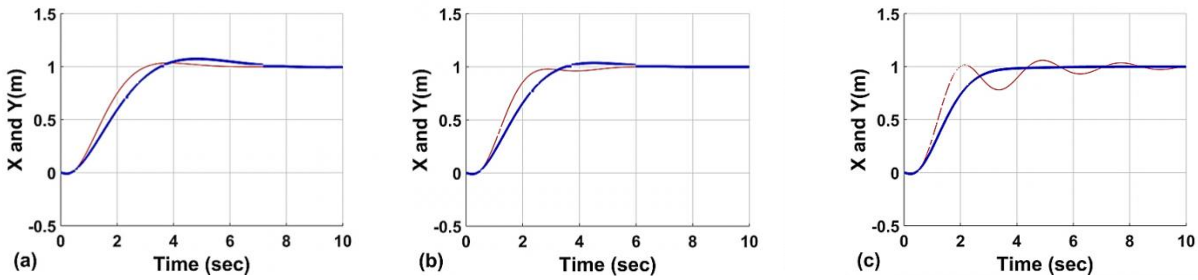

Figure 5.28: Response of x and y position for pendulum mass (m) in kg = a) 0.1; b) 0.8; c) 1.5 (solid line: robust LQR based ANFIS controller and dashed line: LQR controller)

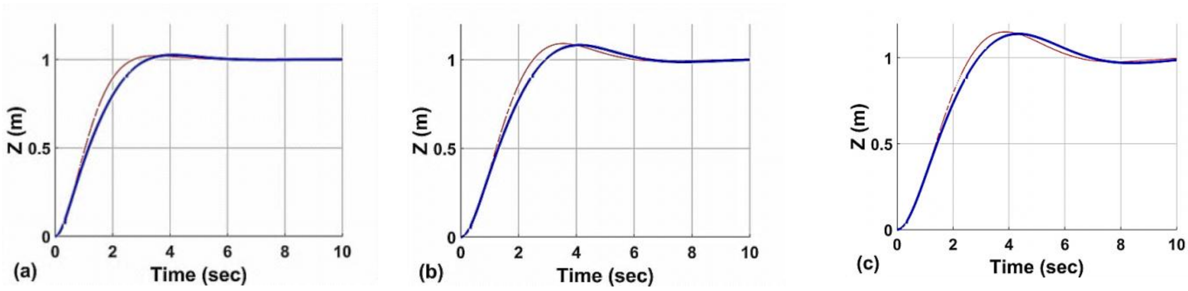

Figure 5.29: Response of z position for pendulum mass (m) in kg = a) 0.1; b) 0.8; c) 1.5 (solid line: robust LQR based ANFIS controller and dashed line: LQR controller)

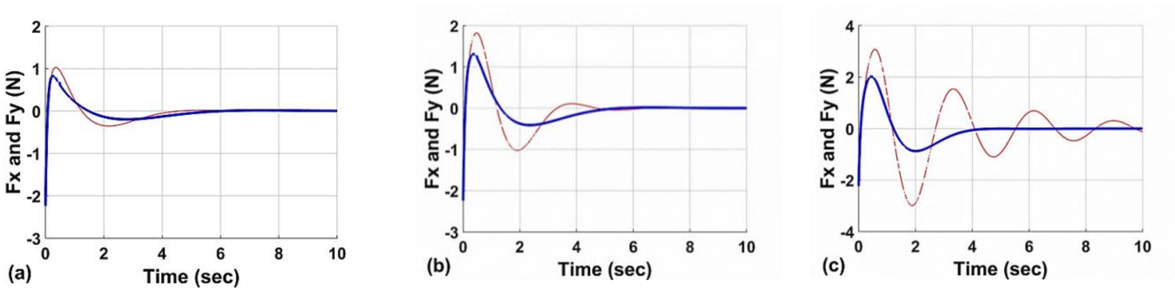

Figure 5.30: Response of input F_x and F_y for pendulum mass (m) in kg = a) 0.1; b) 0.8; c) 1.5 (solid line: robust LQR based ANFIS controller and dashed line: LQR controller)

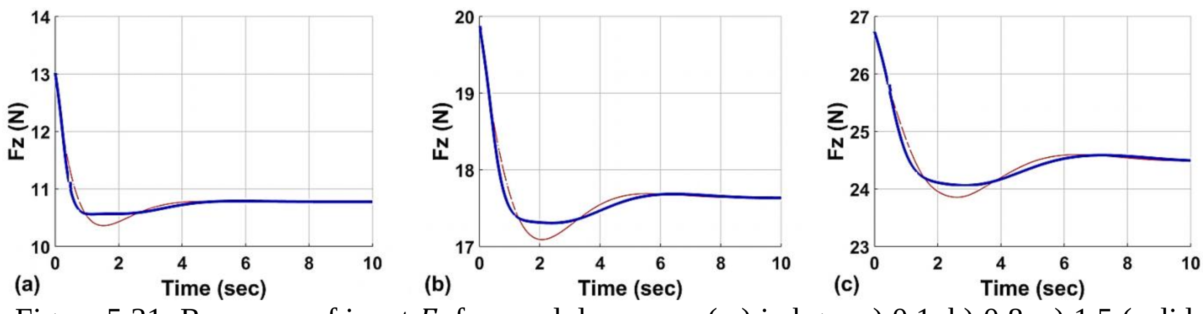

Figure 5.31: Response of input F_z for pendulum mass (m) in kg = a) 0.1; b) 0.8; c) 1.5 (solid line: robust LQR based ANFIS controller and dashed line: LQR controller)

