CONTROL OF ROTARY DOUBLE INVERTED PENDULUM USING NEURAL NETWORK BASED ADAPTIVE LINEAR QUADRATIC REGULATOR

by

Viroch Sukontanakarn

A dissertation submitted in partial fulfillment of the requirements for the degree of Doctor of Engineering in Mechatronics

Examination Committee: Dr. Manukid Parnichkun (Chairperson)
Prof. Vilas Wuwongse
Dr. Nitin V. Afzulpurkar

External Examiner: Dr. Christine Chevallereau
Institute of Research in Communication and Cybernetics, Nantes, France

Nationality: Thai
Previous Degree: Master of Engineering (Electric Power System Management)
Asian Institute of Technology, Thailand

Scholarship Donor: Rajamangala University of Technology Isan
KhonKaen Campus, Thailand

Asian Institute of Technology
School of Engineering and Technology
Thailand
August 2011
Acknowledgements

I would like to express my deepest and sincere gratitude to Associate Professors Dr. Manukid Parnichkun, my advisor, for kindly arrangement a good topic research, helpful guidance, valuable suggestion, thoughtful discussions and improving my mathematical. His supervisions have greatly helped me to complete the dissertation in due time.

I would like to express my sincere appreciation to committees, Prof. Vilas Wuwongse and Associate Professors Dr. Nitin V. Afzulpurkar for their advice and suggestions. I also thank Dr. Christine Chevallereau for being my external examiner and his valuable questions.

I would like to acknowledge the kindness of Rajamangala University of Technology Isan Khonkaen Campus for the full scholarship. This research can never be completed without this financial support.
Abstract

This dissertation introduces the development and control of a rotary double inverted pendulum (RDIP) system. Rotary double inverted pendulum system is a nonlinear and unstable system. The control of a nonlinear system is a challenging topic for control engineers. This dissertation presents an optimization method for balancing control of a rotary double inverted pendulum system. The design of the controller of this system is difficult and challenging because it has under-actuated control input. The input, torque of a motor, is used to control three links of the system; the arm position is maintained in horizontal plane, and the two pendulums are maintained at the upright position. The main controller used in this dissertation is the linear quadratic regulator (LQR). However, the RDIP system cannot be controlled with good results when using only LQR because the system model derived as a linear model is not the exact the same as the actual system. Moreover, system parameters are subjected to change due to varying of operating conditions. The difference between the linear model and the actual model is called modeling error which creates system uncertainty. Therefore, in the controller design, robustness must be taken into account for the uncertainty. The LQR provides optimal performance for the nominal system but not for the system with model uncertainty. LQR cannot guarantee the stability and the optimal performance for the system subjected to parametric uncertainty. Neural networks based adaptive LQR is proposed to improve the performance of the balancing of the conventional LQR. Finally, the performance analysis of the proposed control algorithm from simulation and experiment are conducted.
<table>
<thead>
<tr>
<th>CHAPTER</th>
<th>TITLE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Title page</td>
<td>i</td>
</tr>
<tr>
<td></td>
<td>Acknowledgements</td>
<td>ii</td>
</tr>
<tr>
<td></td>
<td>Abstract</td>
<td>iii</td>
</tr>
<tr>
<td></td>
<td>Table of contents</td>
<td>iv</td>
</tr>
<tr>
<td></td>
<td>List of figures</td>
<td>vi</td>
</tr>
<tr>
<td></td>
<td>List of tables</td>
<td>viii</td>
</tr>
<tr>
<td>1.</td>
<td>Introduction</td>
<td>1</td>
</tr>
<tr>
<td>1.1</td>
<td>Motivation</td>
<td>1</td>
</tr>
<tr>
<td>1.2</td>
<td>Statement of problem</td>
<td>1</td>
</tr>
<tr>
<td>1.3</td>
<td>Objective of the dissertation</td>
<td>1</td>
</tr>
<tr>
<td>1.4</td>
<td>Contributions</td>
<td>2</td>
</tr>
<tr>
<td>1.5</td>
<td>Structure of the dissertation</td>
<td>2</td>
</tr>
<tr>
<td>2.</td>
<td>Literature review</td>
<td>3</td>
</tr>
<tr>
<td>2.1</td>
<td>Introduction</td>
<td>3</td>
</tr>
<tr>
<td>2.2</td>
<td>Mechanical and mathematical system modeling</td>
<td>3</td>
</tr>
</tbody>
</table>
2.3 Type of inverted pendulum configurations

3 2.3.1 Linear inverted pendulum
3 2.3.1.1 Single inverted pendulum
4 2.3.1.2 Double inverted pendulum

4 2.3.2 Rotary inverted pendulum

4 2.3.2.1 Rotary single inverted pendulum

4 2.3.2.2 Rotary double inverted pendulum

6 2.4 Methodology of Control

6 2.4.1 Balancing control

6 2.4.2 Swinging up control

7 2.5 xPC Target Implementation

3. Mathematical modeling design

8 3.1 Mathematical model of a rotary inverted pendulum

8 3.2 Mathematical model of rotary double inverted pendulum

13 3.2.1 Non-linear model of rotary double inverted pendulum

15 3.2.2 Model of direct current motor

20 3.2.3 Linearization of the system model

21 3.2.4 State space representation

22 4. Hardware development, system identification and controller design

26 4.1 Hardware development

26 4.1.1 Servo amplifier of dc motors
4.1.2 xPC Target with MF624 PCI DAQ
26
4.1.3 Quadrature incremental rotary encoder
27
4.2 System architecture of rotary inverted pendulum system
27
4.3 Controller design
29
4.3.1 Linear quadratic regulator for discrete-time state-space system
29
4.3.2 Neural network based adaptive LQR
32
4.3.2.1 System identification
34

5. Simulation and experimental results
37
5.1 Implementation and control of rotary inverted pendulum
37
5.1.1 Balancing control
37
5.1.2 Swinging-up control
37
5.1.3 Simulation and experimental results
39
5.2 Implementation and control of rotary double inverted pendulum
50
5.2.1 Numerical model of rotary double inverted pendulum
50
5.2.2 Design and Implementation
52

6. Conclusions and recommendations
73
6.1 Conclusions
73
6.2 Recommendations
73

References
74
Appendix M-file and simulink implementation program
78
I. Parameter finding of rotary inverted pendulum 78
II. Parameter finding of rotary double inverted pendulum 78
III. Matlab\simulink of nominal reference model and neural 80
network based adaptive LQR

List of figures

<table>
<thead>
<tr>
<th>FIGURE</th>
<th>TITLE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>2.1</td>
<td>Double inverted pendulum on cart</td>
<td>3</td>
</tr>
<tr>
<td>2.2</td>
<td>Furata inverted pendulum</td>
<td>5</td>
</tr>
<tr>
<td>2.3</td>
<td>Rotary inverted pendulum</td>
<td>5</td>
</tr>
<tr>
<td>2.4</td>
<td>Quaser rotary double inverted pendulum</td>
<td>6</td>
</tr>
<tr>
<td>3.1</td>
<td>Complete assembling of rotary single inverted pendulum</td>
<td>8</td>
</tr>
<tr>
<td>3.2</td>
<td>Frames and parameters of rotary single inverted pendulum</td>
<td>9</td>
</tr>
<tr>
<td>3.3</td>
<td>Rotary double inverted pendulum system</td>
<td>13</td>
</tr>
<tr>
<td>3.4</td>
<td>Structure and parameters of rotary double inverted pendulum</td>
<td>14</td>
</tr>
<tr>
<td>4.1</td>
<td>SSA Servo amplifier and block diagram</td>
<td>26</td>
</tr>
<tr>
<td>4.2</td>
<td>MF624 PCI DAQ</td>
<td>26</td>
</tr>
<tr>
<td>4.3</td>
<td>Quadrature incremental rotary encoder</td>
<td>27</td>
</tr>
</tbody>
</table>
4.4 System architecture of rotary inverted pendulum system
28
4.5 Experiment Setup
29
4.6 Block diagram of the closed-loop control system
32
4.7 Block diagram of neural network based adaptive LQR
33
4.8 The closed-loop control for nominal reference model of the system
33
4.9 Approximation model of neural network controller
34
5.1 Position control using position and rate feedback
38
5.2 Simulink block diagram for rotary inverted pendulum
40
5.3 Simulink diagram of swinging-up control
40
5.4 Simulation of swinging-up and balancing mode results of the angular position of the arm
41
5.5 Simulation of swinging-up and balancing mode results of the angular velocity of the pendulum
42
5.6 Simulation of swinging-up and balancing mode results of the angular position of the pendulum
42
5.7 Simulation of swinging-up mode results of the angular position and velocity of the pendulum
43
5.8 Diagram of rotary inverted pendulum system
44
5.9 Low-pass filter of the signal from encoder of rotary inverted pendulum
44
5.10 Block diagram for control of rotary inverted pendulum
45
5.11 Diagram for swinging up control
46
5.12 Simulink diagram for control switch
47
5.13 Experimental result of voltage control input
47
5.14 Experimental result of angular position of arm
48
5.15 Experimental results of angular velocity of arm
48
5.16 Experimental result of angular position of pendulum
5.17 Experimental result of angular velocity angle of pendulum
5.18 Photo from the experiment of swinging-up and balancing of rotary inverted pendulum
5.19 Block diagram of RDIP control
5.20 Overall block diagram of the system
5.21 The identification results of arm position by using the neural network based LQR
5.22 The identification results of arm velocity by using the neural network based LQR
5.23 The identification results of Pendulum1 position by using the neural network based LQR
5.24 The identification results of Pendulum1 velocity by using the neural network based LQR
5.25 The identification results of Pendulum2 position by using the neural network based LQR
5.26 The identification results of Pendulum2 velocity by using the neural network based LQR
5.27 The simulation results of angular position and angular velocity of arm when initial state of RDIP plant system is \([0, 0, 0.0, 0, 0]\) rad
5.28 The simulation results of angular position and angular velocity of Pendulum1 when initial state of RDIP plant system is \([0, 0, 0.0, 0, 0]\) rad
5.29 The simulation results of angular position and angular velocity of Pendulum2 when initial state of RDIP plant system is \([0, 0, 0.0, 0, 0]\) rad
5.30 The simulation results of reference command input and control input when initial state of RDIP plant system is \([0, 0, 0.0, 0, 0]\) rad
5.31 The simulation results of angular position and angular velocity of arm when initial state of RDIP plant system is \([0, 0, 0.1, 0, 0.0569, 0]\) rad
5.32 The simulation results of angular position and angular velocity of arm when initial state of RDIP plant system is \([0, 0, 0.0, 0.0569, 0]\) rad
Pendulum 1 when initial state of RDIP plant system is 
[0, 0, 0.1, 0, 0.0569 ,0] rad

5.33 The simulation results of angular position and angular velocity of
70 Pendulum 2 when initial state of RDIP plant system is
[0, 0, 0.1, 0, 0.0569 ,0] rad

5.34 The simulation results of reference command input and control input
71 when initial state of RDIP plant system is [0, 0, 0.1, 0, 0.0569 ,0] rad

List of tables

<table>
<thead>
<tr>
<th>TABLE</th>
<th>TITLE</th>
<th>PAGE</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Parameters of the model</td>
<td>11</td>
</tr>
<tr>
<td>3.2</td>
<td>Parameters of the model</td>
<td>12</td>
</tr>
<tr>
<td>3.3</td>
<td>Description of notations used in the rotary double inverted</td>
<td>14</td>
</tr>
<tr>
<td></td>
<td>pendulum system</td>
<td></td>
</tr>
<tr>
<td>3.4</td>
<td>Parameters of the model</td>
<td>20</td>
</tr>
<tr>
<td>3.5</td>
<td>Definition of the terms in $H^{-1}$ matrix</td>
<td>23</td>
</tr>
<tr>
<td>5.1</td>
<td>Values of parameters in the rotary inverted pendulum system</td>
<td>39</td>
</tr>
<tr>
<td>5.2</td>
<td>Values of parameters in the rotary double inverted pendulum</td>
<td>51</td>
</tr>
</tbody>
</table>
Chapter 1
Introduction

1.1 Motivation

Inverted pendulum is a simple structure. Control of the inverted pendulum is very challenging problem for control engineers. It is a classic experiment set-up used to study dynamics of a typical nonlinear system with an unstable equilibrium.

Many experiments are conducted to study and design new controllers for swinging-up and balancing the inverted pendulums. Among all the swinging-up methods, a controller that accumulates the swinging-up energy of the arm and the pendulum system is feasible, and effective. Practically, many factors influence the performance of the control. All these reasons make the task of swinging-up and balancing of the inverted pendulum become challenging research topic.

1.2 Statement of the problem

The problem of balancing and swinging-up of the rotary double inverted pendulum is very challenging. It requires accurate position sensor. The rotary single or double inverted pendulum system is used to evaluate and compare various control algorithms by simulations and experiments. The problem of swinging-up and balancing of an unstable system also occurs in practical control area of missile, space rockets, segweys, and robots stabilization.

The balancing and swinging-up of the rotary inverted pendulum is applied to test the performance of several control algorithms. Many researchers proposed different algorithms; such as sliding mode, PID, LQR, fuzzy control [21], high-and-low gain approach [39, 13], neural networks [19, 8, 31], and nonlinear control [42] to control the rotary inverted pendulum. Some researchers proposed the algorithm based on numerical simulation [13, 3, 28].

