Delft University of Technology

SC52035 Integration Project

# Report

*Delft, October 21, 2021*

5217350 Zhehan Li
5227828 Vivek Varma

# Contents

# 1 Introduction

In this report, we design state feedback controllers for a rotary inverted pendulum, which is mounted to a Quanser QUBE-Servo 2 unit. We use a linear quadratic regulator(LQR) and Pole Placement methods in order to find suitable controller gains for the QUBE-Servo 2 system. The essential characteristics of the QUBE-Servo 2 unit are presented, the system is modelled, identified, optimised, validated, and the performances of the closed-loop systems are evaluated and compared based on rise time, settling time and overshoot of the rotary arm's step response. The design is validated using real-time experiments. The resulting controllers stabilize the rotary pendulum at the upright position and can also rotate the rotary arm and pendulum system to the desired angular position while keeping it balanced. We also design a swing-up controller to swing the pendulum up and balance it at the upright position.

Inverted pendulum systems have been studied extensively. Plenty of literature is available with detailed experiments performed on a variety of different pendulum setups. The Quanser QUBE-Servo 2 unit is a small-scale design platform for a variety of control methods. The characteristics of the system is captured by a Single Input Multiple Output(SIMO) model.

The material is organized in the following order-

- Section 2 presents the modelling and derivation of the linear state space approximation of the model.
- Section 3 goes through the calibration procedure for both the arms of the system
- Section 4 explores how we optimize parameters using the grey-box estimation method.
- Section 5 address our controller designs(LQR and pole placement) as well as our swing-up controller design.
- Section 6 deals with the closed-loop system identification where we again optimize parameters using grey-box estimation method, at upright position with our designed controllers.
- Section 7 is for the summary and concluding remarks.

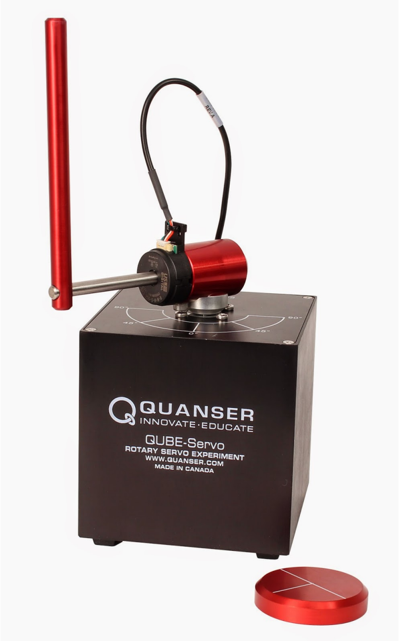

Figure 1.1: Quanser QUBE-Servo 2 Inverted Pendulum

# 2 Model of the Inverted Pendulum System

Before thinking of control strategies, we first need to understand how this system works. Given just the system and input-output data, our objective is to find a linear state space model of the system. There are several ways to approach this. In the Filtering and Identification course, we learnt about subspace identification methods like N4SID which can be used to obtain the state matrices from data. Another approach that we learnt is from Modelling of Dynamical Systems, where we can use the system dynamics to obtain a mathematical relation between quantities using the Euler-Lagrange equations, which can then be linearized(for a non-linear system) and represented in the state space form. We will be using the Euler-Lagrange Energy method for modelling our system, since it is a relatively simple mechanical system, which can be modelled using basic physics. The argument for this will be further discussed in Section 4.1.3.

## 2.1 Measurement convention

The Qube/Rotary Pendulum is an under-actuated system since only one of the rotations is forced where the other one has a free response to it. The rotation of the arm, $\alpha$, is actuated by putting a torque on the motor. The position of $\alpha$ is measured with an optical encoder. The motion, $\beta$, of the pendulum rod is a free rotation depending on the actuated arm rotation.
The convention of two angle measurements and one input are shown in Figure 2.1, and summarized below:

- The rotary arm angle, $\alpha$, increases positively when it rotates counter-clockwise (CCW), and should turn in the CCW direction when the control voltage $V_m$ is positive.

- The unit of $\alpha$ is in radians. The value of $\alpha$ is zero at the initial position.

- The pendulum angle, $\theta$, is 0 when it stays at the downright position. From downright position to the upright position, $\theta$ increases from 0 when rotated CCW. The value of $\theta$ at the top is $\pi$.

- The unit of $\theta$ is in radians. The range of the $\theta$ is $[0, 2\pi)$.

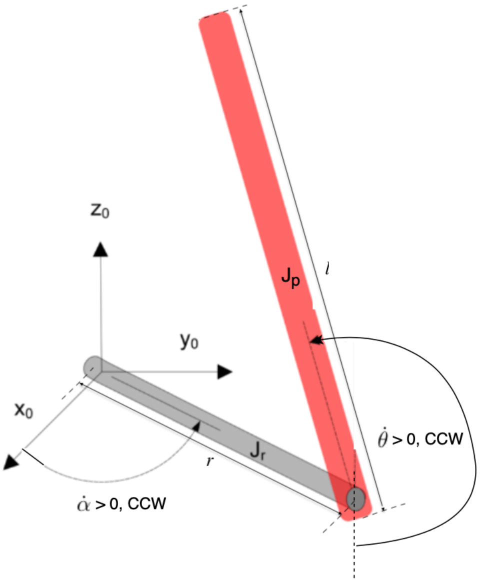

Figure 2.1: Qube/Rotary Pendulum conventions

In Figure 2.1, $x_0$ is the initial position of the rotary arm, and the $\alpha$ increases because the arm is rotating CCW($\dot{\alpha} > 0$). $z_0$ is the upright direction of the pendulum, and the $\theta$ also increases because the pendulum is rotating CCW($\dot{\theta} > 0$). The $\theta$ will become $\pi$ when it reaches the upright position, and goes back to 0 when the pendulum is again at the downright position.

The angular measurement data recorded from the system or raw data, are defined in counts. These values are a result of the output of the optical encoder and servo potentiometer. This raw data has already been converted to useful data, in radians, instead of counts, to obtain the desired input signals. Therefore, the units of the angular data does not need to be converted by us.

## 2.2 Energy of the System

The first step in the Euler-Lagrange derivation is to come up with an expression for the Kinetic Energy(KE) and Potential Energy(PE) of the system. We can refer to the model in Figure 2.1. We consider the center of mass of both these arms to be at their center. Let $m_r$ be the mass of the rotary arm, and $m_p$ be the mass of the pendulum arm, both represented about their center of mass. Let $l$ be the length of the pendulum arm, and $r$ be the length of the rotary arm. Therefore, the COMs of the arms are at $r/2$ and $l/2$ length from their ends.
Now we write out the expression for the potential energy $V$ of the system. If we take the relative potential at the height of the rotary arm to be 0, the only term that will contribute to the potential energy is the pendulum arm since it is hanging down(not in the xy plane).

$$V = -\frac{mglcos\theta}{2} \tag{1}$$

The divided-by-2 factor in the denominator comes from the fact that the COM is at half the length of the pendulum.
Now we write out the Kinetic Energy of the system. We express the COM of the pendulum arm with respect to the generalized coordinates. Let us call this $p_p$.

$$p_p = [rcos\alpha - \frac{lsin\alpha sin\theta}{2}, rsin\alpha + \frac{lcos\alpha sin\theta}{2}, -\frac{lcos\theta}{2}] \tag{2}$$