From the simulation results attained in Figures 5.26-5.31, it can be seen that for pendulum mass of 0.1 kg, with LQR controller, the performance parameters such as pendulum settling time (T_s) and maximum overshoot in pendulum angles (θ_{max} and ϕ_{max}), z position (z_{max}), x and y positions (x_{max} and y_{max}), input forces in x- and y-direction ($F_{x_{max}}$ and $F_{y_{max}}$) and input force in z-direction ($F_{z_{max}}$) are 4.484 sec, 4.0272 deg, 1.0201 m, 1.034 m, 1.0270 N, 13.0161 N respectively, while with RLQR-ANFIS controller performance parameters are 6.2778 sec, 2.9581 deg, 1.0251 m, 1.0733 m, 0.8283 N, 13.0161N respectively. Thus, a reduction of 26.55% in maximum overshoot in pendulum angles (θ_{max} and ϕ_{max}) and a reduction of 19.35% in maximum overshoot in input forces in x- and y-direction ($F_{x_{max}}$ and $F_{y_{max}}$) is obtained while an increment of 40%, 3.72% and 0.40% is obtained in performance parameters such as pendulum settling time (T_s), maximum overshoot in z position (z_{max}) and x position (x_{max}) respectively. It is perceived that even though for nominal pendulum mass, few performance parameters have degraded. However, as the pendulum mass is increased (varied further), the proposed controller gives better performance than LQR controller as reported in Table 5.9, which shows that the proposed controller is robust to pendulum mass while LQR controller performance degrades as the mass is increased. Moreover, it is observed that on further increasing the pendulum mass to 1.9 kg, the LQR controller fails to stabilize the inverted pendulum, while the RLQR-ANFIS controller gives consistent performance as shown in Figure 5.32.

Table 5.9: Performance parameters comparison for different pendulum masses on *x-y-z* inverted pendulum

Pendulum Mass (*m*)	Performance Parameters	Units	Robust LQR based ANFIS	LQR	Percentage Reduction
0.1	θ_{max} and ϕ_{max}	degree	2.9581	4.0272	26.55 %
	x_{max} and y_{max}	meter	1.0733	1.0348	-3.72 %
	z_{max}	meter	1.0251	1.0210	-0.40 %
	$F_{x_{max}}$ and $F_{y_{max}}$	Newton	0.8283	1.0270	19.35 %
	$F_{z_{max}}$	Newton	13.0161	13.0161	0.00 %
	T_s	second	6.2778	4.484	-40.00 %
0.8	θ_{max} and ϕ_{max}	degree	3.4404	4.9294	30.21 %
	x_{max} and y_{max}	meter	1.0360	1.0000	-3.60 %
	z_{max}	meter	1.0823	1.0911	0.81 %
	$F_{x_{max}}$ and $F_{y_{max}}$	Newton	1.3113	1.8235	28.09 %
	$F_{z_{max}}$	Newton	19.8761	19.8761	0.00 %
	T_s	second	5.1489	5.9156	12.96 %
1.5	θ_{max} and ϕ_{max}	degree	4.0746	6.2621	34.93 %
	x_{max} and y_{max}	meter	0.9999	1.0601	5.68 %
	z_{max}	meter	1.1401	1.1513	0.97 %
	$F_{x_{max}}$ and $F_{y_{max}}$	Newton	2.0208	3.0796	34.38 %
	$F_{z_{max}}$	Newton	26.7361	26.7361	0.00 %
	T_s	second	4.2120	13.6483	69.14 %

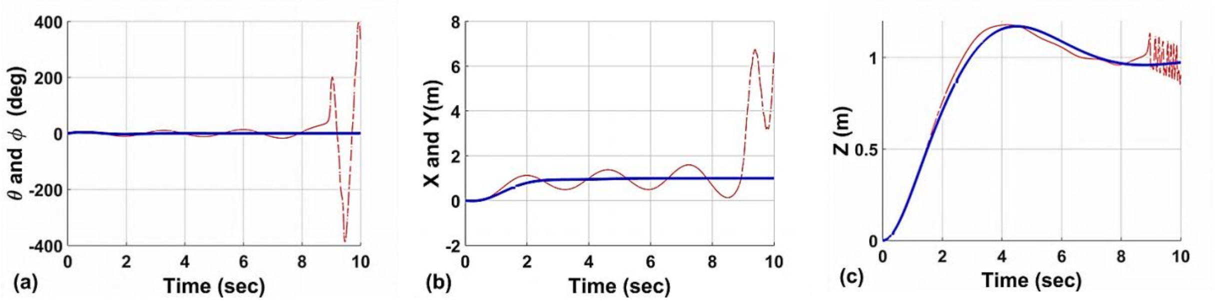

Figure 5.32: Response of a) θ and ϕ; b) x and y position; c) z position for pendulum mass m = 1.9 kg (solid line: robust LQR based ANFIS controller and dashed line: LQR controller)

5.4 Summary

In this chapter, a novel robust LQR based ANFIS controller has been proposed for stabilization problem of different inverted pendulum systems. Step by step design procedure of the RLQR-ANFIS controller has been discussed for different inverted pendulum systems. The designed controller is implemented on different inverted pendulum systems and simulation as well as experimental results has been obtained. Further, the proposed controller has been compared with conventional stabilization controllers such as LQR or PID controller for different pendulum masses and the results have shown that the proposed controller has better robustness along with satisfactory performance as compared to the conventional stabilization controllers. Further, it has been observed that on increasing pendulum mass, the conventional controller failed to stabilize inverted pendulum while the robust LQR based ANFIS controller has insignificant impact on the performance of system. Thus, the proposed controller has solved the problem of robustness in LQR controller along with having an advantage of optimal property of LQR controller. Further, swing-up controller for rotary inverted pendulum is discussed in next chapter.