1.3 Objective of the dissertation

The main objective of this dissertation is to design a controller for swinging-up and balancing rotary single and double inverted pendulums in real-time. There are several sub-objectives as listed below.

- To design and construct rotary single and double inverted pendulum systems
- To derive mathematic model of rotary single and double inverted pendulum systems
- To design LQR to balance single inverted pendulum
- To design a swinging-up controller that raises single inverted pendulum to upright position
- To design a robust controller that stabilizes rotary double inverted pendulum system.
1.4 Contributions

This dissertation implements a real-time control of rotary single and double inverted pendulum systems in two cases swinging-up and balancing. The main contributions of the dissertation includes the following.

- Design and development of rotary single and double inverted pendulum systems
- Identify the mathematic models of rotary single and double inverted pendulum systems
- Design a robust and optimal control algorithm that control rotary double inverted pendulum system with uncertainty

1.5 Structure of the dissertation

This dissertation is separated into six chapters.

Chapter 2 presents a review of researches on the rotary inverted pendulum system. Chapter 3 derives a dynamic model by using Euler-Lagrange equation for rotary single and double inverted pendulum systems. Chapter 4 provides description of the rotary inverted pendulum, system identification, hardware selection, and controller design. Chapter 5 reports simulation and experimental results. Finally, the conclusions and recommendation of this dissertation are presented in chapter 6.
Chapter 2
Literature review

2.1 Introduction

This chapter presents the information relevant to the researches of rotary inverted pendulum and control of position, velocity, and acceleration of the pendulums.

2.2 Mechanical and mathematical system modeling

Inverted pendulum is a mechanical structure used to study behavior of linear and nonlinear system. Zhong and Rock [1] described the dynamic model of the inverted pendulum by using free body diagrams, Newtonian, and the Euler-Lagrange methods.

2.3 Types of inverted pendulum system

The inverted pendulum systems are classified as moving-cart inverted pendulum system and rotary inverted pendulum system.

2.3.1 Moving-cart inverted pendulum

2.3.1.1 Single inverted pendulum

Dorf and Bishop [2] presented a pendulum link attached by a pin joint on a cart. The pendulum link is directly connected to shaft of a motor as shown in Figure 2.1.

Kobayashi et al. [3] described a swinging-up controller for swinging-up the pendulum to upright position. They compared many control laws on the swinging-up time from initial state to the upright position. Sliding mode control with sinusoidal function is used to control the system.

Figure 2.1: Double inverted pendulum on cart
Rong [4] proposed a method using swinging-up energy for bringing the pendulum from down to upright position. LQR is enabled when the pendulum is close to the upright position.

Szymkat et al. [5] described an optimal control for pendulum-cart system in real-time system. They used direct synthesis of the control algorithm by rapid prototype technology, real-time, hardware-in-loop simulation.

Saifizul et al. [6] studied self-erecting and stabilizing control for inverted pendulum by using ANFIS controller. This controller system can guarantee the stability of the pendulum at upright position.

Bugeja [7] proposed a swinging-up controller method using the feedback and control of energy.

**2.3.1.2 Double inverted pendulum**

Rubi and Avello [8] proposed a swinging-up controller of double inverted pendulum on a cart by using LQR for reference trajectory on nonlinear under-actuated mechanism.

Henmi et al. [9] proposed a swinging-up control for a double inverted pendulum. The control is divided into three parts; starting from swinging-up the first pendulum, and then swinging-up the second pendulum while the first pendulum is still balanced at the upright position, and finally, the two pendulums are maintained at the unstable equilibrium point.

Zhao and Yi [10] used GA-tuned bang-bang controller to swing-up pendubot. External torque is applied to swing-up the two pendulums to upright position. The bang-bang controller was applied to the system when the two pendulums were near the down position. For balancing, LQR with bang-bang controller was applied.

Eltohamy and Kuo [11] proposed a triple inverted pendulum system. There was only one control input for stabilization the system in real-time.

**2.3.2 Rotary inverted pendulum**

**2.3.2.1 Rotary single inverted pendulum**

Furuta [12] proposed Furuta inverted pendulum system which is one of the most well-known inverted pendulum system as shown in figure 2.2.
Zhong and Rock [1] described swinging-up control of double inverted pendulum. The control algorithm for swinging-up the pendulums is based on energy shaping. The stabilizing control was proposed based on the linear model of double inverted pendulum at the equilibrium point.

Quanser[13] developed a rotary inverted pendulum system as shown in Figure 2.3. The system was constructed by attaching the pendulum link at a pin joint. The horizontal link is coupled directly via the shaft a motor with gearhead.
2.3.2.2 Rotary double inverted pendulum

Yamakita et al. [14] described swinging-up of a rotary double inverted pendulum by using two algorithms. Bang-bang with zero controller with learning was used to control the system by Anderson [19]. Narendra and Mukhopadhya applied adaptive control using neural network and approximate model [29]. They implemented the controller in real-time with limitation of physical model of the pendulum in duration of tracking.

![Quaser rotary double inverted pendulum](image)

Figure 2.4: Quaser rotary double inverted pendulum

2.4 Methodology of control

Some researchers have focused on swinging-up control to swing-up the pendulum from the stable position to the upright position by using torque from motor. Other research have focused only on balancing control.

2.4.1 Balancing control

This group of researches has focused on balancing or stabilization of rotary inverted pendulum system in order to maintain the pendulum at the upright position. The model of the rotary inverted pendulum system is derived as a nonlinear equation. The system it then linearized at near the equilibrium point. A linear equation of an inverted pendulum system was proposed by Wolfram Research [15] using Taylor series. They assumed small angular position and small angular velocity. Zhong and Rock [1] proposed LQR as an optimal controller. Utkin et al. [16] proposed the sliding mode controller for a linear model of the inverted pendulum system. The pendulum was maintained at the unstable upright equilibrium position. The sliding mode controller was used for controlling the nonlinear model by selecting a control effort to enforce sliding modes and linear behavior along sliding mode of attraction.
2.4.2 Swinging-up control

Zhong and Rock [1] proposed energy shaping control for a cart, which control the acceleration of the potential energy and kinetic energy of the system. The system maintained two links at the upright position where the potential energy equaled to the kinetic energy. The link of pendulum was controlled to have enough kinetic energy and momentum to swing-up to the vertical position.

Astrom and Furuta [17] proposed bang-bang controller to control an inverted pendulum system. The controller regulated energy consumption until it arrived the energy reference level. Swinging-up control for an inverted pendulum systems were also outlined by Kobayashi et al. [3] and Komine et al. [18].

For the goal of control was used for swinging up inverted pendulum from the down stable to upright position and then maintains the pendulum in this position by controlling of the cart position. The method of swinging up controller of the inverted pendulum system was achieved when oscillation control input frequency was a natural frequency of the pendulum.

2.5 xPC Target implementation

The xPC Target of Matlab was applied in real-time control of several prototype systems. The host PC run Matlab and Simulink block software for creating the model. The input/output blocks were then put into the block model of the system. C/C++ compiler and RTW were to generate executable codes. The codes were downloaded from the host PC to the target PC in order to run the xPC target codes in real-time kernel. The host and target can be connected through RS-232 or TCP/IP.
Chapter 3
Mathematical modeling design

3.1 Mathematical model of a rotary inverted pendulum

The rotary inverted pendulum system consists of an arm, a pendulum, a controller, an actuators (dc motor), two increments rotary encoders and a base as shown in Figure 3.1. The controller is used for making the pendulum stand still at the upright position on the rotary arm by moving the arm on the base. The dc motor provides electrical power to rotate the arm. The encoders detect the pendulum and arm angular positions. Reference frames, and parameters of the rotary inverted pendulum system are defined as shown in Figure 3.2.

Figure 3.1: Completed assembling of rotary single inverted pendulum
Figure 3.2: Frames and parameters of rotary single inverted pendulum

The variables $\theta_1$, $\dot{\theta}_1$, and $\ddot{\theta}_1$ are angular position, velocity and acceleration of the arm. $\theta_2$, $\dot{\theta}_2$, and $\ddot{\theta}_2$ are angular position, velocity and acceleration of the pendulum, respectively. $m_1$ and $m_2$ are the masses of links 1 and 2 respectively. The symbol $g$ is the gravitational acceleration and $g = 9.81 \text{ m/s}^2$. $J_1$ and $J_2$ represent the moments of inertia of the two respective links about their center of mass. $L_1$ and $L_2$ are the lengths of links 1 and 2, respectively. $l_1$ and $l_2$ are the distances from the center of rotation of the links to the center of mass of the respective links. The variables $c_1$ and $c_2$ are the viscous coefficients of the bearings on which the links rotate. The mathematical equation
describing dynamics of the rotary inverted pendulum system is derived based on the Euler-Lagrange equation of motion.

\[
\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} + \frac{\partial W}{\partial \dot{q}_i} = Q_i
\]

(3.1)

Where \( q(t) \) is the angular position vector, 
\( \dot{q}(t) \) is the angular velocity vector, 
\( Q_i \) is the external force, 
\( L \) is the Lagrangian, 
\( W \) is the loss energy.

The Lagrangian \( L \) is defined as

\[
L(q, \dot{q}) = T_{\text{total}} - V_{\text{total}}
\]

(3.2)

Where 
\( T_{\text{total}} \) is total kinetic energy of system 
\( V_{\text{total}} \) is total potential energy of system

From Figure 3.2, the total kinetic energy of the RIP system is the sum of kinetic energy of the rotary arm and the pendulum.

\[
T_{\text{total}} = T_{\text{link1}} + T_{\text{link2}}
\]

(3.3)

The kinetic energy of the rotary arm is

\[
T_{\text{link1}} = \frac{1}{2} m_1 l_1^2 \dot{\theta}_1^2 + \frac{1}{2} J_1 \dot{\theta}_1^2
\]

(3.4)

Since the center of mass of the rotary arm is balanced at the origin point, thus \( l_1 = 0 \),

\[
T_{\text{link1}} = \frac{1}{2} J_1 \dot{\theta}_1^2
\]

(3.5)

Similarly, the kinetic energy of the pendulum is determined as

\[
T_{\text{link2}} = \frac{1}{2} m_2 L_2 \dot{\theta}_2^2 + \frac{1}{2} (m_2 l_2^2 + J_2) \dot{\theta}_2^2 + m_2 L \dot{\theta}_1 \dot{\theta}_2 \cos \theta_2
\]

(3.6)
The total potential energy for the RIP system is the sum of potential energy of the arm and the pendulum.

\[ V_{\text{total}} = V_{\text{link1}} + V_{\text{link2}} \]  
(3.7)

\[ V_{\text{total}} = 0 + m_2gl_2 \cos \theta_2 \]  
(3.8)

The total loss energy of the RIP system is the sum of loss energy of the arm and the pendulum.

\[ W = \frac{1}{2} C_i \dot{\theta}_i^2 + \frac{1}{2} C_\epsilon \dot{\theta}_\epsilon^2 \]  
(3.9)

From equation (3.2), Lagrangian is determined as

\[ L = T_{\text{link1}} + T_{\text{link2}} - V_{\text{total}} \]  
(3.10)

Substitution of equations (3.5), (3.6), (3.8) and (3.9) into (3.10) yields

\[ L = \frac{1}{2} J_1 \dot{\theta}_1^2 + \frac{1}{2} m_1 l_1^2 \dot{\theta}_1^2 + \frac{1}{2} \left( m_2 l_2^2 + J_2 \right) \dot{\theta}_2^2 - m_2 gl_2 \cos \theta_2 + m_2 l_4 \dot{\theta}_4 \dot{\theta}_5 \cos \theta_2 \]  
(3.11)

The Euler-Lagrange equation of motion of each variable, thus, becomes

\[ \frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\theta}_1} \right) - \frac{\partial L}{\partial \theta_1} + \frac{\partial W}{\partial \dot{\theta}_1} = \tau_\epsilon \]  
(3.12)

\[ \frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\theta}_2} \right) - \frac{\partial L}{\partial \theta_2} + \frac{\partial W}{\partial \dot{\theta}_2} = 0 \]  
(3.13)

Substitution of (3.11) into (3.12)-(3.13) and solving of Euler-Lagrange equation yield

\[ [J_1 + m_1 l_1^2] \ddot{\theta}_1 + (m_2 l_2 \cos \theta_2) \ddot{\theta}_2 - (m_2 l_2 \sin \theta_2) \dot{\theta}_2^2 + C_i \dot{\theta}_1 = \tau_\epsilon \]  
(3.14)

\[ \left( m_2 l_1 \cos \theta_2 \right) \ddot{\theta}_1 + \left( J_2 + m_2 l_2^2 \right) \ddot{\theta}_2 + \left( m_2 gl_2 \sin \theta_2 \right) + C_\epsilon \dot{\theta}_2 = 0 \]  
(3.15)
The parameters are defined as shown in Table 3.1.