This is computed simply by taking components along the x, y and z axes. For the kinetic energy however, we need a velocity term. The velocity term of the COM is simply the time differentiation of the position of the COM.

$$v_p = [-r\dot{\alpha}sin\alpha - \frac{l\dot{\theta}sin\alpha cos\theta}{2} - \frac{l\dot{\alpha}cos\alpha sin\theta}{2}, r\dot{\alpha}cos\alpha + \frac{l\dot{\theta}cos\alpha cos\theta}{2} - \frac{l\dot{\alpha}sin\alpha sin\theta}{2}, \frac{l\dot{\theta}sin\theta}{2}] \tag{3}$$

The total kinetic energy($T$) of the system is the sum of the rotational kinetic energies of the arm and the pendulum and the transnational kinetic energy of the pendulum.

$$T = \frac{v_p m_p v_p^T}{2} + \frac{J_p\dot{\theta}^2}{2} + \frac{J_r\dot{\alpha}^2}{2} \tag{4}$$

Here $J_r$ and $J_p$ represent the moments of inertia of the rotary arm and the pendulum arm respectively. Substituting the expression for $v_p$ from Equation 3 into Equation 4, we get:

$$T = \frac{m_p\dot{\alpha}^2r^2}{2} + \frac{m_p\dot{\alpha}^2(\frac{l}{2})^2sin^2\theta}{2} + \frac{m_p\dot{\theta}^2(\frac{l}{2})^2}{2} + \frac{m_prl\dot{\alpha}\dot{\theta}}{2} + \frac{J_p\dot{\theta}^2}{2} + \frac{J_r\dot{\alpha}^2}{2} \tag{5}$$

We take a viscous damping force, which will be a non-conservative force. The expression for it is [1]

$$Q = \begin{bmatrix} -b_r\dot{\alpha} + \tau_m \\ -b_p\dot{\theta} \end{bmatrix} \tag{6}$$

## 2.3 Lagrangian and Non-Linear Equations of Motion

The Lagrange equations of motion are

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}} - \frac{\partial L}{\partial q} = Q \tag{7}$$

where $L$ is the Lagrangian or $T - V$ in mathematical terms, and $q$=$[\alpha, \theta]$. Computing this, and rearranging the terms in terms of $\ddot{\alpha}, \ddot{\theta}, \dot{\alpha}, \dot{\theta}$, we get [1]

$$\begin{bmatrix} J_r + m_p r^2 + m_p \frac{l^2}{2} \sin^2\theta & m_p r \frac{l}{2} \cos\theta \\ m_p r \frac{l}{2} \cos\theta & J_p + m_p \frac{l^2}{4} \end{bmatrix} \begin{bmatrix} \ddot{\alpha} \\ \ddot{\theta} \end{bmatrix} +$$
$$\begin{bmatrix} b_r + m_p \frac{l^2}{4} \dot{\theta} \sin\theta \cos\theta & m_p \frac{l^2}{4} \dot{\alpha} \sin\theta \cos\theta - m_p r \frac{l}{2} \dot{\theta} \sin\theta \\ -m_p \frac{l^2}{4} \dot{\alpha} \sin\theta \cos\theta & b_p \end{bmatrix} \begin{bmatrix} \dot{\alpha} \\ \dot{\theta} \end{bmatrix} + \tag{8}$$
$$\begin{bmatrix} 0 \\ m_p g \frac{l}{2} \sin\theta \end{bmatrix} = \begin{bmatrix} \tau_m \\ 0 \end{bmatrix}$$

Now we need to find an expression for $\tau_m$ in terms of the input voltage. Refer to Figure 2.2 where $u$ is the voltage input to the motor, $V_m = Ku$ is the PWM Amplified voltage, $R_a, L_a$ are resistance and inductance of the circuit respectively. $E_b$ is the back EMF of the motor, which is proportional to the rate of change of magnetic flux and hence proportional to the angular velocity of the motor($E_b = K_b \dot{\alpha}$). The torque is proportional to the armature current. Therefore,

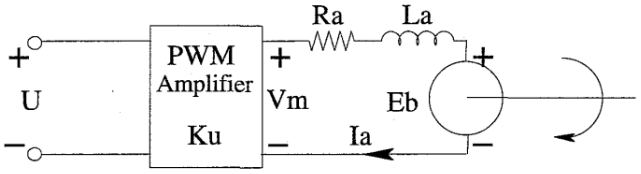

Figure 2.2: Motor Voltage to Torque Conversion Circuit

$$\tau_m = K_m I_a = \frac{K_m \left(V_m - E_b - L_a \frac{dI_a}{dt}\right)}{R_m} \tag{9}$$

Assuming the coil inductance does not have much influence, we can omit the inductance term.

$$\tau_m = \frac{K_m \left(V_m - E_b\right)}{R_m} = \frac{K_m K_u}{R_m} u - \frac{K_m K_u}{R_m} \dot{\alpha} \tag{10}$$

Combining the Voltage to Torque conversion in Equation 10 with the Euler-Lagrange equations in Equation 8, we get

$$\begin{bmatrix} J_r + m_p r^2 + m_p \frac{l^2}{2} \sin^2\theta & m_p r \frac{l}{2} \cos\theta \\ m_p r \frac{l}{2} \cos\theta & J_p + m_p \frac{l^2}{4} \end{bmatrix} \begin{bmatrix} \ddot{\alpha} \\ \ddot{\theta} \end{bmatrix} +$$
$$\begin{bmatrix} b_r + \frac{K_t K_b}{R_a} + m_p \frac{l^2}{4} \dot{\theta} \sin\theta \cos\theta & m_p \frac{l^2}{4} \dot{\alpha} \sin\theta \cos\theta - m_p r \frac{l}{2} \dot{\theta} \sin\theta \\ -m_p \frac{l^2}{4} \dot{\alpha} \sin\theta \cos\theta & b_p \end{bmatrix} \begin{bmatrix} \dot{\alpha} \\ \dot{\theta} \end{bmatrix} + \tag{11}$$
$$\begin{bmatrix} 0 \\ m_p g \frac{l}{2} \sin\theta \end{bmatrix} = \begin{bmatrix} \frac{K_m K_u}{R_m} \\ 0 \end{bmatrix} u$$

## 2.4 Linearized Equations of Motion & Linear State Space Representation

Now we have to linearize both the equations about our desired point. We choose this point to be the down right position with $\alpha = 0, \theta = 0$. We used the MATLAB **Jacobian** function to get the following linearized equations,

$$J_r \ddot{\alpha} + m_p l r \ddot{\theta} = \tau - b_r \dot{\alpha} \tag{12}$$
$$J_p \ddot{\theta} + m_p l r \ddot{\alpha} + m_p g l \alpha = -b_p \dot{\alpha}$$

We can solve for $\ddot{\theta}$ and $\ddot{\alpha}$ and get

$$\ddot{\alpha} = \frac{1}{J_t} (m_p^2 l^2 r g \theta - J_p b_r \dot{\alpha} + m_p l r b_p \dot{\theta} + J_p \tau) \tag{13}$$

$$\ddot{\theta} = \frac{1}{J_t}(-m_p g l J_r \theta - J_p b_p \dot{\theta} + m_p l r b_r \dot{\alpha} - m_p r l \tau)$$

where $J_t = J_p J_r - m_p^2 l^2 r^2$. Now we can represent Equation 13 as a linear state space model, taking the states to be $[\alpha(t)\ \theta(t)\ \dot{\alpha}(t)\ \dot{\theta}(t)]$, the outputs to be $[\alpha(t)\ \theta(t)]$ and the input to be $u(t)$ We now represent our linear state space in the form