Chapter 6

Swing-up Control

6.1 Introduction

In this chapter, swing-up controller design is discussed for rotary inverted pendulum system. Generally, two specific tasks of rotary inverted pendulum system are controlled. The first task is to swing-up the pendulum from its stable equilibrium point to unstable equilibrium point by a specific controller i.e., from downward position to upward position. The second task is to stabilize the pendulum around the upright by designing a state feedback controller. The swing-up controller is a nonlinear problem while stabilization is a linear problem.

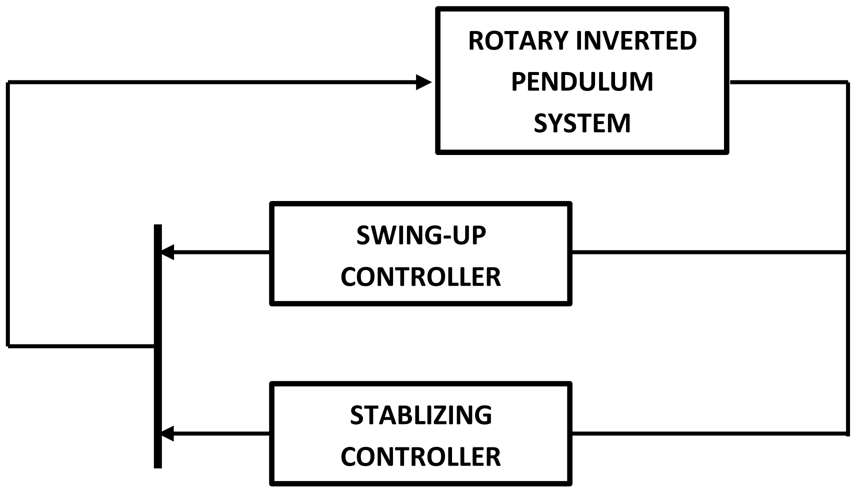

Figure 6.1: Control scheme of rotary inverted pendulum system

The control scheme for the swing-up and stabilizing of the rotary inverted pendulum system is shown in Figure 6.1. Energy controller is used for the swing-up of the pendulum. LQR or pole-placement controllers can be used as stabilizing controller as discussed in Chapter 4. Apart from swing up and stabilization, tracking controller can also be designed for this system which tracks a rotary arm trajectory while balancing the pendulum in upright position

6.2 Swing-up Controller

This controller aims at swinging up the pendulum from downward stable equilibrium position to upward unstable equilibrium position. Energy controller is used for this task. In energy controller, maximum acceleration is provided to the rotary arm to swing the pendulum. The dynamics of the pendulum in terms of pivot acceleration u can be written as:

$$J_p\ddot{\alpha} + \frac{1}{2}M_p g L_p \sin\alpha = \frac{1}{2}M_p u L_p \cos\alpha \tag{6.1}$$

The free body diagram of the pendulum is illustrated in Figure 6.2

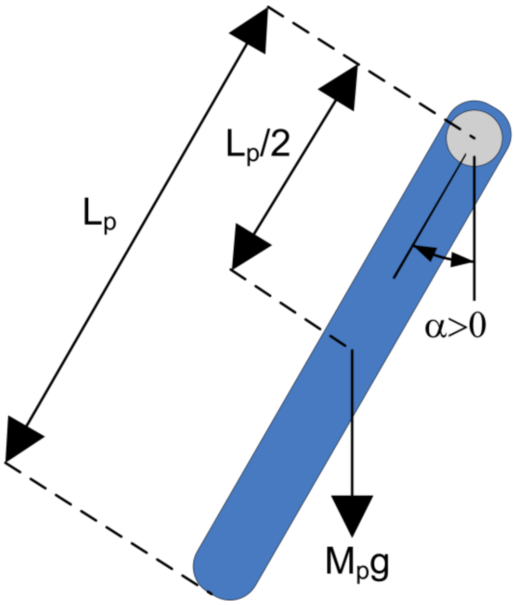

Figure 6.2: Free body diagram of pendulum

The potential energy of the pendulum (taking centre of mass of pendulum in vertical downwards position as reference) can be described as

$$P.E. = \frac{1}{2}M_p g L_p(1 - \cos\alpha) \tag{6.2}$$

So the potential energy is zero when the pendulum is at vertically downwards position i.e. $\alpha = 0$ and equals to $M_p g L_p$ when it is at vertically upright position i.e. $\alpha = \pm\pi$.

Also kinetic energy can be described as

$$K.E. = \frac{1}{2}J_p\dot{\alpha}^2 \tag{6.3}$$

The energy of a single unconstrained IP system is:

$$E = \frac{1}{2}J_p\dot{\alpha}^2 + \frac{1}{2}M_p g L_p(1 - \cos\alpha) \tag{6.4}$$

Computing the derivative of E with respect to time we find that

$$\frac{dE}{dt} = \dot{\alpha}\left(J_p\ddot{\alpha} + \frac{1}{2}M_p g L_p \sin\alpha\right) = \frac{1}{2}M_p u\, L_p\, \dot{\alpha}\cos\alpha \tag{6.5}$$

Here, u is the input to the inverted pendulum which is the acceleration of the rotary arm. Using Lyapunov theory as suggested by Astrom in [47], control input u is chosen as

$$u = (E_r - E)\,\dot{\alpha}\cos\alpha \tag{6.6}$$

Where, *Er* is reference energy of pendulum which is set to P.E. of pendulum at vertically upright position. Here controllability of the system is lost when the coefficient of u in the right side of (6.6) vanishes. It occurs when $\dot{\alpha} = 0$ or $\alpha = \pm\frac{\pi}{2}$, i.e. when the pendulum is horizontal or when it reverses its velocity. At angle 0 or π, controller action is most effective and the velocity is large.