### Table 3.1 Parameters of the model

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>$h_1$</td>
<td>$J_1 + m_1l_1^2$</td>
</tr>
<tr>
<td>$h_2$</td>
<td>$m_1l_1^2$</td>
</tr>
<tr>
<td>$h_3$</td>
<td>$J_2 + m_2l_2^2$</td>
</tr>
<tr>
<td>$h_4$</td>
<td>$m_2l_2g$</td>
</tr>
<tr>
<td>$h_5$</td>
<td>$\frac{K}{R}$</td>
</tr>
<tr>
<td>$h_6$</td>
<td>$\frac{K_aK_v}{R_v}$</td>
</tr>
</tbody>
</table>

Torque, $\tau_e$, is derived from equation of dc motor

$$\tau_e = \frac{KV_a}{R_a} - \frac{K_a\omega}{R_a} = \frac{KV_a}{R_a} - \frac{K_a\dot{\theta}_1}{R_a}$$

(3.16)

After the rearrangement of (3.14)-(3.15) based on the parameters in Table 1, the non-linear model of the system is obtained as follows.

$$h_1\ddot{\theta}_1 + h_2 \cos \theta_2 \ddot{\theta}_2 - h_2 \sin \theta_2 \dot{\theta}_2^2 + C_1\dot{\theta}_1 = (h_3V_a - h_4\dot{\theta}_1)$$

(3.17)

$$h_2\ddot{\theta}_1 + h_2\dot{\theta}_2 + h_4 \sin \theta_2 + C_2\dot{\theta}_2 = 0$$

(3.18)

To linearize the model, the following relations are applied.

$$\cos \theta \approx 1, \ \sin \theta \approx \theta, \ \dot{\theta}^2 \approx 0$$

(3.19)

The linearized model, thus, becomes

$$h_1\ddot{\theta}_1 + (h_6 + C_1)\dot{\theta}_1 + h_2\ddot{\theta}_2 = h_3V_a$$

(3.20)

$$h_2\ddot{\theta}_1 + h_3\ddot{\theta}_2 + C_2\dot{\theta}_2 + h_4\dot{\theta}_2 = 0$$

(3.21)
By using the parameters defined in Table 3.2, the rotary inverted pendulum can be expressed as

\[
\dot{\theta}_1 = -d_1(h_6 + C_1)\dot{\theta}_1 - d_2 h_4 \theta_2 - C_2 d_2 \dot{\theta}_2 + d_1 h_3 V_a
\]

(3.22)

\[
\dot{\theta}_2 = -d_1(h_6 + C_1)\dot{\theta}_1 - d_1 h_4 \theta_2 - d_2 C_2 \dot{\theta}_2 + d_1 h_3 V_a
\]

(3.23)

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>(d_1)</td>
<td>(a_{11} = \frac{h_3}{\det(H)})</td>
</tr>
<tr>
<td>(d_2)</td>
<td>(a_{12} = \frac{h_2}{\det(H)})</td>
</tr>
<tr>
<td>(d_3)</td>
<td>(a_{21} = \frac{h_2}{\det(H)})</td>
</tr>
<tr>
<td>(d_4)</td>
<td>(a_{22} = \frac{h_1}{\det(H)})</td>
</tr>
<tr>
<td></td>
<td>(\det(H) = (h_1 h_3 - h_2^2))</td>
</tr>
</tbody>
</table>

The representation in state space of the rotary inverted pendulum system is shown as

\[
\begin{bmatrix}
\dot{\theta}_1 \\
\dot{\theta}_2 \\
\dot{\theta}_3 \\
\dot{\theta}_4
\end{bmatrix} =
\begin{bmatrix}
0 & 1 & 0 & 0 \\
0 & -d_1(h_6 + C_1) & -d_2 h_4 & -C_2 \\
0 & 0 & 0 & 1 \\
0 & -d_1(h_6 + C_1) & -d_1 h_4 & -d_2 C_2
\end{bmatrix}
\begin{bmatrix}
\theta_1 \\
\theta_2 \\
\theta_3 \\
\theta_4
\end{bmatrix} +
\begin{bmatrix}
0 \\
0 \\
0 \\
0
\end{bmatrix} V_a,
\]

(3.24)

3.2 Mathematical model of a rotary double inverted pendulum

In this section, mathematic model of a rotary double inverted pendulum system is derived. The rotary double inverted pendulum system is shown in Figure 3.3.
Figure 3.3: Rotary double inverted pendulum system
Figure 3.4: Structure and parameters of rotary double inverted pendulum

Figure 3.4 shows the reference frames of the rotary double inverted pendulum. The system consists of a rotary arm that rotates for maintaining two pendulums in upright position. A dc motor controls the arm position.

Table 3.3 Description of notations used in the rotary double inverted pendulum system

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description of parameter</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>$m_1$</td>
<td>The mass of the arm</td>
<td>Kg</td>
</tr>
<tr>
<td>$m_2$</td>
<td>The mass of the rod 1</td>
<td>Kg</td>
</tr>
<tr>
<td>$m_3$</td>
<td>The mass of the rod 2</td>
<td>Kg</td>
</tr>
<tr>
<td>$C_1$</td>
<td>Viscous coefficient of rotary arm</td>
<td>N.m.s</td>
</tr>
<tr>
<td>$C_2$</td>
<td>Viscous coefficient of rod 1</td>
<td>N.m.s</td>
</tr>
<tr>
<td>$C_3$</td>
<td>Viscous coefficient of rod 2</td>
<td>N.m.s</td>
</tr>
<tr>
<td>$L_1$</td>
<td>The length of arm</td>
<td>m</td>
</tr>
<tr>
<td>$L_2$</td>
<td>The length of rod 1</td>
<td>m</td>
</tr>
<tr>
<td>$L_3$</td>
<td>The length of rod 2</td>
<td>m</td>
</tr>
<tr>
<td>$l_1$</td>
<td>The distance to center of mass of arm</td>
<td>m</td>
</tr>
<tr>
<td>$l_2$</td>
<td>The distance to center of mass of rod 1</td>
<td>m</td>
</tr>
<tr>
<td>$l_3$</td>
<td>The distance to center of mass of rod 2</td>
<td>m</td>
</tr>
<tr>
<td>$J_1$</td>
<td>The moment of inertia of arm</td>
<td>Kg / m$^2$</td>
</tr>
<tr>
<td>$J_2$</td>
<td>The moment of inertia of rod 1</td>
<td>Kg / m$^2$</td>
</tr>
<tr>
<td>$J_3$</td>
<td>The moment of inertia of rod 2</td>
<td>Kg / m$^2$</td>
</tr>
<tr>
<td>$\theta_1$</td>
<td>The angular position of Arm</td>
<td>rad</td>
</tr>
<tr>
<td>$\theta_2$</td>
<td>The angular position of rod 1</td>
<td>rad</td>
</tr>
<tr>
<td>$\theta_3$</td>
<td>The angular position of rod 2</td>
<td>rad</td>
</tr>
<tr>
<td>$\theta_4$</td>
<td>The summation of angular position 2 and 3</td>
<td>rad</td>
</tr>
<tr>
<td>$g$</td>
<td>The gravitational acceleration</td>
<td>m / s$^2$</td>
</tr>
<tr>
<td>$R_a$</td>
<td>The armature resistance</td>
<td>$\Omega$</td>
</tr>
<tr>
<td>Symbol</td>
<td>Description of parameter</td>
<td>Unit</td>
</tr>
<tr>
<td>--------</td>
<td>----------------------------------</td>
<td>----------</td>
</tr>
<tr>
<td>( K_i )</td>
<td>The torque constant</td>
<td>( N\cdot M/A )</td>
</tr>
<tr>
<td>( \tau_c )</td>
<td>The control torque to rotor arm</td>
<td>( N\cdot m )</td>
</tr>
<tr>
<td>( V_u )</td>
<td>The voltage to pm dc</td>
<td>( V )</td>
</tr>
<tr>
<td>( K_v )</td>
<td>The back emf constant</td>
<td>( V.s )</td>
</tr>
</tbody>
</table>

3.2.1 Non-linear model of rotary double inverted pendulum

The model of the rotary double inverted pendulum is derived from the Euler-Lagrange equation of motion as follows.

\[
\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{q}_i} \right) - \frac{\partial L}{\partial q_i} + \frac{\partial W}{\partial \dot{q}_i} = Q_i \quad i = 1, ..., m
\]

(3.25)

Where

- \( q(t) \) is the angular position vector
- \( \dot{q}(t) \) is the angular velocity vector
- \( Q_i \) is external force vector
- \( L \) is the Lagrange
- \( W \) is the loss energy

To obtain the Euler-Lagrange equation, a collection of partial derivative of the Lagrange, \( L \) is defined as

\[
L(q, \dot{q}) = T_{total} - V_{total}
\]

(3.26)

Where

- \( T_{total} \) is total kinetic energy of rotary double inverted pendulum system
- \( V_{total} \) is total potential energy of rotary double inverted pendulum system

From Figure 3.4, the total kinetic energy for the system is the sum of the kinetic energies of the rotary arm and the two pendulums (rods), and found to be

\[
T_{total} = T_{arm} + T_{rod1} + T_{rod2}
\]

(3.27)
In order to use the Euler-Lagrange equation for describing the dynamics model of the rotary double inverted pendulum in Figure 3.4, the matrices \( q_i \) and \( Q_i \) for the rotary double inverted pendulum system are defined as follows

\[
q_i = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \theta_3 \end{bmatrix}, \quad \dot{q}_i = \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \\ \dot{\theta}_3 \end{bmatrix} \quad \text{and} \quad Q_i = \begin{bmatrix} \tau_r \\ 0 \\ 0 \end{bmatrix}
\]

(3.28)

The coordinates that describe the system are \( \theta_1, \theta_2, \theta_3 \). Since there are three different coordinates, Euler-Lagrange equation is applied to each coordinate. After solving the Euler-Lagrange equation in each coordinate, the system model is obtained. Consider kinetic energy of the arm,

\[
\begin{bmatrix} x_i \\ y_i \\ z_i \end{bmatrix} = \begin{bmatrix} l_i \cos \theta_i \\ l_i \sin \theta_i \\ 0 \end{bmatrix}
\]

(3.29)

\[
T_i = \frac{1}{2} m_i v_i^2 + \frac{1}{2} J_i \dot{\theta}_i^2
\]

(3.30)

\[
v_i^2 = [\dot{x}_i^2 + \dot{y}_i^2 + \dot{z}_i^2]
\]

(3.31)

\[
v_i^2 = [\dot{x}_i^2 + \dot{y}_i^2 + \dot{z}_i^2]
\]

(3.32)

\[
\dot{x}_i^2 = \frac{d}{dt}(l_i \cos \theta_i) = -l_i \sin \theta_i
\]

(3.33)

\[
\dot{y}_i^2 = \frac{d}{dt}(l_i \sin \theta_i) = l_i \cos \theta_i
\]

(3.34)

\[
\dot{y}_i^2 = \frac{d}{dt}(0) = 0
\]

(3.35)
\[ v_i^2 = [l_i^2 \sin^2 \theta_i \dot{\theta}_i^2 + l_i^2 \cos^2 \theta_i \dot{\theta}_i^2 + 0] \]  
(3.36)

\[ v_i^2 = l_i^2 \dot{\theta}_i^2 \]  
(3.37)

Kinetic energy of the arm, thus, becomes

\[ T_{\text{arm}} = \frac{1}{2} m_i l_i^2 \dot{\theta}_i^2 + \frac{1}{2} J_i \dot{\theta}_i^2 \]  
(3.38)

When the distance to center of mass of arm equals zero,

\[ T_{\text{arm}} = \frac{1}{2} J_i \dot{\theta}_i^2 \]  
(3.39)

Similarly, consider the kinetic energy of rod 1,

\[ T_{\text{rod1}} = \frac{1}{2} m_j [(L_1 \dot{\theta}_1 + l_2 \dot{\theta}_2 \cos \theta_2)^2 + (l_2 \dot{\theta}_2 \sin \theta_2)^2 + \frac{1}{2} J_2 \dot{\theta}_2^2] \]  
(3.40)

\[ = \frac{1}{2} m_j [L_1^2 \dot{\theta}_1^2 + 2L_1 \dot{\theta}_1 l_2 \dot{\theta}_2 \cos \theta_2 + l_2^2 \dot{\theta}_2^2 \cos^2 \theta_2 + l_2^2 \dot{\theta}_2^2 \sin^2 \theta_2 + \frac{1}{2} J_2 \dot{\theta}_2^2] \]  
(3.41)

\[ = \frac{1}{2} m_j L_1^2 \dot{\theta}_1^2 + m_j L_2 \dot{\theta}_1 l_2 \dot{\theta}_2 \cos \theta_2 + \frac{1}{2} m_j l_2^2 \dot{\theta}_2^2 + \frac{1}{2} J_2 \dot{\theta}_2^2 \]  
(3.42)

\[ \therefore T_{\text{rod1}} = \frac{1}{2} m_j L_1^2 \dot{\theta}_1^2 + \frac{1}{2} (m_j l_2^2 + J_2) \dot{\theta}_2^2 + m_j L_2 \dot{\theta}_1 l_2 \dot{\theta}_2 \cos \theta_2 \]  
(3.43)

The kinetic energy of rod 2, thus, becomes

\[ T_{\text{rod2}} = \frac{1}{2} m_j [(L_1 \dot{\theta}_1 + L_2 \dot{\theta}_2 \cos \theta_2 + l_3 \dot{\theta}_3 \cos \theta_4)^2 + (L_2 \dot{\theta}_2 \sin \theta_2 + l_4 \dot{\theta}_4 \sin \theta_4)^2] + \frac{1}{2} J_2 \dot{\theta}_4^2 \]  
(3.44)

\[ T_{\text{rod2}} = \frac{1}{2} m_j L_1^2 \dot{\theta}_1^2 + \frac{1}{2} m_j L_2^2 \dot{\theta}_2^2 + \frac{1}{2} (m_j l_2^2 + J_4) \dot{\theta}_4^2 + m_j L_2 \dot{\theta}_1 l_2 \dot{\theta}_2 \cos \theta_2 + m_j L_3 \dot{\theta}_1 l_3 \dot{\theta}_3 \cos \theta_4 \]
\[ +m_3L_4l_3\dot{\theta}_4\dot{\theta}_4\cos(\theta_2 - \theta_4) \]