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{14}$$

$$y(t) = Cx(t) + Du(t)$$

where,

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{k_m k_u J_p}{R_m J_t} & \frac{m_p^2 l^2 r g}{J_t} & -\frac{b_r J_p}{J_t} & \frac{m_p l r b_p}{J_t} \\ -\frac{k_m k_u m_p l r}{R_m J_t} & -\frac{m_p g l J_r}{J_t} & \frac{m_p l r b_r}{J_t} & -\frac{J_p b_p}{J_t} \end{bmatrix},\ B = \begin{bmatrix} 0 \\ 0 \\ -\frac{J_p k_m}{J_t R_m} \\ \frac{m_p r b_p k_m}{J_t R_m} \end{bmatrix},\ C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix},\ D = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$x(t) = [\alpha(t)\ \theta(t)\ \dot{\alpha}(t)\ \dot{\theta}(t)]^T,\ \ u(t) = u,\ \ y(t) = [\alpha(t)\ \theta(t)]$$

# 3 Calibration

Calibration refers to the method which is used to convert the raw sensor measurements into physical units, and check the accuracy of the sensor measurements compared to the predefined conventions discussed in Section 2.1, in order to minimise uncertainty of measurements and ensure both precision and consistency.
Regarding the Qube/Rotary Pendulum, this section explains the calibration procedure consisting of three sensor measurements: (1) Optical encoder measurement $\alpha$ (in Section 3.1); (2) Potentiometer measurement $\theta$ (in Section 3.2); (3) Motor dead-zone detection (in Section 3.3).

## 3.1 Optical encoder measurement $\alpha$

For the calibration of the optical encoder, the rotary arm is manually moved while the encoder records the signals. The extended calibration is done in [1] so here the same experiments will be performed and the results are compared.
To stay within the angular range drawn on the panel of the Qube, the arm is initialized at the middle of the panel and is given a 90 degree rotation CCW (reaching the right boundary), and then a 180 degree rotation CW (reaching the left boundary). Afterwards, the directions of movement will be inverted and the experiment will be done another three times.

| | Step 1: CCW | | Step 2: CW | |
|---|---|---|---|---|
| | $\alpha = 0°$ | $\alpha = 90°$ | $\alpha = 90°$ | $\alpha = -90°$ |
| Experiment 1 | 0 | $0.497\pi$ | $0.497\pi$ | $-0.501\pi$ |
| Experiment 2 | 0 | $0.5\pi$ | $0.5\pi$ | $-0.501\pi$ |
| Experiment 3 | 0 | $0.501\pi$ | $0.501\pi$ | $-0.501\pi$ |
| | Step 1: CW | | Step 2: CCW | |
| | $\alpha = 0°$ | $\alpha = -90°$ | $\alpha = -90°$ | $\alpha = 90°$ |
| Experiment 4 | 0 | $-0.501\pi$ | $-0.501\pi$ | $0.5\pi$ |
| Experiment 5 | 0 | $-0.5\pi$ | $-0.5\pi$ | $0.498\pi$ |
| Experiment 6 | 0 | $-0.501\pi$ | $-0.501\pi$ | $0.5\pi$ |

(15)

Table 3.1: Optical encoder calibration results

The experimental results are in agreement with the results of [1], with an average of $\pi$ rotation per 180 degree, as summarized in Table 3.1. It is worth noting that the left and right boundary both have an average of $0.001\pi$ bias (0.18 degree). Since the rotary arm should not be controlled on the boundary and such small bias on the edge could result from the manual movement, the optical encoder is taken to be calibrated successfully.

## 3.2 Potentiometer measurement $\theta$

The raw pendulum angle measurement $\theta_{raw}$ by the potentiometer does not factor in the periodicity of the rotation. Therefore, such raw measurement should be converted, according to the Equation 16, into the useful measurement $\theta$ satisfying the convention discussed in Figure 2.1.

$$\theta = \mathbf{rem}(\frac{\theta_{raw}}{2\pi}) \tag{16}$$

where **rem** is the remainder operator. For the calibration of the Potentiometer the same method is used as with the optical encoder: there are four kinds of movements to detect the bias.

- From 0° (downright) to 180°(upright) in CCW
- From 0° (downright )to 180°(upright) in CW
- From 0° to 270° in CW and back, 0° to 90° in CCW and back
- From 0° to 270° in CW, 270° to 180°(upright) in CW, 180°(upright) to 90° in CW

Every movement within a program is repeated 5 times. The results are summarised in Table 3.2:

| | Movement 1 | | Movement 2 | | Movement 3 | | | | | Movement 4 | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $\theta = 0°$ | $\theta = 180°$ | $\theta = 0°$ | $\theta = 180°$ | $\theta = 0°$ | $\theta = 270°$ | $\theta = 0°$ | $\theta = 90°$ | $\theta = 0°$ | $\theta = 0°$ | $\theta = 270°$ | $\theta = 180°$ | $\theta = 90°$ |
| Experiment 1 | 0 | $\pi$ | 0 | $\pi$ | 0 | $1.497\pi$ | 0 | $0.5\pi$ | 0 | 0 | $1.498\pi$ | $\pi$ | $0.5\pi$ |
| Experiment 2 | 0 | $0.998\pi$ | 0 | $\pi$ | 0 | $1.502\pi$ | 0 | $0.51\pi$ | 0 | 0 | $1.5\pi$ | $\pi$ | $0.51\pi$ |
| Experiment 3 | 0 | $\pi$ | 0 | $\pi$ | 0 | $1.49\pi$ | 0 | $0.48\pi$ | 0 | 0 | $1.49\pi$ | $\pi$ | $0.48\pi$ |
| Experiment 4 | 0 | $1.01\pi$ | 0 | $\pi$ | 0 | $1.503\pi$ | 0 | $0.52\pi$ | 0 | 0 | $1.503\pi$ | $\pi$ | $0.52\pi$ |
| Experiment 5 | 0 | $\pi$ | 0 | $\pi$ | 0 | $1.492\pi$ | 0 | $0.51\pi$ | 0 | 0 | $1.492\pi$ | $\pi$ | $0.51\pi$ |

(17)

Table 3.2: Servo potentiometer calibration results

After processing the data of the experiments one can see they are, for four movements, almost the same as already obtained in [1]. There are bias at $\theta = 270°$ of $0.02\pi$ radians (3.6 degree), and $\theta = 90°$ of $0.01\pi$ radians (1.8 degree). Therefore, we can conclude that there are blind zones of potentiometer alone the horizontal line where the angular accuracy is lower, and such blind zones are direction dependent.

These blind zones in the measurements will not disturb the estimation of the parameters because the excitation of the pendulum can be limited to avoid entering this blind zones. However, they are taken into consideration during the swing-up controller design since it is not possible to avoid these blind zones.