To prevent chattering and to change the energy of pendulum as fast as possible the magnitude of control input is modified as

$$u = sat_{u_{max}}(\mu(E_r - E)sign(\dot{\alpha}\cos\alpha)) \tag{6.7}$$

Where μ is a tunable controller gain and the $sat_{u_{max}}$ function saturates the control signal at the maximum acceleration of the pendulum pivot, u_{max}. The expression $sign(\dot{\alpha}\cos\alpha)$ is used to obtain fast control switching.

Where $sign(u) = \begin{cases} 1 & if\ u > 0 \\ 0 & if\ u = 0 \\ -1 & if\ u < 0 \end{cases}$

6.3 Analytical and Experimental Results

6.3.1 Swing up Control

The controller for swinging up the pendulum is designed using Matlab Simulink where swing-up control parameters are set to the following:

μ = 50 m/s^2/J

E_r = 20.0 mJ

u_{max} = 6 m/s^2

The Matlab Simulink block diagram used for implementing controller is shown in Figure 6.3. Using the above control parameters following response is obtained as shown in Figure 6.4. It is noted that pendulum energy is increasing with time and pendulum is slowly swinging to vertically upright position.

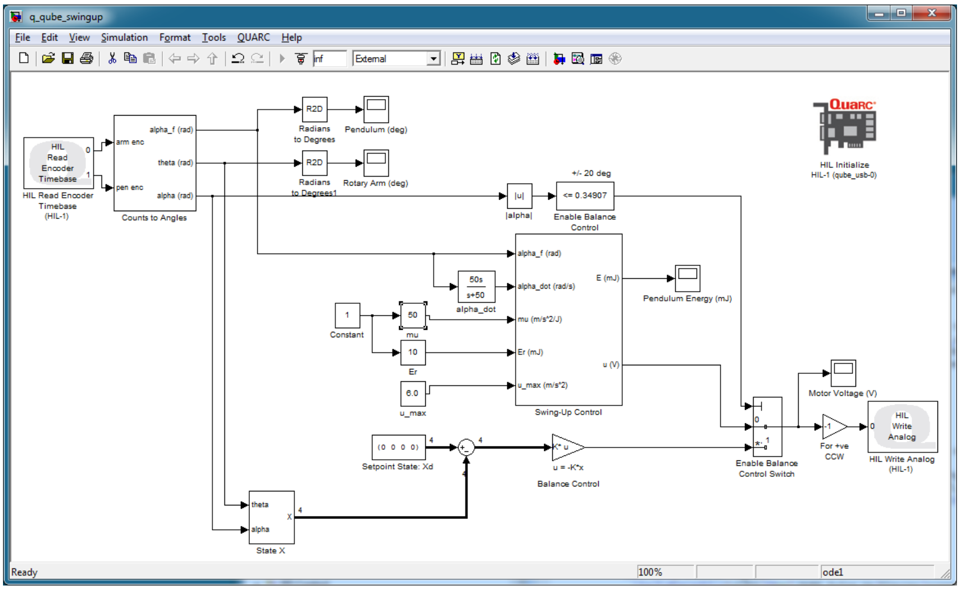

Figure 6.3: Simulink block diagram for swing-up controller

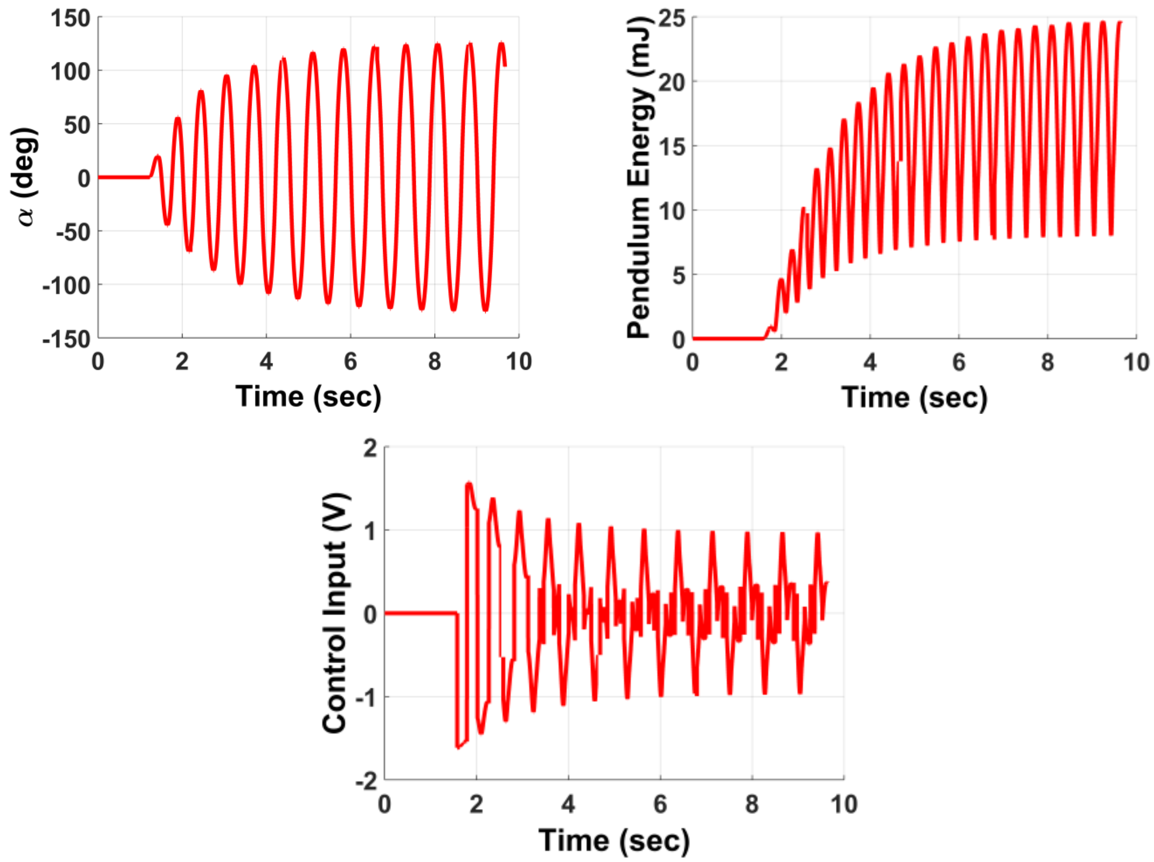

Figure 6.4: Experimental response of swing-up controller on rotary inverted pendulum

Now if E_r is increased from 20 to 30 mJ, it is observed that the amplitude of control signal increases. While if the value of μ is increased from 50 to 60, it is observed that as μ increases, the amplitude of pendulum swings become larger.