(3.45)

From Figure 3.4, the total potential energy of the system is the sum of the potential energies of the rotary arm and the two pendulums (rods).

\[ V_{total} = V_{arm} + V_{rod1} + V_{rod2} \]

(3.46)

\[ V_{arm} = 0 \]

(3.47)

\[ V_{rod1} = m_2gl_2\cos\theta_2 \]

(3.48)

\[ V_{rod2} = m_3g(L_2\cos\theta_2 + l_3\cos\theta_4) \]

(3.49)

The total loss energy of the system is the sum of the loss energies for the rotary arm and the two pendulums (rods).

\[ W = \frac{1}{2}C_1\dot{\theta}_1^2 + \frac{1}{2}C_2\dot{\theta}_1^2 + \frac{1}{2}C_3\dot{\theta}_1^2 \]

(3.50)

The Lagrange, \( L \), of the system, thus, becomes

\[ L = T_{total} - V_{total} \]

(3.51)

\[ L = T_{arm} + T_{rod1} + T_{rod2} - V_{rod1} - V_{rod2} \]

(3.52)

Substitution of equations (3.43), (3.45) and (3.49) into (3.52), yields

\[ L = \frac{1}{2}J_1\dot{\theta}_1^2 + \frac{1}{2}m_2L_1\dot{\theta}_1^2 + \frac{1}{2}(m_2l_2^2 + J_2)\dot{\theta}_2^2 + m_2L_1\dot{\theta}_2\dot{\theta}_3\cos\theta_2 + \frac{1}{2}m_3L_3\dot{\theta}_3^2 + \frac{1}{2}m_4L_4\dot{\theta}_4^2 + \frac{1}{2}(m_3l_3^2 + J_3)\dot{\theta}_4^2 + m_1L_2\dot{\theta}_1\dot{\theta}_2\cos\theta_4 + m_1L_2\dot{\theta}_1\dot{\theta}_4\cos(\theta_2 - \theta_4) \]

\[ -m_2gl_2\cos\theta_2 - m_3g(L_2\cos\theta_2 + l_3\cos\theta_4) \]

(3.53)

From Euler-Lagrange equation of motion
\[
\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\theta}_1} \right) - \frac{\partial L}{\partial \theta_1} + \frac{\partial W}{\partial \dot{\theta}_1} = \tau_c
\]  
(3.54)

\[
\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\theta}_2} \right) - \frac{\partial L}{\partial \theta_2} + \frac{\partial W}{\partial \dot{\theta}_2} = 0
\]  
(3.55)

\[
\frac{d}{dt} \left( \frac{\partial L}{\partial \dot{\theta}_3} \right) - \frac{\partial L}{\partial \theta_3} + \frac{\partial W}{\partial \dot{\theta}_3} = 0
\]  
(3.56)

Substitution of equation (3.50) and (3.53) into (3.54), (3.55), (3.56), and solving the Euler-Lagrange equation, yield

\[
\begin{aligned}
&[J_1 + m_1 L_1^2 + m_2 L_2^2] \ddot{\theta}_1 + (m_1 L_1 L_2 \cos \theta_2 + m_1 L_2 L_3 \cos \theta_3) \ddot{\theta}_2 + [m_2 L_4 \cos \theta_4] \ddot{\theta}_4 \\
&- [m_1 \dot{L}_1 \dot{L}_2 \sin \theta_2 - [m_1 L_2 \sin \theta_2] \dot{\theta}_2^2 - [m_1 L_4 \sin \theta_4] \dot{\theta}_4^2 + C_1 \dot{\theta}_1 = \tau_c
\end{aligned}
\]  
(3.57)

\[
\begin{aligned}
&m_2 L_1 \cos \theta_1 \ddot{\theta}_1 + [m_1 L_1 L_2 \cos \theta_2] \ddot{\theta}_2 + [J_2 + m_2 L_2^2] \ddot{\theta}_2 + [m_2 L_4 \cos (\theta_2 - \theta_4)] \ddot{\theta}_4 \\
&+ m_2 L_4 \sin (\theta_2 - \theta_4) \ddot{\theta}_4 - [m_2 g L_2 \sin \theta_2 + m_2 g L_2 \sin \theta_2] + C_2 \dot{\theta}_2 = 0
\end{aligned}
\]  
(3.58)

\[
\begin{aligned}
&m_1 L_4 \cos \theta_4 \ddot{\theta}_1 + [m_1 L_4 \cos (\theta_2 - \theta_4)] \ddot{\theta}_2 + (J_3 + m_2 L_2^2) \ddot{\theta}_3 - m_3 L_2 \dot{L}_2^2 \sin (\theta_2 - \theta_4) \\
&m_3 L_4 \sin \theta_4 + C_3 \dot{\theta}_4 = 0
\end{aligned}
\]  
(3.59)

Rewriting equations (3.57) to (3.59), yield

\[
\begin{aligned}
&[J_1 + m_1 L_1^2 + m_2 L_2^2] \ddot{\theta}_1 + [(m_1 L_1 L_2 + m_3 L_3 L_2) \cos (\theta_2)] \ddot{\theta}_2 + [m_2 L_4 \cos \theta_4] \ddot{\theta}_4 \\
&- [m_1 \dot{L}_1 \dot{L}_2 \sin \theta_2 - [m_1 L_2 \sin \theta_2] \dot{\theta}_2^2 - [m_1 L_4 \sin \theta_4] \dot{\theta}_4^2 + C_1 \dot{\theta}_1 = \tau_c
\end{aligned}
\]  
(3.60)

\[
\begin{aligned}
&[(m_1 L_1 L_2 + m_2 L_2^2) \cos (\theta_2)] \ddot{\theta}_1 + [J_2 + m_2 L_2^2 + m_3 L_2^2] \ddot{\theta}_2 + [m_2 L_4 \cos (\theta_2 - \theta_4)] \ddot{\theta}_4 \\
&+ m_2 L_4 \sin (\theta_2 - \theta_4) - [m_2 g L_2 \sin \theta_2 + m_2 g L_2 \sin \theta_2] + C_2 \dot{\theta}_2 = 0
\end{aligned}
\]  
(3.61)

\[
\begin{aligned}
&m_1 L_4 \cos \theta_4 \ddot{\theta}_1 + [m_1 L_2 \cos (\theta_2 - \theta_4)] \ddot{\theta}_2 + [J_3 + m_2 L_2^2] \ddot{\theta}_3 - m_3 L_2 \dot{L}_2^2 \sin (\theta_2 - \theta_4) \\
&m_3 L_4 \sin \theta_4 + C_3 \dot{\theta}_4 = 0
\end{aligned}
\]  
(3.62)
Three differential equations using parameters as defined in Table 3.4 become

\[ h_1 \ddot{\theta}_1 + [(h_2 + h_3) \cos \theta_2] \ddot{\theta}_2 + [h_4 \cos \theta_4] \ddot{\theta}_4 - [h_2 + h_3 \sin \theta_2] \dot{\theta}_2^2 - [h_4 \sin \theta_4] \dot{\theta}_4^2 + C_1 \dot{\theta}_1 = \tau_e \]

(3.63)

\[ [(h_2 + h_3) \cos \theta_2] \ddot{\theta}_1 + h_3 \ddot{\theta}_2 + [h_6 \cos(\theta_2 - \theta_4)] \ddot{\theta}_4 + h_6 \sin(\theta_2 - \theta_4) \dot{\theta}_4^2 - (h_2 + h_3) \sin \theta_2 + C_2 \dot{\theta}_2 = 0 \]

(3.64)

\[ (h_4 \cos \theta_3) \ddot{\theta}_1 + h_6 \cos(\theta_2 - \theta_4) \ddot{\theta}_2 + h_6 \ddot{\theta}_4 - [h_6 \sin(\theta_2 - \theta_4)] \dot{\theta}_2^2 - h_6 \sin \theta_4 + C_3 \dot{\theta}_4 = 0 \]

(3.65)

The model is represented by non-linear mathematical equations. The angular position of the second pendulum is defined relative to the first pendulum, thus \( \theta_4 \) is substituted by \( \theta_2 + \theta_3 \).

\[ h_1 \ddot{\theta}_1 + (h_2 + h_3) \cos \theta_2 \ddot{\theta}_2 + h_4 \cos(\theta_2 + \theta_3)(\ddot{\theta}_2 + \dot{\theta}_3) - [h_2 + h_3 \sin \theta_2] \dot{\theta}_2^2 - h_4 \sin(\theta_2 + \theta_3)(\dot{\theta}_2 + \dot{\theta}_3)^2 + C_1 \dot{\theta}_1 = \tau_e \]

(3.66)

\[ [(h_2 + h_3) \cos \theta_2] \ddot{\theta}_1 + h_6 \ddot{\theta}_2 + [h_6 \cos \theta_3](\ddot{\theta}_2 + \dot{\theta}_3) + h_6 \sin \theta_3)(\dot{\theta}_2 + \dot{\theta}_3)^2 - (h_2 + h_3) \sin \theta_2 + C_2 \dot{\theta}_2 = 0 \]

(3.67)

\[ [h_4 \cos(\theta_2 + \theta_3)] \ddot{\theta}_1 + [h_6 \cos \theta_3] \ddot{\theta}_2 + h_6 (\ddot{\theta}_2 + \dot{\theta}_3) - [h_6 \sin \theta_4] \dot{\theta}_2^2 - h_6 \sin(\theta_2 + \theta_3) + C_3 (\dot{\theta}_2 + \dot{\theta}_3) = 0 \]

(3.68)

The parameters used in equations (3.66) to (3.68) are defined as shown in Table 3.4.

<table>
<thead>
<tr>
<th>Parameter</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>( h_1 )</td>
<td>( J_1 + m_2 L_1^2 + m_3 L_2^2 )</td>
</tr>
<tr>
<td>( h_2 )</td>
<td>( m_2 L_1 L_2 )</td>
</tr>
<tr>
<td>( h_3 )</td>
<td>( m_3 L_1 L_2 )</td>
</tr>
<tr>
<td>( h_4 )</td>
<td>( m_3 L_1 L_3 )</td>
</tr>
<tr>
<td>( h_5 )</td>
<td>( J_2 + m_4 L_2^2 + m_5 L_3^2 )</td>
</tr>
<tr>
<td>( h_6 )</td>
<td>( m_5 L_2 L_3 )</td>
</tr>
<tr>
<td>Parameter</td>
<td>Value</td>
</tr>
<tr>
<td>-----------</td>
<td>-------</td>
</tr>
<tr>
<td>$h_2$</td>
<td>$m_2l_2g$</td>
</tr>
<tr>
<td>$h_3$</td>
<td>$m_3l_3g$</td>
</tr>
<tr>
<td>$h_5$</td>
<td>$J_3 + m_3l_3^2$</td>
</tr>
<tr>
<td>$h_{10}$</td>
<td>$m_3l_3g$</td>
</tr>
<tr>
<td>$h'_{13}$</td>
<td>$\frac{K_i}{R_a}$</td>
</tr>
<tr>
<td>$h'_{12}$</td>
<td>$\frac{K_iK_v}{R_a}$</td>
</tr>
</tbody>
</table>

### 3.2.2 Model of direct current motor

In a direct current motor, the back-emf, $E_a$, is proportional to the rotor speed.

$$E_a = K_v \omega_r = K_v \dot{\omega}_i$$  \hspace{1cm} (3.69)

The electromagnetic torque is proportional to the armature current.

$$\tau_e = K_i I_a = K_v I_a$$  \hspace{1cm} (3.70)

Assuming that the armature inductance is very small and negligible and applying the Kirchhoff of voltage of the armature circuit result in

$$V_a = R_a I_a + E_a$$  \hspace{1cm} (3.71)

Armature current, $I_a$, is determined as follows

$$I_a = \frac{V_a}{R_a} = \frac{E_a}{R_a} = \frac{V_a}{R_a} = \frac{K_v \omega_r}{R_a}$$  \hspace{1cm} (3.72)

Substitution of equations (3.72) into (3.70) yields the following result.

$$\tau_e = \frac{K_v V_a}{R_a} - \frac{K_iK_v \omega_r}{R_a} = \frac{K_v V_a}{R_a} - \frac{K_iK_v \dot{\omega}_i}{R_a}$$  \hspace{1cm} (3.73)
Where $V_a$ is the applied dc voltage source. The electromagnetic torque of a dc motor is determined. The motor used in the system should have sufficient power to generate sufficient torque and sufficient speed. The torque is necessary for the rotary arm to change direction of rotation quickly enough in order to maintain the two pendulums balance. Sufficient speed is required such that the rotary arm is able to move fast enough before the two pendulums fall.