## 3.3 Motor dead-zone detection

Because of the static friction in the arm suspension and the motor, the voltage input $u$ applied to the motor has to reach a certain value before the motor will start to move. This will be called the dead-zone of the motor. These boundaries have to be known before performing experiments for the parameter estimation.

Two kind of experiments have been performed to measure the transition from static to dynamic friction (the moment when the arm will start to move) [1].

- Type (A) will increase the input step by step (0.0005 unit per step) every 2 seconds.
- Type (B) will do the same but will jump back to zero between the two steps.

The experiment is performed in positive as well as negative direction. The results of two experiments are presented in Table 3.3, and shown in Figure 3.1.

| | Positive input (CCW) | | Negative input (CW) | |
|---|---|---|---|---|
| | Type A | Type B | Type A | Type B |
| Experiment 1 | 0.0025 | 0.002 | $-0.003$ | $-0.0025$ |
| Experiment 2 | 0.0025 | 0.0025 | $-0.003$ | $-0.0025$ |

(18)

Table 3.3: Motor dead-zone calibration results

Based on Table 2.3 and Figure 3.1, the motor dead-zone is determined: the effective value for $u$ should be $> 0.0025$ for positive input and $< -0.003$ for negative input to move the rotary arm. This means there is more static friction presented and so the dead-zone will be bigger. However, the difference between the two directions is negligible so we can assume the dead zone is direction independent. Therefore, the dead-zone of the motor, especially in CW, is taken into consideration during the balance up controller design.

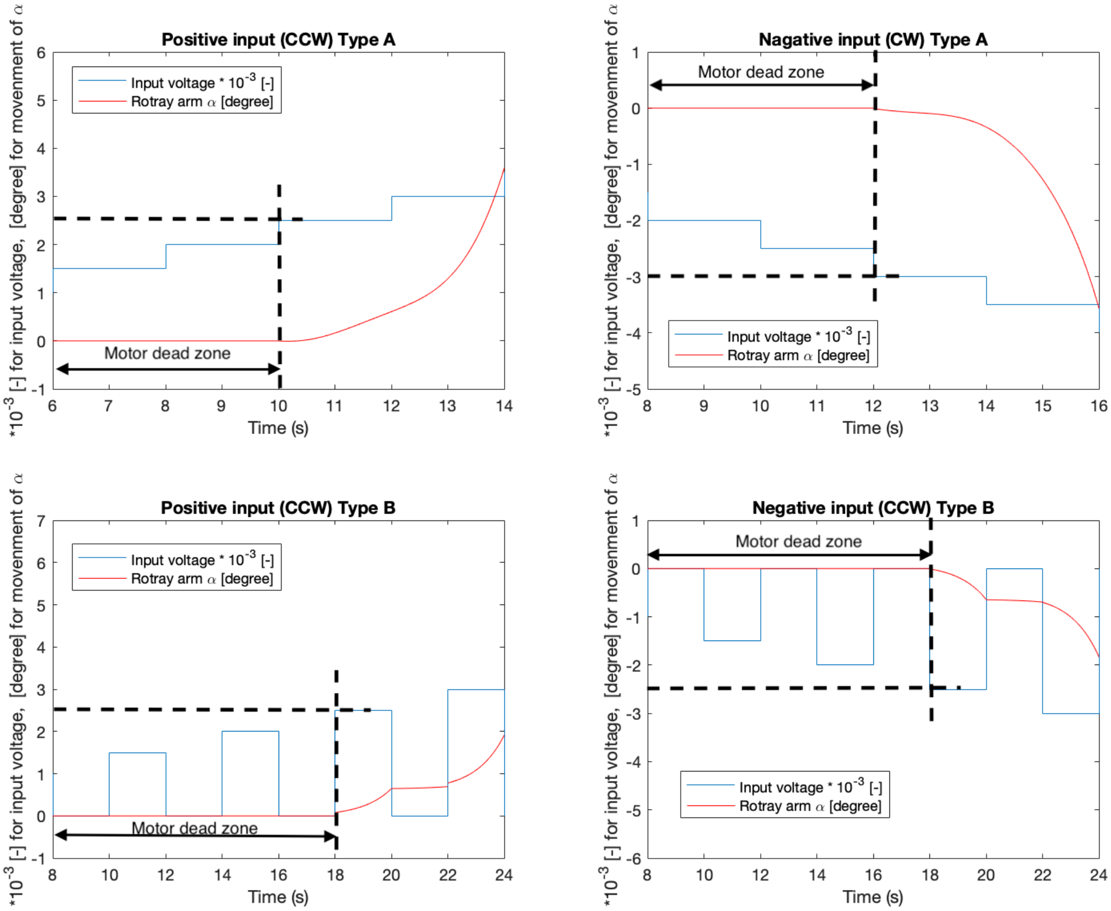

Figure 3.1: Experiments to detect the motor dead-zone

# 4 Parameter optimization

In order to design an effective controller for the linearized system, all the parameters involved in the state-space presentation must be known. However, there are several parameters that cannot be measured or calculated directly, so these unknown parameters have to be estimated and optimized. In total, there are 14 parameters involved. The estimation of the parameters can be divided in four groups:

- Parameters that can be measured directly or provided by the official documentation from Quanser;
- Parameters that can be directly calculated using default parameters;
- Parameters that can be estimated by moving the rotary arm movement while keeping the pendulum unmoved;
- Parameters that can only be estimated with an experiment involving the total setup

| **Parameter** | **Symbol** | **Estimation method** |
|---|---|---|
| **Group 1: Measured directly or defaults** | | **Numerical values** |
| Mass of pendulum | $m_p$ | 0.024 kg |
| Mass of rotary arm | $m_r$ | 0.095 kg |
| Total length of rotary arm | $r$ | 0.085 m |
| Total length of pendulum | $l$ | 0.129 m |
| Motor resistance | $R_m$ | 8.4 Ω |
| Gravity | $g$ | $9.81 m/s^2$ |
| **Group 2: Calculated using default parameters** | | **Calculation equation** |
| Length of pendulum center of mass | $\frac{l}{2}$ | = 0.0645 m |
| Inertia of rotary arm about pivot | $J_r$ | $= \frac{m_r r^2}{3} = 2.2879 * 10^{-4}$ kgm$^2$ |
| Inertia of pendulum about pivot | $J_p$ | $= \frac{m_p l^2}{3} = 1.3313 * 10^{-4}$ kgm$^2$ |
| Common denominator | $J_t$ | $= J_r J_p - m_p^2 r^2 l^2 = 1.3145 * 10^{-8}$ kgm$^2$ |
| **Group 3: Experiments only involving rotary arm** | | **Initial guess values** |
| Motor back-emf constant/PWM amplifier gain | $k_m/k_u$ | 0.042 Vs/rad |
| Motor driver gain constant | $k_g$ | 0.042 |
| Equivalent viscous damping coefficient of arm | $b_r$ | 0.003 Nms/rad |
| **Group 4: Experiments involving all setup** | | **Initial guess values** |
| Equivalent viscous damping coefficient of pendulum | $b_p$ | 0.00005 Nms/rad |

Table 4.1: Mechanical and electrical system parameters.

(19)

In Table 4.1 the four groups parameters and the estimation method are presented.