6.3.2 Hybrid Swing up Control

In this controller, a swing up controller is combined with a stabilization control i.e. LQR control. This is done by switching from swing up control to stabilization control when the pendulum angle α is within ± 20 degree from vertically upright position. Thus the switching can be described mathematically by:

$$u = \begin{cases} u_{stab}, & if\ |\alpha| - \pi \leq 20\ deg \\ u_{swing}, & otherwise \end{cases}$$

where u_{stab} is the controller input for stabilization control and u_{swing} is the controller input for swing up control. Using the Matlab Simulink block diagram as illustrated in Figure 6.3 with swing up control parameters as:

$$\mu = 38\ \text{m/s}^2/\text{J}$$

$$E_r = 30.0\ \text{mJ}$$

$$u_{max} = 6\ \text{m/s}^2$$

$$\boldsymbol{K} = [-2.24, 53.86, -1.70, 6.53]$$

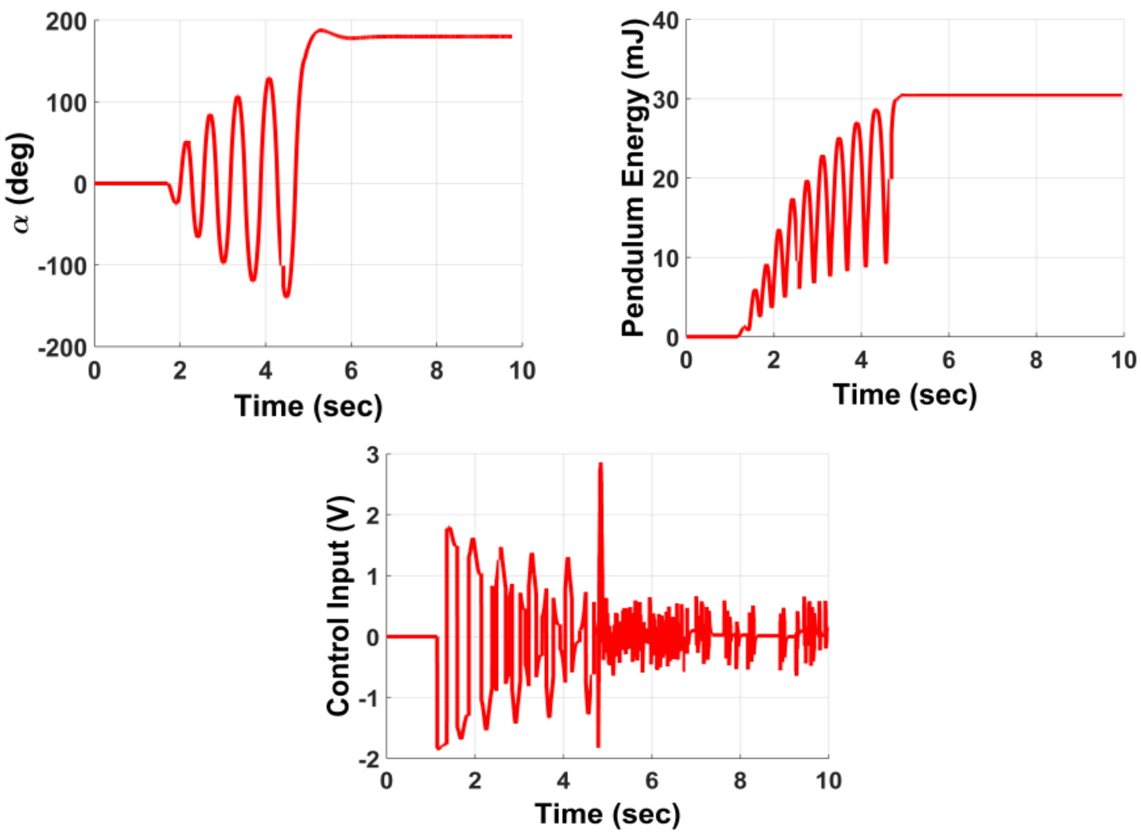

Figure 6.5:Experimental response of hybrid swing-up controller on rotary inverted pendulum

From the response as given in Figure 6.5, observed that

- Pendulum is stabilized from vertically downward position to vertically upward position in 7 seconds and 4 swings. This time can be further reduced by increasing factors like E_r (reference energy) and μ.
- Energy of the pendulum is increasing with time.
- LQR controller stabilizes the pendulum with good performance when the pendulum angle α reaches within ± 20 degree from vertically upright position.
- Sudden increase in control input while switching from swing up controller to stabilization controller.

6.4 Summary

This chapter presented swing-up controller for rotary inverted pendulum. Energy based controller has been used for swinging up the pendulum. The controller has been combined with stabilization controller i.e. LQR controller to create a hybrid control algorithm which has been implemented on rotary inverted pendulum.