### 3.2.3 Linearization of the system model

The dynamic model of the system is linearized before expressing in the state-space form.

\[
 h_1 \ddot{\theta}_1 + (h_2 + h_3) \cos \theta_2 \dot{\theta}_2 + h_4 \cos(\theta_2 + \theta_3)(\ddot{\theta}_2 + \dot{\theta}_3) - [h_2 + h_3 \sin \theta_2] \ddot{\theta}_2^2 \\
 - h_1 \sin(\theta_2 + \theta_3)(\ddot{\theta}_2 + \dot{\theta}_3)^2 + C_1 \ddot{\theta}_1 = (h_1 V_a - h_2 \dot{\theta}_1)
\]

(3.74)

\[
 [(h_2 + h_3) \cos \theta_2] \ddot{\theta}_2 + h_7 \dot{\theta}_2 + [h_8 \cos \theta_3](\dot{\theta}_2 + \dot{\theta}_3) + h_9 \sin \theta_3(\ddot{\theta}_2 + \dot{\theta}_3)^2 \\
 - (h_7 + h_8) \sin \theta_2 + C_2 \dot{\theta}_2 = 0
\]

(3.75)

\[
 [(h_4 \cos(\theta_2 + \theta_3)] \ddot{\theta}_1 + [h_5 \cos \theta_3]\dot{\theta}_2 + h_6(\dot{\theta}_2 + \dot{\theta}_3) - [h_6 \sin \theta_3] \ddot{\theta}_2^2 \\
 - h_{10} \sin(\theta_2 + \theta_3) + C_3(\ddot{\theta}_2 + \dot{\theta}_3) = 0
\]

(3.76)

When, the two rods are close to the upright position. The system can be linearized using small angle approximation.

\[
 \cos \theta \approx 1 \quad \sin \theta \approx \theta \quad \dot{\theta}^2 \approx 0
\]

(3.77)

Applying the small angle approximation into equations (3.74) to (3.76) yields the following linear equations.

\[
 h_1 \ddot{\theta}_1 + (h_{12} + C_1) \dot{\theta}_1 + (h_2 + h_3 + h_4) \dot{\theta}_2 + h_5 \dot{\theta}_3 = h_1 V_a
\]

(3.78)

\[
 (h_2 + h_3) \ddot{\theta}_1 + (h_5 + h_6) \ddot{\theta}_2 + C_2 \dot{\theta}_2 - (h_7 + h_8) \theta_2 + h_9 \dot{\theta}_3 = 0
\]

(3.79)

\[
 h_6 \ddot{\theta}_1 + (h_8 + h_9) \ddot{\theta}_2 + C_3 \dot{\theta}_2 - h_{10} \theta_2 + h_9 \dot{\theta}_3 + C_3 \dot{\theta}_3 - h_{10} \theta_3 = 0
\]

(3.80)
3.2.4 State space representation

Equations (3.51) to (3.53) are expressed in the matrix form as follows.

\[
\begin{bmatrix}
  h_1 & (h_2 + h_3 + h_4) & h_4 \\
  (h_2 + h_3) & (h_3 + h_6) & h_6 \\
  h_4 & (h_6 + h_8) & h_8 \\
\end{bmatrix}
\begin{bmatrix}
  \dot{\theta}_1 \\
  \dot{\theta}_2 \\
  \dot{\theta}_3 \\
\end{bmatrix}
= \begin{bmatrix}
  h_1 V_a - (h_2 + C_1) \dot{\theta}_1 \\
  (h_2 + h_6) \dot{\theta}_2 - C_2 \dot{\theta}_2 \\
  h_6 \dot{\theta}_3 - C_3 \dot{\theta}_3 - C_4 \dot{\theta}_2 + h_6 \dot{\theta}_2 \\
\end{bmatrix}
\]

(3.81)

The acceleration \( \ddot{\theta}_1, \ddot{\theta}_2 \) and \( \ddot{\theta}_3 \) can be determined by multiplying both sides of the equation by the inverse of the coefficient matrix.

\[
\begin{bmatrix}
  \ddot{\theta}_1 \\
  \ddot{\theta}_2 \\
  \ddot{\theta}_3 \\
\end{bmatrix}
= \begin{bmatrix}
  h_1 & (h_2 + h_3 + h_4) & h_4 \\
  (h_2 + h_3) & (h_3 + h_6) & h_6 \\
  h_4 & (h_6 + h_8) & h_8 \\
\end{bmatrix}^{-1}
\begin{bmatrix}
  h_1 V_a - (h_2 + C_1) \dot{\theta}_1 \\
  (h_2 + h_6) \dot{\theta}_2 - C_2 \dot{\theta}_2 \\
  h_6 \dot{\theta}_3 - C_3 \dot{\theta}_3 - C_4 \dot{\theta}_2 + h_6 \dot{\theta}_2 \\
\end{bmatrix}
\]

(3.82)

The inverse of the coefficient matrix \( H \) is calculated by

\[
H^{-1} = \frac{1}{\det(H)} \text{adj}(H)
\]

(3.83)

When the determinant of \( H \) is

\[
\det(H) = h_1 h_5 h_9 + 2 h_2 h_3 h_6 + 2 h_2 h_4 h_6 - 2 h_2 h_3 h_6 - h_2^2 h_6 - h_3^2 h_5 - h_6^2 h_1
\]

(3.84)

Therefore, the inverse of \( H \) is

\[
H^{-1} = \frac{1}{\det(H)} \begin{bmatrix}
  a_{11} & a_{12} & a_{13} \\
  a_{12} & a_{12} & a_{23} \\
  a_{13} & a_{23} & a_{33} \\
\end{bmatrix}
\]

(3.85)

Where

\[
\begin{align*}
a_{11} &= h_5 h_9 - h_6^2 \\
a_{12} &= h_3 h_6 - h_2 h_9 - h_3 h_9 \\
a_{13} &= h_2 h_6 + h_3 h_6 - h_3 h_9 \\
a_{21} &= h_4 h_6 - h_2 h_9 - h_3 h_9 \\
\end{align*}
\]
The moment of inertia of the arm (rectangular shape) and the rods (solid circular shape) are functions of their masses and lengths.

The moment of inertia of the arm (rectangular shape) is

\[ J_1 = \frac{1}{12} m_1 (a^2 + b^2) \]

(3.86)

The moment of inertia of rod 1 (solid circular shape) is

\[ J_2 = m_2 r^2 \]

(3.87)

The moment of inertia of rod 2 (solid circular shape) is

\[ J_3 = m_3 r^2 \]

(3.88)

Since the values of \( m_2, m_3, L_1 \) and \( L_2 \) are all positive, thus

\[ \det(H) > 0 \]

(3.89)

Substitution of \( H^{-1} \) into equations (3.85) results in the following equation.

\[
\begin{bmatrix}
\dot{\theta}_1 \\
\dot{\theta}_2 \\
\dot{\theta}_3 \\
\end{bmatrix} =
\begin{bmatrix}
d_1 & d_2 & d_3 \\
d_2 & d_4 & d_3 \\
(d_3 - d_2) & (d_3 - d_4) & (d_6 - d_3) \\
\end{bmatrix}
\begin{bmatrix}
h_{11} V_a - (h_{12} + C_1) \dot{\theta}_1 \\
(h_{12} + h_{20}) \dot{\theta}_2 - C_2 \dot{\theta}_2 \\
h_{40} \dot{\theta}_3 - C_3 \dot{\theta}_3 - C_2 \dot{\theta}_2 + h_{10} \dot{\theta}_2 \\
\end{bmatrix}
\]

(3.90)

Table 3.5 shows the definition of the terms in \( H^{-1} \).

<table>
<thead>
<tr>
<th>Constant</th>
<th>Value</th>
<th>Constant</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>( d_1 )</td>
<td>( \frac{a_{11} - h_1 h_6 - h_4^2}{\det(H)} )</td>
<td>( d_4 )</td>
<td>( \frac{a_{12}}{\det(H)} = \frac{h_3 h_6 - h_1 h_9 - h_6 h_4}{\det(H)} )</td>
</tr>
</tbody>
</table>

Table 3.5 Definition of the terms in \( H^{-1} \) matrix
The differential equations are expressed by

\[ \ddot{\theta}_i = d_1 h_1 V_a (d_1 h_{12} + d_1 C_i) \dot{\theta}_1 + (d_2 h_7 + d_2 h_8 + d_3 h_{10}) \theta_2 - \big( d_2 C_2 + d_3 C_3 \big) \dot{\theta}_2 - d_4 C_4 \dot{\theta}_3 + d_1 h_{10} \theta_3 \]  
(3.91)

\[ \ddot{\theta}_2 = d_2 h_1 V_a (d_2 h_{12} + d_2 C_4) \dot{\theta}_1 + (d_4 h_7 + d_4 h_8 + d_5 h_{10}) \theta_2 - \big( d_4 C_2 + d_5 C_3 \big) \dot{\theta}_2 - d_5 C_3 \dot{\theta}_3 + d_2 h_{10} \theta_3 \]  
(3.92)

\[ \ddot{\theta}_3 = [d_3 h_1 - d_2 h_{12}] V_a (d_2 h_{12} + d_2 C_1 - d_3 h_{12} - d_3 C_1) \dot{\theta}_1 + [d_4 C_2 + d_5 C_3 - d_4 C_2 - d_6 C_3] \dot{\theta}_2 + [d_4 C_2 + d_5 C_3 - d_6 C_3] \dot{\theta}_3 + [d_4 h_{10} - d_5 h_{10} + d_5 h_7 + d_6 h_8 - d_5 h_7 - d_6 h_8] \theta_2 + \big[ d_4 h_{10} - d_5 h_{10} \big] \theta_3 \]  
(3.93)

In order to express the model of the rotary double inverted pendulum into the state-space form, the following state variables are defined.

\[ x_1 = \theta_1, x_2 = \dot{\theta}_1, x_3 = \theta_2, x_4 = \dot{\theta}_2, x_5 = \theta_3, \text{ and } x_6 = \dot{\theta}_3 \]  
(3.94)

The state-space representation of the system is in the form of

\[ \dot{x} = Ax + BV_a \]
\[ y = Cx + DV_a \]  
(3.95)

The matrices A, B and C of the rotary double inverted pendulum are defined as follows.
When
\[
\begin{align*}
a_{22} &= d_1 h_{12} + d_1 C_1 \\
a_{23} &= d_2 h_{22} + d_2 C_3 \\
a_{24} &= d_3 h_{32} + d_3 C_4 \\
a_{25} &= d_4 h_{42} + d_4 C_5 \\
a_{26} &= -d_5 C_2 \\
a_{42} &= -d_2 h_{22} + d_2 C_1 \\
a_{43} &= d_3 h_{32} + d_3 C_3 \\
a_{44} &= -d_4 C_2 + d_4 C_3 \\
a_{45} &= d_5 h_{52} \\
a_{46} &= -d_5 C_3
\end{align*}
\]

The rotary double inverted pendulum is represented in state space form as
\[
\begin{bmatrix}
\dot{x}_1 \\
\dot{x}_2 \\
\dot{x}_3 \\
\dot{x}_4 \\
\dot{x}_5 \\
\dot{x}_6
\end{bmatrix} =
\begin{bmatrix}
0 & 1 & 0 & 0 & 0 & 0 \\
0 & a_{22} & a_{23} & a_{24} & a_{25} & a_{26} \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & a_{42} & a_{43} & a_{44} & a_{45} & a_{46} \\
0 & 0 & 0 & 0 & 0 & 1 \\
0 & a_{62} & a_{63} & a_{64} & a_{65} & a_{66}
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2 \\
x_3 \\
x_4 \\
x_5 \\
x_6
\end{bmatrix} +
\begin{bmatrix}
0 \\
d_1 h_{11} \\
0 \\
d_2 h_{11} \\
0 \\
(d_3 h_{11} - d_2 h_{11})
\end{bmatrix} \cdot V_u,
\end{equation}
\[
\begin{bmatrix}
x_1 \\
x_2 \\
x_3 \\
x_4 \\
x_5 \\
x_6
\end{bmatrix} =
\begin{bmatrix}
0 \\
d_1 h_{11} \\
0 \\
d_2 h_{11} \\
0 \\
(d_3 h_{11} - d_2 h_{11})
\end{bmatrix} +
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2 \\
x_3 \\
x_4 \\
x_5 \\
x_6
\end{bmatrix},
\end{equation}
\[
\begin{bmatrix}
x_1 \\
x_2 \\
x_3 \\
x_4 \\
x_5 \\
x_6
\end{bmatrix} =
\begin{bmatrix}
0 \\
d_1 h_{11} \\
0 \\
d_2 h_{11} \\
0 \\
(d_3 h_{11} - d_2 h_{11})
\end{bmatrix} +
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & 0 & 0 & 0 & 1 & 0 \\
0 & 0 & 0 & 0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2 \\
x_3 \\
x_4 \\
x_5 \\
x_6
\end{bmatrix}
\]
Chapter 4
Hardware development, system identification, and controller design

4.1 Hardware development
4.1.1 Servo amplifier of DC Motor

Servo amplifier is used to drive a dc motor to rotate the arm of the rotary inverted pendulum system. The servo amplifier used in the system is made of power MOSFET, which works with high power with high efficiency in small size.

![SSA Servo amplifier and block diagram]

4.1.2 xPC Target with MF624 PCI DAQ
xPC Target from Matlab operates in real-time. The code can be developed in Simulink block and can be run on any personal computers (PC) with input/output boards. The host PC runs MATLAB/Simulink and the target PC runs the program on real-time kernel. The host PC generates and compiles the program and sends the compiled code to the target PC through TCP/IP protocol.