## 4.1 Estimator algorithm

### 4.1.1 System identification method candidates

The field of system identification uses statistical methods to build mathematical models of dynamical systems from measured data [2]. In control systems applications, the objective of engineers is to obtain a good performance of the closed-loop system, which is the one comprising the physical system, the feedback loop and the controller. This performance is typically achieved by designing the control law relying on a model of the system, which needs to be identified starting from experimental data.

If the model identification procedure is aimed at control purposes, what really matters is not to obtain the best possible model that fits the data, as in the classical system identification approach, but to obtain a model satisfying enough for the closed-loop performance. This more recent approach is called identification for control, or I4C in short. There are two types of system identification toolbox in Matlab being considered and compared in Table 4.2:

| **Method name** | Black-box model | Grey-box model |
|---|---|---|
| **Data dependence** | True | True |
| **Physical principles** | Unknown | Known |
| **System order** | Tunable | Specified |
| **Model linearity** | Tunable | Tunable |
| **Identifiable parameters** | Not specifiable | Specifiable |
| **Linear state definition** | Not specifiable | Specifiable |
| **Applicability between different linearization positions** | No | Yes |

Table 4.2: Comparison of System Identification Methods

(20)

### 4.1.2 Linear grey-box model description

1. Definition:
   Linear grey-box model estimation is a method that estimates and optimizes the value of user-specified parameters with a given initial guess and value range, based on: (1) the linearly -structured state-space model with identifiable parameters, which is derived from the first principles, and (2) the measurement data, and (3) the inputs to the system collected from the setup [3][4].

2. Estimation Algorithm:
   The estimation algorithm uses the prediction-error minimization algorithm to estimate/optimize the values of unknown parameters of the linear state space model. The default numerical search methods used for iterative parameter estimation consist of several line search algorithms at each iteration, such as Subspace Gauss-Newton least squares search and Steepest descent least squares search, where the first descent direction leading to a reduction in estimation cost is used [5].

3. Assessment Metrics:
   The quantitative assessment of the estimation is returned in terms of FitPercent, which calculates the Normalized Root Mean Squared Error (NRMSE) to measure of how well the response of the estimated model fits the data, expressed as the percentage

$$FitPercent = 100(1 - NRMSE) = 100(1 - \frac{\sqrt{\frac{\sum_{i=1}^{N}(y_i - \hat{y}_i)^2}{N}}}{\overline{y}}) \tag{21}$$

   where $\hat{y}$ is the estimated output, and $y$ is the collected output data, and $\overline{y}$ is the average of $y$. FitPercent varies between -Inf (bad fit) to 100 (perfect fit). If the value is equal to zero, then the model is no better at fitting the measured data than a straight line equal to the mean of the data. There are other quantitative assessment metrics listed as options, such as Mean Squared Error (MSE), and Bayesian Information Criteria (BIC), etc[6].

4. Optimization Problem:
   The objective function is minimizing the NRMSE by optimizing the parameter values contained in state space representation. The high non-convexity of such optimization problem comes from the high order multiplication of matrix A in the expression for output $\hat{y}$.

$$\begin{aligned} x_{k+1} &= Ax_k + Bu_k \\ \hat{y}_k &= Cx_k + Du_k \\ \hat{y}_{k+1} &= Cx_{k+1} + Du_{k+1} = C\left(Ax_k + Bu_k\right) + Du_{k+1} = CAx_k + CBu_k + Du_{k+1} \\ \hat{y}_{k+in-1} &= CA^{n-1}x_k + CA^{n-2}Bu_k + CA^{n-3}Bu_{k+1} + \cdots + CBu_{k+(n-2)} + Du_{k+(n-1)} \end{aligned} \tag{22}$$

### 4.1.3 Argument of the choice of identification method: linear grey-box model

The reasons why the linear grey-box model estimation is preferred in this report are:

- The physical properties of the set up can be identified accurately in the form of the first principles. The complexity of the physical modeling of this set up is simple with the help of the reference material. Therefore, the grey-box model is preferred because its modeling can make good use of these priors.

- Black-box model is infamous for its over-fitting because the system behavior may change under different input signals. It is difficult to estimate a model that can obtain high validation rate for any types of the input signals using the black-box model. However, the over-fitting problem can be mitigated in the grey-box model by applying the first principles, or close-to-real initial guess for unknown parameters with the help of several references, or decrease the non-convexity of the optimization problem by decreasing the number of parameters needed to be estimated.

- The linearization of the set up at downright position and the upright position is highly relevant, which can be verified by the small difference between the state-space representations of the linearized system at two positions. Compared to black-box estimation that is not able to use such relevance because its model is not based on the first principles, the grey box estimation is more flexible to take good use of these physics priors.

- There are only 4 out of 14 parameters needed to be estimated, which are listed in Table 3.1. The grey-box model can estimate these parameters with the "contribution" of the 11 known parameters. However, such advantage of the already known parameters is not applicable for black-box model.

- There is also a non-linear grey-box option to choose from. However, we go ahead with the linear grey-box tool since we are working with a linear state space model, where we apply concepts that can only be used for linear systems.

## 4.2 Grey-box estimation experiment for parameters in Group 3

The first experiment considers only the movement of rotary arm and keep the pendulum at the downright position, by choosing a proper input. For this experiment, the angular value of pendulum $\theta$, its derivative $\dot{\theta}$, and its double derivative $\ddot{\theta}$ are assumed to be zero. Under such assumption, the linear EOM around the downright position for the setup in Equation 13 are simplified as below:

$$\ddot{\alpha} = \frac{1}{J_t}\left(m_p^2 l^2 r g\theta - J_p b_r \dot{\alpha} + m_p l r b_p \dot{\theta} + J_p \tau\right) \xrightarrow{\theta=0,\ \theta\dot{=}0} \ddot{\alpha} = \frac{1}{J_t}\left(-J_p b_r \dot{\alpha} + J_p \tau\right) \tag{23}$$

$$\ddot{\theta} = \frac{1}{J_t}\left(-m_p g l J_r \theta + m_p l r b_r \dot{\alpha} - J_p b_p \dot{\theta} - m_p r l \tau\right) \xrightarrow{\theta=0,\ \dot{\theta}=0} \ddot{\theta} = \frac{1}{J_t}\left(m_p l r b_r \dot{\alpha} - m_p r l \tau\right) = 0 \tag{24}$$

where

$$J_t = J_p J_r - m_p^2 l^2 r^2$$

By combining the two equations, the new system matrix $A_{new}$ in the state-space representation is simplified as below

$$A_{new} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{k_m k_u J_p}{R_m J_t} & 0 & -\frac{b_r J_p}{J_t} & 0 \\ -\frac{k_m k_m m_p l r}{R_m J_t} & 0 & \frac{m_p l r b_r}{J t} & 0 \end{bmatrix} \tag{25}$$

The two unknown parameters $b_r$, $k_m$ in $A_{new}$, and the parameter $k_u$ in the system matrix B in Equation 14 can be estimated using the grey-box parameter estimation method. The input signal $u_1(t)$ is a combination of different styles of signal at each time period, such as square, sine wave signals, that can rotate the arm and keep the pendulum at the downright position. In order to decrease the over-fitting of the estimated model to the input signal, a different combination of input signal $u_2(t)$ is chosen to validate the estimated model.