Chapter 7

Closing Remarks

7.1 Conclusions

The major contribution of this dissertation is the design of a novel robust LQR based ANFIS controller. The step by step procedure to design the proposed controller has been explained in this dissertation. Further the proposed controller is implemented on different inverted pendulum systems i.e. *x-y-z*, *x-y*, *x-z*, *x* and rotary inverted pendulum to verify its effectiveness. The robustness of the controlled is verified by varying the pendulum mass and then analysing its performance parameters. Moreover, the results obtained are compared with LQR and PID controller where the percentage reduction in performance parameters has been analysed for different cases. Finally, after rigorous analysis, it can be concluded that the proposed controller have outperformed LQR and PID controller in the domain of robustness. Other contributions of this dissection are listed below:

- Accurate modeling of different inverted pendulum systems i.e. *x-y-z*, *x-y*, *x-z*, *x* and rotary had been done using Euler-Lagrange approach. The obtained models were further linearized at unstable equilibrium point i.e. vertically upright pendulum position and a state-space representation is obtained. Actuator dynamics were taken into account while modeling rotary inverted pendulum. Further, the mathematical model of rotary inverted pendulum had been compared with real-time rotary inverted pendulum and it is observed that model approximately resembles real-time rotary inverted pendulum.
- Fine tuning of pole placement controller and LQR controller is done and their results are compared on the basis of tracking performance and it has been observed that LQR gives a better tracking performance than pole placement.
- It has been concluded that tuning of LQR is relatively easier than pole placement approach as performance of state variables in LQR is directly dependent on state weighing matrix $\boldsymbol{Q}$, while for pole placement controller, desired poles are difficult to decide and are not directly dependent on performance of state variables, thus making the tuning of pole placement controller a relatively complex task.

- A swing-up controller based on energy method is designed and tuned so as to obtain the swing up of pendulum. Effect of different control parameters is observed on the swing up action of pendulum.
- Hybrid swing-up controller is designed which switches form swing-up controller to stabilization controller as the pendulum angle reaches in the described range. The stabilization controller used here is LQR.

7.2 Future Work

On the basis of literature survey and the work done it this dissertation, work on following areas can be done in future:

- Experimental validation of dynamic mathematical model of *x*, *x*-*y*, *x*-*z* and *x*-*y*-*z* inverted pendulum can be done by comparing simulation response with real-time setup response.
- Improvement in dynamic mathematical model of rotary inverted pendulum can be achieved by considering actuator wire dynamics, actuator delay, static friction, stick slip friction, working table friction, etc. so that, the analytical and experimental results can be resembled more closely.
- Use of system identification techniques can also be done to obtain the accurate model of the inverted pendulum system. ANFIS can also be used as a system identification tool to do so.
- Optimization of tuning parameters of LQR controller i.e. Q and R matrix can be done using optimization algorithm such as particle swarm optimization (PSO) or genetic algorithm (GA) using performance index as the cost function to be minimized.
- On the similar basis, appropriate optimization algorithms can also be used to obtain accurate gains for swing-up controller.
- The sensory noise can be addressed using Kalman filters.
- Designing a robust LQR based ANFIS controller which is robust to other parametric variations such as mass of cart, length of pendulum, actuator parameters etc. can also be done.
- Work on smooth switching between swing-up and stabilization controller can also be done.

References

[1] Davison, E.J., *Benchmark problems for control system design.* IFAC Theory Comitee, 1990.

[2] Mason, P., M. Broucke, and B. Piccoli, *Time optimal swing-up of the planar pendulum.* IEEE Transactions on Automatic Control, 2008. **53**(8): p. 1876-1886.

[3] Muskinja, N. and B. Tovornik, *Swinging up and stabilization of a real inverted pendulum.* IEEE Transactions on Industrial Electronics, 2006. **53**(2): p. 631-639.

[4] Solihin, M.I. and R. Akmeliawati, *Particle Swam Optimization for Stabilizing Controller of a Self-erecting Linear Inverted Pendulum.* International Journal of Electrical and Electronic Systems Research, 2010. **3**: p. 410-415.

[5] Chanchareon, R., V. Sangveraphunsiri, and S. Chantranuwathana. *Tracking control of an inverted pendulum using computed feedback linearization technique*. in *Robotics, Automation and Mechatronics, 2006 IEEE Conference on*. 2006. IEEE.

[6] Liu, Y., S. Wane, and T. Yang, *On tracking control of a pendulum-driven cart-pole underactuated system.* International Journal of Modelling, Identification and Control, 2008. **4**(4): p. 357-372.

[7] Younis, W. and M. Abdelati, *Design and implementation of an experimental segway model*. in *AIP Conference Proceedings*. 2009. AIP.

[8] Tsai, C.-C., H.-C. Huang, and S.-C. Lin, *Adaptive neural network control of a self-balancing two-wheeled scooter.* IEEE Transactions on Industrial Electronics, 2010. **57**(4): p. 1420-1428.

[9] Anderson, C.W., *Learning to control an inverted pendulum using neural networks.* IEEE Control Systems Magazine, 1989. **9**(3): p. 31-37.

[10] Kennedy, E.A., *Swing-up and stabilization of a single inverted pendulum: real-time implementation*. 2015: North Carolina State University.

[11] Housner, G.W., *The behavior of inverted pendulum structures during earthquakes.* Bulletin of the seismological society of America, 1963. **53**(2): p. 403-417.

[12] Loram, I.D. and M. Lakie, *Human balancing of an inverted pendulum: position control by small, ballistic-like, throw and catch movements.* The Journal of physiology, 2002. **540**(3): p. 1111-1124.

[13] Elhasairi, A. and A. Pechev, *Humanoid Robot Balance Control Using the Spherical Inverted Pendulum Mode.* Frontiers in Robotics and AI, 2015. **2**: p. 21.

[14] Martin, P., S. Devasia, and B. Paden, *A different look at output tracking: control of a VTOL aircraft.* Automatica, 1996. **32**(1): p. 101-107.