![Figure 4.2: MF624 PCI DAQ](image)

4.1.3 Quadrature incremental rotary encoder

Quadrature incremental encoders are used to measure the position and velocity of the rotary double inverted pendulum. The incremental rotary encoder consists of a slotted wheel and an emitter/detector module attached to the shaft of the dc motor. The quadrature incremental rotary encoder has three outputs, channel A, channel B and Index.

The relation between channels A and B is used to determine the rotating direction. When channel A is leading channel B, the dc motor is rotating in a direction. In contrary, when channel A is lagging channel B, the dc motor is rotating in opposite direction. The Index is used as a reference position for one revolution.

The quadrature decoder captures the phase signals, index pulse and converts the data into a numeric count of the position pulses. The count of the position increases when the shaft of the encoder is rotating in one direction and decreases when it is rotating in the opposite direction. The quadrature incremental rotary encoder is shown in Figure 4.3.
4.2 System architecture of rotary inverted pendulum system

The architecture of the rotary inverted pendulum system by xPC target with real time control [27] is shown in Figure 4.4.

Figure 4.4: System architecture of rotary inverted pendulum system

From Figure 4.4, the system consists of two stations. The first station is called the host PC, and the second is called the target PC. The host PC runs on Microsoft Windows XP, which has Matlab/Simulink, Real-time workshop (RTW), xPC target, and C/C++ compiler. The simulink is used to model the system and the controller. RTW and C/C++ compiler convert Simulink model into C code and build a target that is then downloaded to the target PC through TCP/IP protocol. The target PC runs on DOS system in real time,
which is booted from floppy disk. The executable code is generated from the host PC. The hardware consists of

- The host PC station
- The target PC station
- Humusoft multifunction card model MF624
- Elmo (SSA) Servo amplifier
- Brush type DC motor and incremental rotary encoder

Figure 4.5: Experiment Setup

4.3 Controller design
4.3.1 Linear-quadratic regulator (LQR) for discrete-time state-space system
The discrete-time linear quadratic regulator (LQR) can be used to control the pendulums to stand at upright equilibrium point. Consider a state controllable linear system expressed by given linear discrete-time control system

\[ x(k + 1) = A_dx(k) + B_du(k) \]  \hspace{1cm} (4.1)

Where it is assumed to be completely state controllable. Defining performance index for a finite time process

\[ J = \frac{1}{2} x(N)Sx(N) + \frac{1}{2} \sum_{k=0}^{N-1} \left[ x(k)Qx(k) + u(k)Ru(k) \right] \]  \hspace{1cm} (4.2)

Where

\[ 0 \leq k \leq N \]

\[ Q = 6 \times 6 \text{ positive definite Matrix} \]
\[ R = 1 \times 1, \text{ matrix} \]
\[ S = 6 \times 6 \text{ positive definite Matrix} \]

Matrices \( Q, R \) and \( S \) are selected to weigh the relative importance of the performance measures. Let initial condition on state vector be \( x(0) = 0 \). Now by using a set of Lagrange multipliers \( \lambda (1), \lambda (2), \ldots, \lambda (N) \), we define new performance index \( L \) as follows

\[ L = \frac{1}{2} x(N)Sx(N) + \frac{1}{2} \sum_{k=0}^{N-1} \left[ \left[ x(k)Qx(k) + u(k)Ru(k) \right] + \lambda(k+1) \right] \]
\[ + \left[ A_dx(k) + B_du(k) - x(k+1) \right] \lambda(k+1) \]  \hspace{1cm} (4.3)

To minimize \( L \), the partial derivatives w.r.t. \( x_i(k), u_i(k), A_i(k) \) must be zero. Since the partial derivatives w.r.t. vectors are defined as,

\[ \frac{\partial}{\partial x} x^T Ax = Ax \quad \frac{\partial}{\partial u} x^T Ay = Ay \]  \hspace{1cm} (4.4)

When

\[ \frac{\partial L}{\partial x_i(k)} = 0; \quad Qx(k) + A_d^T \lambda(k+1) - \lambda(k) = 0, k = 1, 2, \ldots, N-1 \]  \hspace{1cm} (4.5)
\[ \frac{\partial L}{\partial x(N)} = 0; \quad Sx(N) - \lambda(N) = 0 \]  \hspace{1cm} (4.6)
\[ \frac{\partial L}{\partial u(k)} = 0; \quad RU(k) + B_d^T \lambda(k+1) = 0, k = 1, 2, \ldots, N-1 \]  \hspace{1cm} (4.7)
\[ \frac{\partial L}{\partial A_i(k)} = 0; \quad A_dx(k-1) + A_du(k+1) - x(k) = 0, k = 1, 2, \ldots, N-1 \]  \hspace{1cm} (4.8)
Now, from equation (4.9), (4.13) and (4.11) optimal control vector \( \lambda(k) \) becomes,

\[
\lambda(k) = Qx(k) + A_{d}'\lambda(k+1),
\]

(4.9)

\[
u(k) = -R^{-1}B_{d}'\lambda(k+1),
\]

(4.10)

And

\[x(k+1) = A_{d}x(k) + B_{d}u(k+1).\]

(4.11)

From equation (4.10) and (4.11) can be derived by

\[x(k+1) = A_{d}x(k) - B_{d}R^{-1}B_{d}'\lambda(k+1)\]

(4.12)

To obtain solution of minimization problem, I need to solve Equation (4.11) and (4.12) simultaneously. Defining input in the form \( u(k) = -K(k)x(k) \) is 1x6 feedback matrix. Now, I will obtain the optimal control vector \( u(k) \) in the closed loop form by first obtaining the Riccati Equation. Assuming that

\[
\lambda(k) = P(k)x(k).
\]

(4.13)

Where \( \Lambda(k) \) is 6x6 Hermitian matrices. Substituting it in equation (4.11)

\[
P(k)x(k) = Qx(k) + A_{d}'P(k+1)x(k+1)
\]

(4.14)

\[x(k+1) = A_{d}x(k) - B_{d}R^{-1}B_{d}'P(k+1)x(k+1)\]

(4.15)

Substituting equation (4.15) to (4.14) yield

\[
P(k) = Q + A_{d}'P(k+1)[1 + B_{d}R^{-1}B_{d}'P(k+1)]^{-1}A_{d}
\]

(4.16)

Now, from equation (4.9), (4.13) and (4.11) optimal control vector \( u(k) \) becomes,

\[
u(k) = -R^{-1}B_{d}'\Lambda(k+1) = -R^{-1}B_{d}'(A_{d})^{-1}[\Lambda(k) - Qx(k)]
\]

\[= -R^{-1}B_{d}'(A_{d})^{-1}[P(k) - Q]x(k)
\]

\[= -R(k)x(k)\]

(4.17)
Where
\[ K(k) = -R^{-1}B_d'(A_d')^{-1}[P(k) - Q] \]  
(4.18)

For \( N = \infty \), the performance index may be modified to
\[ J = \frac{1}{2} \sum_{k=0}^{\infty} [x(k)Qx(k) + u(k)Ru(k)] \]  
(4.19)

If optimal regulator is stable, the term \( \frac{1}{2}x(\infty)Sx(\infty) = 0 \). Let define the steady state matrix \( P(k) \) as \( P \), referring to equation (4.14) yield
\[ P = Q + A_d'P[1 + B_dR^{-1}B_d'P]^{-1}A_d = Q + A_d'[P^{-1} + B_dR^{-1}B_d']^{-1}A_d \]  
(4.20)

The steady state gain matrix \( K \) can be found from equation (4.18) as following
\[ K(k) = -R^{-1}B_d'(A_d')^{-1}[P(k) - Q] \]  
(4.21)

The optimal control law for steady state operation is given by \( u(k) = -Kx(k) \). The linear quadratic regulator provides one or more control variables that are formed as a linear combination of the RDIP system states, as expressed in the following equation. The gain matrix, \( K \), consists of six feedback gains that are determined from controller design of the optimal process.
4.3.2 Neural network based Adaptive LQR

The neural network based adaptive LQR is used to stabilize control of RDIP system. The nonlinear system can be transformed into multiple linear systems at certain operating points. Figure 4.7 shows a block diagram of the RDIP system controlled by neural network based adaptive LQR. The block diagram of controller consists of the LQR and a neural network model. LQR is applied as a main controller for the RDIP system. The neural network is used to improve performance of the conventional LQR controller. Input of the neural network is the error between the actual and the reference models. Output of the neural networks is summed with the LQR output to balance the system.
Figure 4.7: Block diagram of neural network based adaptive LQR

Figure 4.8 shows the reference model, the closed-loop of the nominal model of the RDIP system which is controlled by LQR. $d$ is the disturbance in the system. $K$ is the LQR gain.

Figure 4.8: The closed-loop control for nominal model of the system

4.3.2.1 System identification

The neural networks is used to identify the nominal model of the RDIP system controlled by LQR. The neural network is trained in order to represent the dynamics of the system. The first stage for nonlinear identification is the selection of the structure of the neural networks model using feedback linearization as:
\[ X_p(k+1) = f[X_p(k), X_p(k-1),..., X_p(k-n+1), e(k-1),..., e(k-m+1)] + \\
g[X_p(k), X_p(k-1),..., X_p(k-n+1), u(k-1),..., u(k-m+1)].u(k) \]

(4.22)

Equation (4.22) can be represented as a neural network model as shown in Figure 4.9.

Figure 4.9: Approximation model of neural network controller

The neural network uses two sub networks (\(\text{NN}_1\) and \(\text{NN}_2\)) for approximation of nonlinear function \(\hat{f}\) and \(\hat{g}\) respectively. The nonlinear function of \(\hat{f}\) and \(\hat{g}\) are used to identify the system of equation (4.23) as

\[ \hat{X}_p(k+1) = \hat{f}[X_p(k), X_p(k-1),..., X_p(k-n+1), e(k-1),..., e(k-m+1)] + \\
\hat{g}[X_p(k), X_p(k-1),..., X_p(k-n+1), e(k-1),..., e(k-m+1)].u(k) \]

(4.23)

Where 
- \(u(k)\) is system input
- \(X_p(k)\) is the system state
- \(k\) is the sampling instant number
- \(m\) is the number of part inputs
- \(n\) is the number of part outputs

The system has to be regulated. The system identification of the model of neural network controller starts from collecting the dataset of input. The collected dataset consists of two parts: one part is used for the training the neural network and the other part is used for validating the neural network model. The \(\text{NN}_1\) is a feedforward neural network plus one
hidden layer with $p$ neurons of hyperbolic tangent (tanh) activation functions and includes the output layer of one neural with linear activation function. Also, the NN$_2$ is a feedforward neural network plus one hidden layer with $q$, tanh, neurons, and one output layer. The neural network is trained by using the input and the desired output from the dataset. The weights and biases of the two sub networks that approximate the nonlinear model of the RDIP system are determined. The nominal reference model is defined as

$$\dot{X}_{m} = A_{m}X_{m}(t) + B_{m}r(t)$$

(4.24)

The control input for the nominal reference model is designed as

$$u_{m}(t) = -K_{m}(r(t) - X_{m}(t))$$

(4.25)

Apply neural network to the RDIP system,

$$X^{(n)}_{p} = f(X_{p}) + g(X_{p})u$$

(4.26)

Where $X_{p} = [x_{p}, \frac{dx_{p}}{dt}, \ldots, \frac{d^{(n-1)}x_{p}}{dt^{(n-1)}}]^T$ is state vector of the system. $u$ is the control input

From the block diagram in figure 4.7, the control input is described as follows.

$$u = \frac{1}{g(X_{p})}[-f(X_{p}) - K^{T}(X_{p} - d) + r]$$

(4.27)

Where

$K$ is the feedback gains and

$d$ is the disturbance for testing the system

$r$ is reference input

The approximate functions $f(\cdot)$ and $g(\cdot)$ are substituted by the neural networks NN$_1$ and NN$_2$. Thus the control input to the RDIP system is defined as follows.

$$u = \frac{1}{NN_{2}(X_{p})}[-NN_{1}(X_{p}) - K^{T}(X_{p} - d) + r]$$

(4.28)

The system follows the reference model as
Substitution of equation (4.28) into equation (4.26) yields

\[ X_p^{(n)} = f(X_p) + \frac{g(X_p)}{NN_2(X_p)}[-NN_1(X_p) - K^T X_p - d] + r \]  

(4.30)

The controller error is defined as

\[ e = X_p - X_m \]  

(4.31)

Then, the error differential equation is expressed by

\[ e^{(n)} = -K^T e + \{ f(X_p) - NN_1(X_p) \} + \{ g(X_p) - NN_2(X_p) \} u \]  

(4.32)

The error differential equation will be stable with the appropriate training algorithm. The error will converge to zero when the structure error terms are sufficiently small.
Chapter 5
Simulation and experimental results

5.1 Implementation and control of rotary inverted pendulum

5.1.1 Balancing control

The balancing control is based on linear quadratic regulator (LQR). If all of the state variables are measurable, the control vector is determined from

\[ u = -Kx(t) \]

(5.1)

The performance index (PI) is defined as

\[ J = \frac{1}{2} \int_{0}^{\infty} [x'(t)Qx(t) + u'Ru(t)]dt \]

(5.2)

5.1.2 Swinging-up control

Swinging-up control is used to swing-up the inverted pendulum to the upright position. The swinging-up controller brings the pendulum upright close to the unstable equilibrium point. Swinging-up of a rotating type pendulum from the down position state to the inverted state is one of most difficult control problems, since the system is nonlinear, under-actuated, and has uncontrollable states. Swinging-up of the pendulum upright using minimal energy is achieved when the oscillation frequency of control input is the natural frequency \( \omega_n \) of the rotary inverted pendulum. The control function is given by

\[ V = A \sin(\omega_n t) \]