The procedure of this experiment is:

- First, the measurement data $y(t) = \left[ \begin{array}{cc} \alpha(t) & \theta(t) \end{array} \right]^T$ and input signal $u_1(t)$ are collected;
- Secondly, the linear grey-box model are built in the form of the new state-space representations, with three unknown parameters as objective parameters;
- Thirdly, the prediction rates of the estimated system with different combination of initial guess and value range for the three objective parameters, are collected and repeat this step till obtaining the highest prediction rate.
- Finally, the validation rate of the above final estimated system to random input signal $u_2(t)$ is obtained. The last two steps are repeated till both the prediction rate $P_r$ and the validation rate $V_r$ are high enough.

The experiment records are summarized in the Table 4.3 below, where one can notice that the initial values play an important role for searching the global optimum. There is most likely an optimal hyper-plane in the optimization problem, which means that different value combination of estimated variables may yield the same performance when some terms in the the state space expression consist of these variables. Therefore, under same input data, different initial guess of estimated variables may return different optimal values.

Different input signals are applied to prevent the over-fitting problem. The prediction input signals $s_{p1}$ and $s_{p2}$ are combination of square and sine signals with different amplitude and frequency starting at the Clockwise direction. The validation input signal $s_v$ is the random signal with similar amplitude and frequency. The value range of three parameters are $b_r \in [0.00001, 0.01]$, $k_m \in [0.001, 2]$, $k_g \in [1, 100]$. In experiment 4, both validation rates $P_r$ and $V_r$ are the highest compared to all the other included and dis-included experiments, and the optimal values of three parameters are $b_r^* = 0.0027$, $k_m^* = 0.857$, and $k_u^* = 35.243$.

| | Input signal | | Initial values | | | Optimal values | | | $P_r$ | | $V_r$ | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | $u_1(t)$ | $u_2(t)$ | $b_r$ | $k_m$ | $k_g$ | $b_r^*$ | $k_m^*$ | $k_g^*$ | $\alpha$ | $\theta$ | $\alpha$ | $\theta$ |
| Experiment 1 | $s_{p1}$ | $s_v$ | 0.01 | 2 | 100 | 0.00143 | 0.03 | 100 | 12.23% | 23.44% | 22.34% | 34.253% |
| Experiment 2 | $s_{p1}$ | $s_v$ | 0.003 | 0.84 | 30 | 0.0035 | 0.871 | 30.234 | 47.45% | 62.45% | 45.245% | 56.234% |
| Experiment 3 | $s_{p2}$ | $s_v$ | 0.003 | 0.84 | 30 | 0.0033 | 0.867 | 34.459 | 57.324% | 68.234% | 42.46% | 53.876% |
| Experiment 4 | $-s_{p2}$ | $s_v$ | 0.003 | 0.84 | 30 | 0.0027 | 0.857 | 35.243 | 59.213% | 68.657% | 46.234% | 59.324% |

Table 4.3: Experiment records for estimating the parameters in Group 3

(26)

In Figure 4.1 and 4.2, the prediction and validation performances of the system identification are compared among the setup measurement (with the legend as "Collected"), grey box system with initial guessed parameters ("Initial"), with optimal parameters estimated in Experiment 1 ("exp1") and Experiment 2("exp2"). The $\theta$ angular movement stays within the range of [-0.74°, 1.14°], which is almost similar to the assumed motionless state, proving that the input signals are designed properly.

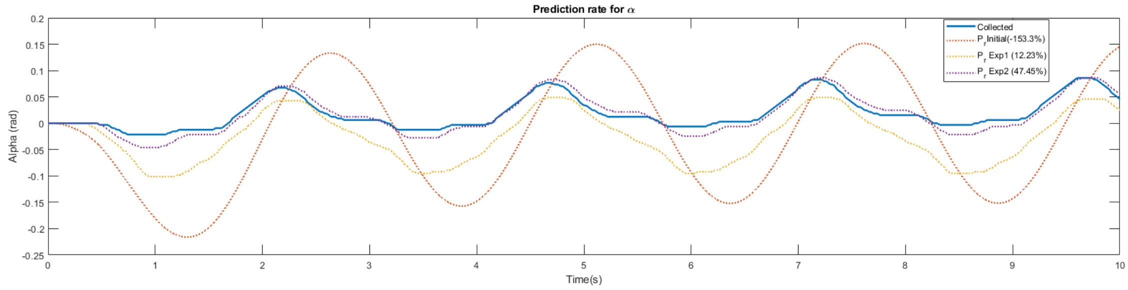

Figure 4.1: Prediction rate for the same input $u_1(t) = s_{p1}$ of parameters estimated in Group 3

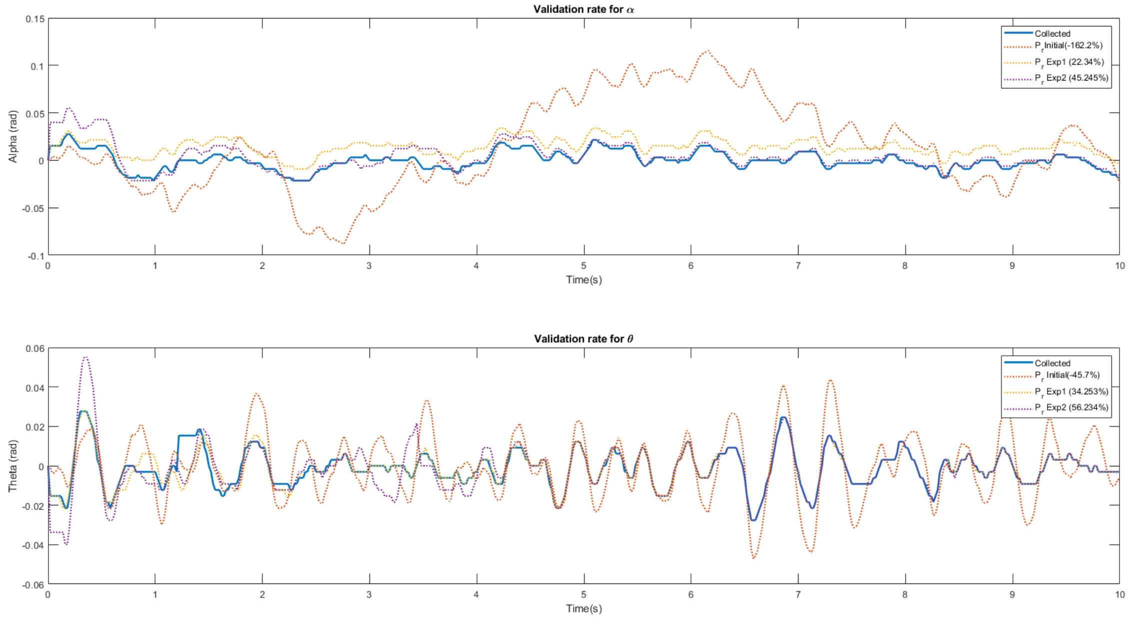

Figure 4.2: Validation rate for the random input $u_2(t) = s_v$ of parameters estimated in Group 3

## 4.3 Grey-box estimation experiment for parameters in Group 4

After estimating the three parameters in Group 3 with both acceptably high enough prediction $P_r$ and validation $V_r$ rate, the last parameter $b_p$ can be estimated in the experiment involving rotating both the rotary arm and pendulum. The experiment procedure is same as discussed above. The state-space model in Equation 14 is used as the grey-box model.