[15] Furuta, K., M. Yamakita, and S. Kobayashi, *Swing-up control of inverted pendulum using pseudo-state feedback.* Proceedings of the Institution of Mechanical Engineers, Part I: Journal of Systems and Control Engineering, 1992. **206**(4): p. 263-269.

[16] Yang, R., Y.-Y. Kuen, and Z. Li. *Stabilization of a 2-DOF spherical pendulum on xy table*. in *Control Applications, 2000. Proceedings of the 2000 IEEE International Conference on*. 2000. IEEE.

[17] Jia-Jun, W. *Stabilization and tracking control of xz inverted pendulum based on pid controllers*. in *Control Conference (CCC), 2015 34th Chinese*. 2015. IEEE.

[18] Spong, M.W., *The swing up control problem for the acrobot.* IEEE control systems, 1995. **15**(1): p. 49-55.

[19] Fantoni, I., R. Lozano, and M.W. Spong, *Energy based control of the pendubot.* IEEE Transactions on Automatic Control, 2000. **45**(4): p. 725-729.

[20] Block, D.J., K.J. Åström, and M.W. Spong, *The reaction wheel pendulum.* Synthesis Lectures on Control and Mechatronics, 2007. **1**(1): p. 1-105.

[21] Acosta, J., *Furuta's pendulum: A conservative nonlinear model for theory and practise.* Mathematical Problems in Engineering, 2010. **2010**.

[22] Wang, J.-J., *Stabilization and tracking control of X–Z inverted pendulum with sliding-mode control.* ISA transactions, 2012. **51**(6): p. 763-770.

[23] Kumar, E.V. and J. Jerome, *Robust LQR controller design for stabilizing and trajectory tracking of inverted pendulum.* Procedia Engineering, 2013. **64**: p. 169-178.

[24] Kathpal, A. and A. Singla, *SimMechanics Based Modeling, Simulation and Real-Time Control of Rotary Inverted Pendulum*, in *11th IEEE International Conference on Intelligent Systems and Control (ISCO-2017)*2016: Coimbatore, Tamilnadu. p. 166-172.

[25] Wang, J.-J., *Simulation studies of inverted pendulum based on PID controllers.* Simulation Modelling Practice and Theory, 2011. **19**(1): p. 440-449.

[26] Eide, R., P.M. Egelid, and H.R. Karimi, *LQG control design for balancing an inverted pendulum mobile robot.* Intelligent Control and Automation, 2011. **2**(02): p. 160.

[27] Singh, N., J. Dubey, and G. Laddha, *Control of pendulum on a cart with state dependent riccati equations.* Analysis, 1870. **2**(4): p. 2008.

[28] Yiğit, İ., *Model free sliding mode stabilizing control of a real rotary inverted pendulum.* Journal of Vibration and Control, 2015: p. 1077546315598031.

[29] Bui, H.-L., D.-T. Tran, and N.-L. Vu, *Optimal fuzzy control of an inverted pendulum.* Journal of vibration and control, 2012. **18**(14): p. 2097-2110.

[30] Mohan, V. and N. Singh. *Performance comparison of LQR and ANFIS controller for stabilizing double inverted pendulum system*. in *Signal Processing, Computing and Control (ISPCC), 2013 IEEE International Conference on*. 2013. IEEE.

[31] Kailath, T., *Linear systems*. Vol. 156. 1980: Prentice-Hall Englewood Cliffs, NJ.

[32] Boubaker, O., *The inverted pendulum benchmark in nonlinear control theory: a survey.* International Journal of Advanced Robotic Systems, 2013. **10**(5): p. 233.

[33] Mori, S., H. Nishihara, and K. Furuta, *Control of unstable mechanical system control of pendulum.* International Journal of Control, 1976. **23**(5): p. 673-692.

[34] Furuta, K., M. Yamakita, and S. Kobayashi. *Swing up control of inverted pendulum*. in *Industrial Electronics, Control and Instrumentation, 1991. Proceedings. IECON'91., 1991 International Conference on*. 1991. IEEE.

[35] Zadeh, L.A., *Fuzzy sets.* Information and control, 1965. **8**(3): p. 338-353.

[36] Yi, J., N. Yubazaki, and K. Hirota, *Upswing and stabilization control of inverted pendulum system based on the SIRMs dynamically connected fuzzy inference model.* Fuzzy Sets and Systems, 2001. **122**(1): p. 139-152.

[37] Hasan, M., *Balancing of an inverted pendulum using PD controller.* Dhaka University Journal of Science, 2012. **60**(1): p. 115-120.

[38] Buonocunto, P. and F. Corucci, *Real-time PID control of an inverted pendulum.* MSc in Comp. Engineering, Univ. of Pisa, Italy, 2012.

[39] Ghosh, A., T. Krishnan, and B. Subudhi, *Robust proportional–integral–derivative compensation of an inverted cart–pendulum system: an experimental study.* IET control theory & applications, 2012. **6**(8): p. 1145-1152.

[40] Chang, W.-D., R.-C. Hwang, and J.-G. Hsieh, *A self-tuning PID control for a class of nonlinear systems based on the Lyapunov approach.* Journal of Process Control, 2002. **12**(2): p. 233-242.

[41] Rajak, N., *Stabilization of cart-inverted pendulum using pole-placement method.* Int. J. Adv. Engg. Res. Studies/IV/II/Jan.-March, 2015. **308**: p. 312.

[42] Roshdy, A.A. and T. Wang, *Stabilization of real inverted pendulum using pole separation factor*. in *International Conference on Mechanical Engineering and Material Science*. 2012.

[43] Akhtaruzzaman, M. and A.A. Shafie. *Modeling and control of a rotary inverted pendulum using various methods, comparative assessment and result analysis*. in *Mechatronics and Automation (ICMA), 2010 International Conference on*. 2010. IEEE.