(5.3)

\[ \omega_n = \sqrt{\frac{3g}{2l}} \]

(5.4)

The term \( A \) of the control function adjusts the amplitude of the sinusoidal function. However, the motor voltage is limited to \( \pm 10V \) during swinging up; any values of \( A \)
above this value will be saturated. The swinging-up controller is designed to swing-up the pendulum close to upright position from the stable downward position, where the balancing controller can takeover. Many algorithms can be applied to provide driving torque for suitable trajectory in such the manner that energy is gradually added to the system for bringing the pendulum to the upright position. The swinging-up controller applies a rate feedback and a position feedback as expressed by

$$V_i(s) = K_p[\theta_i(s) - \theta_n(s)] - K_p\Omega_n(s)$$  \hspace{1cm} (5.5)$$

This feedback control system [11] is shown in Figure 5.1. The goal of this system is to control the output angle, $\theta_n(t)$ to follow the desired position $\theta_i(t)$. The inner loop for position control is designed to meet the following time domain specifications:

- The step response damping ratio $\zeta = 0.8$
- The step response peak time $t_p = 0.1$ second

By applying the Mason’s gain formula, the overall transfer function of the system becomes

$$\theta_n = \frac{K_p a_m}{s^2 + (K_D a_m + b_m)s + K_p a_m}$$  \hspace{1cm} (5.6)$$

$$a_m = \frac{K_m}{R_s J_m} \text{ and } b_m = \frac{B_m}{J_m} + \frac{K_m^2}{R_s J_m}$$  \hspace{1cm} (5.7)$$

The peak time of the second-order response $t_p$, is given by
Standard second order transfer functions is in the form

\[
\frac{\theta_n}{\theta_i} = \frac{\omega_n^2}{s^2 + 2\zeta \omega_n s + \omega_n^2}
\]

(5.9)

From equation (5.6)

\[
K_p = \frac{\omega_n^2}{a_m}, \quad K_d = \frac{2\zeta \omega_n - b_m}{a_m}
\]

(5.10)

From the specifications, \( K_p \) and \( K_d \) can be determined.

5.1.3 Simulation and experimental results

The values of parameters of the rotary inverted pendulum system are measured, identified and calculated from the real system as shown in Table 1. These values are then used in the state space model.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Value</th>
<th>Symbol</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>( m_1 ) (kg)</td>
<td>0.83</td>
<td>( l_1 ) (kg/m²)</td>
<td>0.00208</td>
</tr>
<tr>
<td>( m_2 ) (kg)</td>
<td>0.10</td>
<td>( l_2 ) (kg/m²)</td>
<td>0.001</td>
</tr>
<tr>
<td>( L_1 ) (m)</td>
<td>0.60</td>
<td>( g ) (m/s²)</td>
<td>9.81</td>
</tr>
<tr>
<td>( L_2 ) (m)</td>
<td>0.30</td>
<td>( K_y ) (Ω)</td>
<td>28.6</td>
</tr>
<tr>
<td>( C_i ) (Nms)</td>
<td>0</td>
<td>( K_e ) (Vs)</td>
<td>0.168</td>
</tr>
<tr>
<td>( C_s ) (Nms)</td>
<td>0</td>
<td>( K_t ) (NmA)</td>
<td>1.68</td>
</tr>
<tr>
<td>( l_1 ) (m)</td>
<td>0.30</td>
<td>( \tau_x ) (N-m)</td>
<td>0.0981</td>
</tr>
<tr>
<td>( l_2 ) (m)</td>
<td>0.10</td>
<td>( V_y ) (V)</td>
<td>60</td>
</tr>
</tbody>
</table>

After substitution the values from Table 5.1 into the equations, matrices \( A, B, \) and \( C \) representing the state space system are obtained.
\[ \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \\ \dot{x}_3 \\ \dot{x}_4 \end{bmatrix} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -0.4915 & 14.6564 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 1.4744 & -93.0191 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} + \begin{bmatrix} 0 \\ 29254 \\ 0 \\ -8781 \end{bmatrix} V_a \]

\[ y = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} \theta_1 \\ \theta_2 \\ \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \end{bmatrix} V_a \]

(5.11)

When

\[ x_1 = \theta_1 \text{ (Arm Position), } x_2 = \dot{\theta}_1 \text{ (Arm Velocity),} \]
\[ x_3 = \theta_2 \text{ (Pendulum Position), } x_4 = \dot{\theta}_2 \text{ (Pendulum Velocity)} \]

By using MATLAB, the LQR of the rotary inverted pendulum can be designed according to the weight matrices. By choosing

\[ Q = \text{diag}([60, 0.1, 60, 0.1]) \]
\[ R = 1 \]

(5.12)
The optimal gains are obtained

\[ K = [7.746, 3.6278, -4.4714, -0.0469] \]

(5.13)

By using MATLAB/Simulink, the simulation results are shown in Figure 5.3-5.4. The results show the balancing of the pendulum is achieved. The simulink diagram is constructed with state feedback as shown in Figure 5.2
Figure 5.2: Simulink block diagram for rotary inverted pendulum

Simulink diagram of the swinging-up controller of the rotary inverted pendulum is shown in Figure 5.3. The results of simulation are shown in Figure 5.6, and Figure 5.7.

Figure 5.3: Simulink diagram of swinging-up control
Figure 5.4: Simulation of swinging-up and balancing mode results of the angular position of the arm

Figure 5.5: Simulation of swinging-up and balancing mode results of the angular velocity of the arm
Figure 5.6: Simulation of swinging-up and balancing mode results of the angular position of the pendulum

Figure 5.7: Simulation of swinging-up and balancing mode results of the angular velocity of the arm

In the simulation, the weighing matrices are chosen so that the closed-loop response meets the following specifications:
1. Arm Regulation: $|\theta_1| < 120^\circ$
2. Pendulum Regulation: $|\theta_2| < 3^\circ$
3. Control input limit: $V_{\text{in}} < \pm 10V$

The controller should be able to regulate the arm about zero degree within $120^\circ$ as to balance the pendulum with the angle $\theta_2$ within $3^\circ$. The diagram of the closed-loop system used as swinging-up controller and balancing controller is depicted in Figure 5.8.

Two encoders are used for measurement of the angular position of the arm, $\theta_1$ and the pendulum position $\theta_2$. The state variables $\dot{\theta}_1$, and $\dot{\theta}_2$ are obtained as the first time derivatives of the position. The low-pass band-limited filter $\frac{50}{s + 50}$ and $\frac{500}{s + 500}$ are applied to $\theta_1$, and $\theta_2$ to suppress noise in the system as shown in Figure 5.9.
Figure 5.9: Low-pass filter of the signal from encoder of rotary inverted pendulum

The control of the rotary inverted pendulum is implemented as shown in Figure 5.10. The block of analog output is obtained from MF624 library. Balancing control mode and swinging-up mode can be selected manually.

![Block diagram for control of rotary inverted pendulum](image)

Figure 5.10: Block diagram for control of rotary inverted pendulum

In the swinging-up mode, the positive feedback based on the pendulum angle and its velocity creates a trajectory with growing amplitude. The gains $P$ and $D$ to swing-up
the pendulum smoothly are $K_p = 10$, $K_D = 0.765$ as shown in the block diagram in Figure 5.11.

![Diagram for swinging-up control](image)

**Figure 5.11: Diagram for swinging-up control**

The control switch is shown in Figure 5.12. It is used to track the pendulum angle, $\theta_2$, and to switch between the swing-up and balancing modes. Balancing controller is enabled when $\theta_2$ is in the neighborhood of zero, within $\pm 3^\circ$, and for as long as $|\theta_2| < 25^\circ$. 
Figure 5.12: Simulink diagram for control switch

The experimental results of swinging-up and balancing control of the rotary inverted pendulum is shown in Figure 5.13 to Figure 5.17.
Figure 5.13: Experimental result of voltage control input

Figure 5.14: Experimental result of angular position of arm
Figure 5.15: Experimental result of angular velocity of arm

Figure 5.16: Experimental result of angular position of pendulum
Figure 5.17: Experimental result of angular velocity of pendulum
Photo from the experiment of swing-up and balancing of the rotary inverted pendulum is shown in Figure 5.18.
Figure 5.18: Photo from the experiment of swinging-up and balancing of rotary inverted pendulum

5.2 Implementation and control of rotary double inverted pendulum
5.2.1 Numerical model of rotary double inverted pendulum

Values of parameters of the nominal model of the rotary double inverted pendulum system are identified and found as shown in Table 5.1. These values are used in the state space model to obtain a numerical representation for the purpose to design the controller.

<table>
<thead>
<tr>
<th>Symbol</th>
<th>Description of parameter</th>
<th>Value</th>
<th>Unit</th>
</tr>
</thead>
<tbody>
<tr>
<td>$m_1$</td>
<td>Mass of rotary arm</td>
<td>0.7</td>
<td>kg</td>
</tr>
<tr>
<td>$m_2$</td>
<td>Mass of pendulum #1</td>
<td>0.10</td>
<td>kg</td>
</tr>
<tr>
<td>$m_3$</td>
<td>Mass of pendulum 2</td>
<td>0.05</td>
<td>kg</td>
</tr>
<tr>
<td>$L_1$</td>
<td>Length of rotary arm</td>
<td>0.2</td>
<td>m</td>
</tr>
<tr>
<td>$L_2$</td>
<td>Length of pendulum 1</td>
<td>0.205</td>
<td>m</td>
</tr>
<tr>
<td>$L_3$</td>
<td>Length of pendulum 2</td>
<td>0.40</td>
<td>m</td>
</tr>
<tr>
<td>$C_1$</td>
<td>Viscous coefficient of rotary arm</td>
<td>0</td>
<td>N.m.s</td>
</tr>
<tr>
<td>$C_2$</td>
<td>Viscous coefficient of pendulum 1</td>
<td>0</td>
<td>N.m.s</td>
</tr>
<tr>
<td>$C_3$</td>
<td>Viscous coefficient of pendulum 2</td>
<td>0</td>
<td>N.m.s</td>
</tr>
<tr>
<td>$l_1$</td>
<td>Length to center of mass of rotary arm</td>
<td>0.10</td>
<td>m</td>
</tr>
<tr>
<td>$l_2$</td>
<td>Length to center of mass of pendulum 1</td>
<td>0.10</td>
<td>m</td>
</tr>
<tr>
<td>$l_3$</td>
<td>Length to center of mass of pendulum 2</td>
<td>0.20</td>
<td>m</td>
</tr>
<tr>
<td>$J_1$</td>
<td>Moment of inertia of rotary arm</td>
<td>0.00068012</td>
<td>kg/m^2</td>
</tr>
<tr>
<td>$J_2$</td>
<td>Moment of inertia of pendulum 1</td>
<td>0.00001623</td>
<td>kg/m^2</td>
</tr>
<tr>
<td>$J_3$</td>
<td>Moment of inertia of pendulum #2</td>
<td>0.00000806</td>
<td>kg/m^2</td>
</tr>
<tr>
<td>$g$</td>
<td>Gravitational acceleration constant</td>
<td>9.81</td>
<td>m/s^2</td>
</tr>
<tr>
<td>$R_a$</td>
<td>Armature resistance</td>
<td>28.6</td>
<td>Ω</td>
</tr>
<tr>
<td>$K_e$</td>
<td>Back E.M.F constant</td>
<td>0.168</td>
<td>V.s</td>
</tr>
<tr>
<td>$K_t$</td>
<td>Torque Constant</td>
<td>1.68</td>
<td>N.m/A</td>
</tr>
<tr>
<td>Symbol</td>
<td>Description of parameter</td>
<td>Value</td>
<td>Unit</td>
</tr>
<tr>
<td>--------</td>
<td>--------------------------</td>
<td>-------</td>
<td>------</td>
</tr>
<tr>
<td>( r_e )</td>
<td>Control torque to rotary arm</td>
<td>0.0981</td>
<td>( N \cdot m )</td>
</tr>
<tr>
<td>( V_s )</td>
<td>Voltage to dc motor</td>
<td>75</td>
<td>V</td>
</tr>
</tbody>
</table>

The state variables are defined as follows:

\[ x_1 = \theta_1 \] (Arm Position), \( x_2 = \dot{\theta}_1 \) (Arm Velocity), \( x_3 = \theta_2 \) (Pendulum #1 Position), \( x_4 = \dot{\theta}_2 \) (Pendulum #1 Velocity), \( x_5 = \theta_3 \) (Pendulum #2 Position), and \( x_6 = \dot{\theta}_3 \) (Pendulum #2 Velocity). Substitution of the values from Table 3. into the linearized nominal model of RDIP results

\[
\frac{dx}{dt} =
\begin{bmatrix}
0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0.028721 & 0.71398 & 0 & -1.2994 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & -1.0002 & 3.3926 & 0 & 1.3792 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0.23073 & 0.94737 & 0.16773 & 0 & 1.3734 \\
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2 \\
x_3 \\
x_4 \\
x_5 \\
x_6 \\
\end{bmatrix}
+ \begin{bmatrix}
0 \\
x_1 \\
x_2 \\
x_3 \\
x_4 \\
x_5 \\
\end{bmatrix} \cdot V_s,
\]

(5.13)

\[
y =
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 1 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2 \\
x_3 \\
x_4 \\
x_5 \\
x_6 \\
\end{bmatrix}
\]

(5.14)

In the LQR design, the weighting matrices \( Q \) and \( R \) are defined as follows.

\[
R = \begin{bmatrix}
500 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 100 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 100 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}
\]

(5.15)

The optimal gain, \( K \), of the nominal plant model and the actual plant is obtained.

\[
K = [-22.36, -49.494, 2585.5, 1334.2, 1067.3, 567.63]
\]

(5.16)

### 5.2.2 Design and Implementation

The overall block diagram of the system is shown in Figure 4.7 in the previous chapter. The implementation of the control system of RDIP is detailed in the block
diagram as shown in Figures 5.19 - 5.20. The Matlab/Simulink source code is included in Appendix III.