In this experiment, the square input signals with different combinations of amplitude and frequency are used for grey-box estimation (prediction), and random signals with same amplitude are used for system validation. The reason to use random signals to validate the estimated system is that the input signal for balancing up the pendulum at upright (or downright) position is similar to random signal that is not a simple combination of sine or square signal. If the validation rate is high enough for the random signals, then the parameters are estimated precisely.

| | Input signal | | Initial values | Optimal values | Fixed values | $P_r$ | | $V_r$ | |
|---|---|---|---|---|---|---|---|---|---|
| | $u_1(t)$ | $u_2(t)$ | $b_p$ | $b_p^*$ | $b_r^*$ | $\alpha$ | $\theta$ | $\alpha$ | $\theta$ |
| Experiment 1 | $s_{p3}$ | $s_v$ | 0.0001 | 0.000245 | 0.0027 | −20.3% | 12.453% | −45.234% | 16.233% |
| Experiment 2 | $s_{p3}$ | $s_v$ | 0.00005 | 0.0000542 | 0.0027 | 67.78% | 85.23% | 65.324% | 78.234% |
| Experiment 3 | $s_{p4}$ | $s_v$ | 0.000005 | 0.0000523 | 0.0027 | 73.22% | 87.234% | 69.23% | 81.03% |
| Experiment 4 | $-s_{p4}$ | $s_v$ | 0.000005 | 0.000437 | 0.0027 | 71.23% | 86.213% | 67.213% | 80.234% |
| Experiment 5 | $-s_{p4}$ | $s_v$ | 0.000005 | 0.0000334 | 0.0038 | 33.324% | 45.324% | 23.425% | 40.324% |

Table 4.4: Experiment records for estimating the parameters in Group 4

(27)

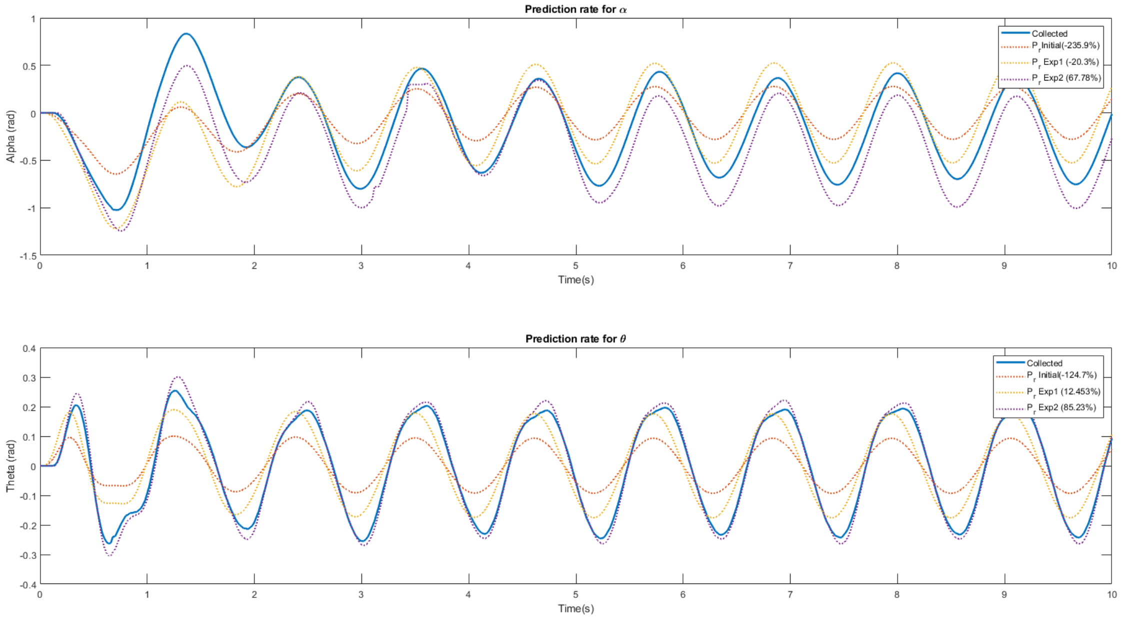

Figure 4.3: Prediction rate for the input $u_1(t) = s_{p3}$ of parameter estimated in Group 4

In Table 4.4 and Figure 4.3 and 4.4, some representative experiment records are demonstrated. The input signal $s_{p3}$, $s_{p4}$ and $s_v$ are designed in the same way as in Table 3.3. Based on the records, there are several problems being recognised, and corresponding potential reasons are given below:

1. **Problem 1**: Both the prediction rate $P_r$ and validation rate $V_r$ for $\alpha$ (the rotary arm movement angle) is averagely lower than the ones for $\theta$ (the pendulum rotation angle).
   **Potential reasons**: The rotary arm is activated by the motor whose mechanical system is modelled as rotational friction damper at first, which dissipates seismic energy by friction generated through a rotating friction joint. The influence of the torque applied to the rotary arm is larger than to the pendulum, which can be validated in Equation ?. Therefore, the low accuracy of the modelling of the motor may lower the prediction (or validation) rate for $\alpha$ in a stronger degree than for $\theta$.

2. **Problem 2**: The prediction rate and validation rate are sensitive to the input signals such that both rates are higher when the input signal has relatively lower amplitude or simpler pattern.
   **Potential reasons**: The optimum value obtained may not be the true parameter value of the setup because the optimum obtained from greyest is based on the linearized model at the downright position, and in reality the true model is nonlinear and most likely there are dynamical aspects which cannot be taken into account. Such ignored dynamical aspects may influence the

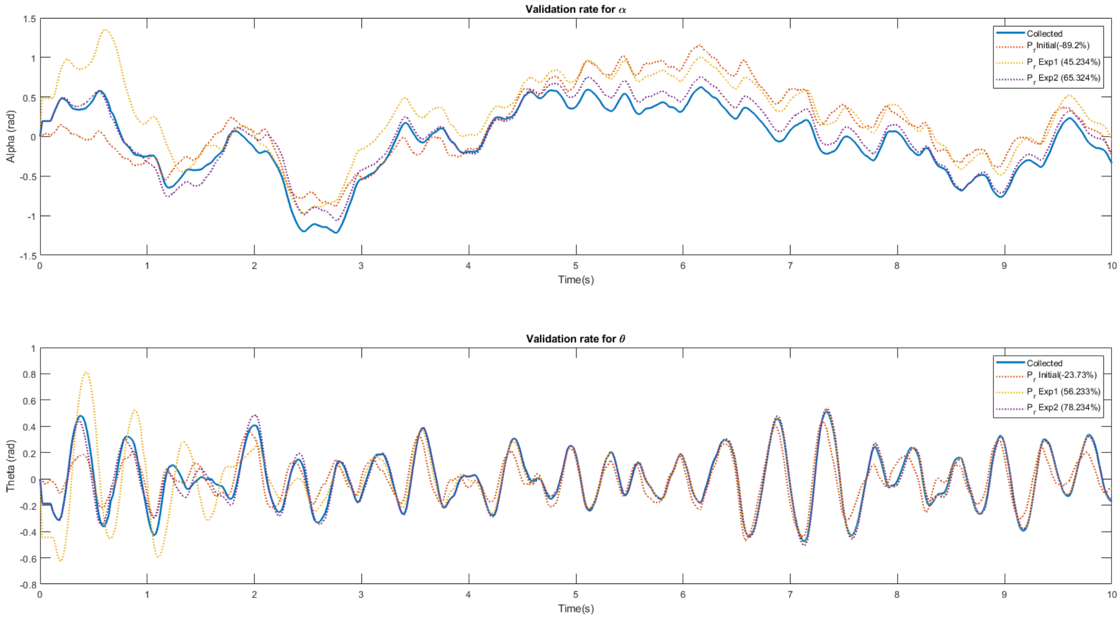

Figure 4.4: Validation rate for the random input $u_2(t) = s_v$ of parameter estimated in Group 4

accuracy of the estimated model when the state is far away from the linearied (downright) position, or when the input signal has larger amplitude or complex pattern.