[44] Trimpe, S. and A. Millane, *A self-tuning LQR approach demonstrated on an inverted pendulum.* IFAC Proceedings Volumes, 2014. **47**(3): p. 11281-11287.

[45] Huifeng, C., L. Hongxing, and Y. Peipei. *Swinging-up and stabilization of the inverted pendulum by energy well and SDRE control*. in *Control and Decision Conference, 2009. CCDC'09. Chinese*. 2009. IEEE.

[46] Dang, P. and F.L. Lewis. *Controller for swing-up and balance of single inverted pendulum using SDRE-based solution*. in *Industrial Electronics Society, 2005. IECON 2005. 31st Annual Conference of IEEE*. 2005. IEEE.

[47] Åström, K.J. and K. Furuta, *Swinging up a pendulum by energy control.* Automatica, 2000. **36**(2): p. 287-295.

[48] Lozano, R., I. Fantoni, and D.J. Block, *Stabilization of the inverted pendulum around its homoclinic orbit.* Systems & control letters, 2000. **40**(3): p. 197-204.

[49] Chatterjee, D., A. Patra, and H.K. Joglekar, *Swing-up and stabilization of a cart–pendulum system under restricted cart track length.* Systems & control letters, 2002. **47**(4): p. 355-364.

[50] Yang, J.-H and J.-H Seo, *Swing-up control for an inverted pendulum with restricted cart rail length.* International Journal of Control, Automation and Systems, 2009. **7**(4): p. 674-680.

[51] Zhao, J. and M.W. Spong, *Hybrid control for global stabilization of the cart–pendulum system.* Automatica, 2001. **37**(12): p. 1941-1951.

[52] Chung, C.C. and J. Hauser, *Nonlinear control of a swinging pendulum.* Automatica, 1995. **31**(6): p. 851-862.

[53] Santiesteban, R and T. Floquet, *Second-order sliding mode control of underactuated mechanical systems II: Orbital stabilization of an inverted pendulum with application to swing up/balancing control.* International Journal of Robust and Nonlinear Control, 2008. **18**(4-5): p. 544-556.

[54] Park, M.-S. and D. Chwa, *Swing-up and stabilization control of inverted-pendulum systems via coupled sliding-mode control method.* IEEE Transactions on Industrial Electronics, 2009. **56**(9): p. 3541-3555.

[55] Bhavsar, P. and V. Kumar, *Trajectory tracking of linear inverted pendulum using integral sliding mode control.* International Journal of Intelligent Systems and Applications, 2012. **4**(6): p. 31.

[56] Ogata, K., *System dynamics*. Vol. 3. 1998: Prentice Hall New Jersey.

[57] Shabana, A.A., *Dynamics of multibody systems*. 2013: Cambridge university press.

[58] Tewari, A., *Modern control design.* NY: John Wiley & sons, 2002.

[59] Jang, J.-S., *ANFIS: adaptive-network-based fuzzy inference system.* IEEE transactions on systems, man, and cybernetics, 1993. **23**(3): p. 665-685.

[60] Takagi, T. and M. Sugeno. *Derivation of fuzzy control rules from human operator's control actions*. in *Proceedings of the IFAC symposium on fuzzy information, knowledge representation and decision analysis*. 1983. sn.

Web References

[W1] http://people.kth.se/~crro/segway_challenge/model.html
(last accessed on 1 July, 2017)

[W2] https://en.wikipedia.org/wiki/Inverted_pendulum
(last accessed on 1 July, 2017)

[W3] http://www.segway.com/
(last accessed on 1 July, 2017)

[W4] http://www.fastcodesign.com/3054003/dont-gift-a-hoverboard-thischristmas
(last accessed on 1 July, 2017)

[W5] http://carfixers.in/electric-unicycle-one-upcoming-tech-toys-will-makelife-easier/
(last accessed on 1 July, 2017)

[W6] http://rspb.royalsocietypublishing.org/content/280/1762/20130700
(last accessed on 1 July, 2017)

[W7] https://en.wikipedia.org/wiki/PID_controller
(last accessed on 1 July, 2017)

Ishan Dissertation

ORIGINALITY REPORT

%14	%8	%10	%5
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	mariovalle.name Internet Source	%1
2	Sukontanak, Viroch, and Manukid Parnichkun. "Real-Time Optimal Control for Rotary Inverted Pendulum", American Journal of Applied Sciences, 2009. Publication	<%1
3	ethesis.nitrkl.ac.in Internet Source	<%1
4	www.easysafe.com.cn Internet Source	<%1
5	Submitted to University of Southampton Student Paper	<%1
6	en.wikipedia.org Internet Source	<%1
7	Submitted to University of Florida Student Paper	<%1
8	於2012-06-24提交至Universiti Teknologi Malaysia	<%1

Student Paper

9	www.coursehero.com Internet Source	<%1
10	Abhishek Kathpal, Ashish Singla. "SimMechanics™ based modeling, simulation and real-time control of Rotary Inverted Pendulum", 2017 11th International Conference on Intelligent Systems and Control (ISCO), 2017 Publication	<%1
11	www.netbsd.org Internet Source	<%1
12	faculty.nipissingu.ca Internet Source	<%1
13	Vinodh Kumar, E. and Jerome, Jovitha. "Stabilizing Controller Design for Self Erecting Single Inverted Pendulum using Robust LQR", Australian Journal of Basic & Applied Sciences, 2013. Publication	<%1
14	Ozbek, Necdet Sinan, and Mehmet Onder Efe. "Swing up and stabilization control experiments for a rotary inverted pendulum— An educational comparison", 2010 IEEE International Conference on Systems Man and Cybernetics, 2010. Publication	<%1