**Figure 5.19: Block diagram of RDIP control**

**Figure 5.20: Overall block diagram of the system**
The data set of RDIP system \((x_1, x_2, ..., x_6)\) is prepared by using the model as shown in equation (3.97) when 10% noise is included to the nominal model.

\[
\begin{bmatrix}
\dot{x}_1 \\
\dot{x}_2 \\
\dot{x}_3 \\
\dot{x}_4 \\
\dot{x}_5 \\
\dot{x}_6 \\
\end{bmatrix} =
\begin{bmatrix}
0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0.0316 & 0.7854 & 0 & -1.4293 \\
0 & 0 & 0 & 1 & 0 & 0 \\
0 & -0.1100 & 3.7318 & 0 & 1.5172 & 0 \\
0 & 0 & 0 & 0 & 0 & 1 \\
0 & 0.2538 & 1.0421 & 0 & 0.1845 & 0 \\
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2 \\
x_3 \\
x_4 \\
x_5 \\
x_6 \\
\end{bmatrix} +
\begin{bmatrix}
0 \\
0.1881 \\
0 \\
0.6549 \\
0 \\
-1.5107 \\
\end{bmatrix} V_n. 
\tag{5.17}
\]

\[
y =
\begin{bmatrix}
1 & 0 & 0 & 0 & 0 & 0 \\
0 & 1 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 1 & 0 & 0 \\
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2 \\
x_3 \\
x_4 \\
x_5 \\
x_6 \\
\end{bmatrix} 
\tag{5.18}
\]

In the LQR design, the weighting matrices \(Q\) and \(R\) are defined as follows.

\[
R = [1], \quad Q =
\begin{bmatrix}
500 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 100 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 100 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 \\
\end{bmatrix}
\tag{5.19}
\]

The optimal gain, \(K\) of the actual RDIP model is obtained.

\[
K = [-22.36, -49.494, 2585.5, 1334.2, 1067.3, 567.63] 
\tag{5.20}
\]

The controller is designed by using Matlab/Simulink with the NN toolbox (Appendix III). The network architecture (number of inputs, outputs, layers, type and number of neurons in each layer), training data, limitations and training parameters (number of epochs, training algorithm) are defined. In the simulations, the following parameters are selected, size of hidden layer = 13, sampling interval = 0.01 (sec), training data sample = 3000, training epochs = 200 and training function = trainlm. The voltage range is between -100 to 100 volts. The neural network is trained in a batch mode by Lavenberg Marquardt algorithm. The first step in the controller design process is the system modeling. The performance of neural network based adaptive LQR is highly dependent on the accuracy of the system identification. In the system identification, a series of pulses with random
amplitude is presented to the RDIP system, and the NN model output of the rotary arm, pendulum 1 and pendulum 2 are then recorded as shown in Figure. 5.21-5.26. The data is divided into training, testing, and validation parts.

(a) Training data
Testing data

Validation data sets

Figure. 5.21: The identification results of arm position by using the neural network based adaptive LQR

(a) Training data
(b) Testing data

(c) Validation data sets
Figure 5.22: The identification results of arm velocity by using the neural network based adaptive LQR

(a) Training data

(b) Testing data
(c) Validation data sets

Figure 5.23: The identification results of Pendulum1 position by using the neural network based adaptive LQR

(a) Training data
(b) Testing data

Figure 5.24: The identification results of Pendulum1 velocity by using the neural network based adaptive LQR

(c) Validation data sets

Figure 5.24: The identification results of Pendulum1 velocity by using the neural network based adaptive LQR
(a) Training data

(b) Testing data
(c) Validation data sets

Figure 5.25: The identification results of Pendulum2 position by using the neural network based adaptive LQR

(a) Training data
Testing data
The learning stops when the error of the model of the system is sufficiently small. Both the LQR and neural network based adaptive LQR are investigated and compared in numerous simulations. When supply the disturbance as a sinusoidal of magnitude of 0.100 N-m at 0.8 Hz plus band-limited white noise disturbance input, the output from the system is shown in Figure. 5.27-5.34.
Figure. 5.27: The simulation results of angular position and angular velocity of arm when initial state of RDIP plant system is \([0, 0, 0, 0, 0, 0]\) rad.
Simulation results of angular position of Pendulum 1

Simulation results of velocity angle of Pendulum 1

Figure. 5.28 : The simulation results of angular position and angular velocity of Pendulum 1 when initial state of RDIP plant system is \([0, 0, 0, 0, 0, 0]\) rad
Figure 5.29: The simulation results of angular position and angular velocity of Pendulum 2 when initial state of RDIP plant system is [0, 0, 0, 0, 0, 0] rad
Figure 5.30: The simulation results of disturbance input and control input when initial state of RDIP plant system is \([0, 0, 0, 0, 0, 0]\) rad
Figure 5.31: The simulation results of angular position and angular velocity of arm when initial state of RDIP plant system is [0, 0, 0.1, 0, 0.0569, 0] rad
Figure 5.32: The simulation results of angular position and angular velocity of Pendulum1 when initial state of RDIP plant system is [0, 0, 0.1, 0, 0.0569 ,0] rad
Figure. 5.33 : The simulation results of angular position and angular velocity of Pendulum2 when initial state of RDIP plant system is \([0, 0, 0.1, 0, 0.0569 ,0]\) rad
Figure. 5.34: The simulation results of disturbance input and control input when initial state of RDIP plant system is \( [0, 0, 0.1, 0, 0.0569, 0] \) rad
From figures 5.27-5.34, we can see that the Pendulum 1 and Pendulum 2 still oscillate when the conventional LQR is applied. In contrary, when the RDIP system is controlled by neural network based adaptive LQR, there are small oscillation of Pendulum 1 and Pendulum 2, and the system smoothly approaches to the reference position.
Chapter 6
Conclusions and recommendations

6.1 Conclusions

In this dissertation, the implementation of control algorithm was separated into two parts. In the first part, energy based PD control and LQR optimal controller were applied for swinging-up and balancing of a rotary inverted pendulum. In the real-time control, xPC Target was used as the real-time computation platform at 0.001 sec sampling rate. The results from both simulations and experiments confirmed the control performance of the method. In the second part, the stabilization of rotary double inverted pendulum was simulated. Neural network based adaptive LQR, which guarantees stability of system with larger range of uncertainty, was proposed. At the training stage, the LQR controller was applied as reference input to the neural network. The controller learns to generate correction to LQR to compensate the sub-optimality of the LQR.

6.2 Recommendations

In the rotary inverted pendulum, during balancing mode, small movement of the rotating arm is still visible, especially in the region around zero velocity. An accurate value of friction could improve the control performance. The swinging-up controller that was implemented neglects the effect of the rotational motion of the pendulum joint. Although the system worked quite well, a quicker swinging-up might be possible when the dynamics of these rotations are taken into account. The nonlinear quadratic term of the velocity should be considered. Since the theory of forwarding does not support this, the forwarding procedure achieved local asymptotic stability, but does not guarantee global convergence. Outside a certain attraction region, unstable behavior might occur.
References


Appendix
M-file and simulink implementation program

I. Parameter finding of rotary inverted pendulum

\[
\begin{align*}
m_1 &= 0.83; \quad m_2 = 0.1; \quad c_1 = 0; \quad c_2 = 0; \quad L_1 = 0.6; \quad L_2 = 0.3; \quad l_1 = 0.3; \quad l_2 = 0.10; \\
J_1 &= 0.00208; \\
J_2 &= 0.001; \quad V_A = 60; \\
R_A &= 28.6; \quad g = 9.81; \quad K_t = 1.68; \quad K_v = 0.168; \quad T_e = 0.0981; \\
h_1 &= J_1 + m_2 L_1^2; \\
h_2 &= m_2 L_1 l_2; \\
h_3 &= J_2 + m_2 l_2^2; \\
h_4 &= m_2 l_2 g; \\
h_5 &= K_t / R_A; \\
h_6 &= K_t K_v / R_A; \\
DetH &= (h_1 h_3 - h_2^2); \\
d_1 &= h_3 / DetH; \\
d_2 &= -h_2 / DetH; \\
d_3 &= -h_2 / DetH; \\
d_4 &= h_1 / DetH; \\
A &= \begin{bmatrix} 0 & 1 & 0 & 0; \\
0 & d_1 (h_6) & -d_2 h_4 & 0; \\
0 & 0 & 0 & 1; \\
0 & -d_3 h_6 & -d_4 h_4 & 0 \end{bmatrix}; \\
B &= \begin{bmatrix} 0; \\
d_1 h_5; \\
0; \\
d_3 h_5 \end{bmatrix}; \\
C &= \begin{bmatrix} 1, 0, 0, 0; 
\end{bmatrix}
\end{align*}
\]
D = [0; 0; ];
p = eig(A);
C'*C;

Q= diag([60, 0.1, 60, 0.5]);
R = 1;
K = dlqr(A, B, Q, R)
Ac = [(A - B*K)];
Bc = [B];
Cc = [C];
Dc = [D];

II. Parameter finding of rotary double inverted pendulum

m1=0.7; m2=0.1; m3=0.05; c1=0; c2=0; c3=0; L1=0.2; L2=0.205; L3=0.4; l1=0.1; l2=0.1; l3=0.2; J1=0.00068012;
J2=0.000016129; J3=0.0000080645;
Ra=28.6; g=9.81; Kt=1.68; Kv=0.168; Te=0.0981;
h1=J1+m2*L1^2+m3*L1^2;
h2=m2*L1+l1;
h3=m3*L1*L2;
h4=m3*L1+l3;
h5=J2+m2*l2^2+m3*l2^2;
h6=m3*L2+l3;
h7=m2*l2*g;
h8=m3*L2*g;
h9=J3+m3*l3^2;
h10=m3*l3*g;
h11=Kt/Ra;
h12=Kt*Kv/Ra;
detH=h1*h5*h9+2*h2*h4*h6+2*h3*h4*h6-2*h2*h3*h9-h2^2*h7-h3^2*h9-h4^2*h5-h6^2*h1;
d1=(h5*h9-h6^2)/detH;
d2=(h4*h6-h2*h9-h3*h9)/detH;
d3=(h2*h6+h3*h6-h4*h5)/detH;
d4=(h4*h6-h2*h9-h3*h9)/detH;
d5=(h2*h4+h3*h4-h1*h6)/detH;
d6=(h1*h5-2*h2*h3-h2*h3*h3)/detH;
a22=d1*h12+d1*c1;
a23=d2*h7+d2*h8+d3*h10;
a24=-d2*c2-d3*c3;
a25=d3*h10;
a26=-d3*c3;
a42=-d2*h12-d2*c1;
a43=d4*h7+d4*h8+d5*h10;
a44=-d4*c2-d5*c3;
a45=d5*h10;
a46=-d5*c3;
a62=d2*h12+d2*c1-d3*h12-d3*c1;
a63=d6*h10-d5*h10+d5*h7+d5*h8-d4*h7-d4*h8;
a64=d4*c2+d5*c3-d5*c2-d6*c3;
a65=d6*h10-d5*h10;
a66=d5*c3-d6*c3;
b22=d1*h11;  
b24=d2*h11;  
b26=d3*h11-d2*h11;

A=[0 1 0 0 0 0;  
   0 a22 a23 a24 a25 a26;  
   0 0 0 1 0 0;  
   0 a42 a43 a44 a45 a46;  
   0 0 0 0 0 1;  
   0 a62 a63 a64 a65 a66];

B=[0;  
   b22;  
   0;  
   b24;  
   0;  
   b26];

C=[1,0,0,0,0,0;  
   0,0,1,0,0,0;  
   0,0,0,0,1,0];

D=[0;0;0];  
Q=diag([500,0,100,0,100,0]);  
R=1;  
K=lqr(A,B,Q,R);

III. Matlab\Simulink of nominal reference model and neural network based adaptive LQR

To run this RDIP model, follow these steps:

1. Start MATLAB®.
2. Run the RDIP model by typing `nnrdip` in the MATLAB Command Window. This command starts Simulink and creates the following model window. The NARMA-L2 Control block is already in the model.
3. Double-click the NARMA-L2 Controller block. This brings up the following window.

4. This window enables you to train the NARMA-L2 model. There is no separate window for the controller, because the controller is determined directly from the model, unlike the model predictive controller.
The nominal reference model is shown as following
5. This window works the same as the other Plant Identification windows, so the training process is not repeated. Instead, simulate the NARMA-L2 controller.

6. Return to the Simulink model and start the simulation by choosing the **Start** command from the **Simulation** menu. As the simulation runs, the plant output and the reference signal are displayed, as in the following Figure 5.21-5.26.