3. **Problem 3** :When the motor is activated at the beginning, the prediction rate is much lower than the period when the rotary arm is swinging under a pattern.
   **Potential reasons**: The activation procedure of the motor when receiving a step signal during the first second is highly nonlinear that it is difficult to model the torque provided by the motor from the static condition using first principles. However, after the activation, the torque applied from the motor follows the electronic knowledge discussed in Equation ?.

4. **Problem 4**: The improvement of the prediction (or validation) rate for $\alpha$ sacrifices the one for $\theta$.
   **Potential reasons**: In this experiment, there is only one parameter being estimated. The three parameters in Group 3 estimated only involve the movement of rotary arm may not be accurate for this experiment involving the whole set up because the state-space models are different in two experiments. However, estimating four parameters at the same time renders the optimization problem become highly non-convex that it is difficult to find the true values for all the parameters at the same iteration.

## 4.4 Parameter estimation summary

### 4.4.1 Direction-dependence of estimated parameters

Based on the results of the Experiment 3&4 in Table 4.3 and Experiment 3&4 in Table 4.4, it can be noticed that the system responses with same input signal in different directions are not the same. In Table 4.5, the difference of the estimated parameters for CW and CCW experiment are summarized, where the big difference indicates that the viscous friction between clockwise experiments and counterclockwise experiments may not be the same. This further hints us that the friction present in the setup is not symmetric anymore and this phenomenon is difficult to explain but will be useful while designing the controller.

|  | $\mathbf{b_r}[Nms/rad]$ | $\mathbf{b_p}[Nms/rad]$ | $\mathbf{k_m}[Vs/rad]$ |
|---|---|---|---|
| Clockwise<br>(C) | 0.0033<br>±0.13 | 0.000052<br>±0.023 | 0.867<br>±0.004 |
| Counterclockwise<br>(CC) | 0.0027<br>±0.0009 | 0.000043<br>±0.000017 | 0.857<br>±0.012 |

(28)

Table 4.5: Difference friction for C- and CC-direction

### 4.4.2 Connection between the optimal value of $b_r$ and $b_p$

Based on Experiment 4&5 in Table 4.4, the optimal value of $b_r$ and $b_p$ estimated from the grey box optimization seems to have a kind of connection. Specifically, if the optimal value of $b_r$ increases, the optimal value of $b_p$ will decrease and vice versa. This is probably caused because they both depend on $\ddot{\theta}$ in the linearization relation in Equation 13. This fact, possible small model errors and the time limit will be the reason the grey box model estimation does not yield the global optimum. It is difficult to estimate the dynamics as well as the parameters within a short time where the parameters depend on the same state ($\ddot{\theta}$). It could work if the time length of the experiments is increased to a much more longer period (in the order of minutes). The grey box model will have enough data to distinguish the contribution of $b_r$ and $b_p$ to the dynamics separately (and so, not depending on each other). For now the results presented above will be used to design a controller.

### 4.4.3 Estimated parameter values summary

All the parameter values are collected in the Table 4.6 below. The numerical values of the four estimated parameters are almost identical to the result in [2]. Readers are recommended to identify their setup with these values and adjust according to the setup performance.

| **Description** | **Symbol** | **Value** |
|---|---|---|
| **Rotary arm** | | |
| Mass of rotary arm | $m_r$ | 0.095 kg |
| Total length of rotary arm | $r$ | 0.085 m |
| Inertia of rotary arm | $J_r$ | $2.2879 * 10^{-4}$ kgm$^2$ |
| Equivalent viscous damping coefficient of arm | $b_r$ | 0.0027 Nms/rad |
| Gravity constant | $g$ | 9.81 |
| **Pendulum link** | | |
| Mass of pendulum | $m_p$ | 0.024 kg |
| Total length of pendulum | $L_p$ | 0.129 m |
| Length of pendulum center of mass | $l$ | 0.0645 m |
| Inertia of pendulum | $J_p$ | $1.3313 * 10^{-4}$ kgm$^2$ |
| Equivalent viscous damping coefficient of pendulum | $b_p$ | 0.000043 Nms/rad |
| **DC Motor** | | |
| Motor resistance | $R_m$ | 8.4 Ω |
| Motor back-emf constant | $k_m$ | 0.84 Vs/rad |
| Motor driver gain constant | $k_u$ | 40 |

Table 4.6: Mechanical and electrical system parameters.

(29)

# 5 Control of the pendulum

## 5.1 Upright position linearization

The resultant modelling state space matrices that we saw in Chapter 2, were the result of the Jacobian taken at the downright equilibrium position of $\alpha = 0, \theta = 0$. However, we now require a linear model of the system at the upright position. The modelling equation fortunately does not change much for this case. The reason for this is that at the downright position where $\theta = 0$, $cos\theta = 1$ and $sin\theta = 0$. At the upright position, $\theta = \pi$, and $cos\theta = -1$, $sin\theta = 0$. The only terms in our equations of motion and the Jacobian are the sine and cosine of the $\theta$, so any changes to the system model are effectively just inverting the signs of the terms which have $sin\theta$ and $cos\theta$ terms. We used the MATLAB **Jacobian** function to get the following linearized equations,

$$J_r\ddot{\alpha} - m_p l r\ddot{\theta} = \tau - b_r\dot{\alpha} \tag{30}$$

$$J_p\ddot{\theta} - m_p l r\ddot{\alpha} + m_p g l\alpha = -b_p\dot{\alpha}$$

We can solve for $\ddot{\theta}$ and $\ddot{\alpha}$ and get

$$\ddot{\alpha} = \frac{1}{J_t}(m_p^2 l^2 r g\theta - J_p b_r\dot{\alpha} - m_p l r b_p\dot{\theta} + J_p\tau) \tag{31}$$

$$\ddot{\theta} = \frac{1}{J_t}(m_p g l J_r\theta - J_p b_p\dot{\theta} - m_p l r b_r\dot{\alpha} + m_p r l\tau)$$

where $J_t = J_pJ_r - m_p^2 l^2 r^2$. Now we can represent Equation 31 as a linear state space model, taking the states to be $[\alpha(t)\ \theta(t)\ \dot{\alpha}(t)\ \dot{\theta}(t)]$, the outputs to be $[\alpha(t)\ \theta(t)]$ and the input to be $u(t)$ We now represent our linear state space in the form

$$\dot{x}(t) = Ax(t) + Bu(t) \tag{32}$$

$$y(t) = Cx(t) + Du(t)$$

where,

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ -\frac{k_m k_u J_p}{R_m J_t} & \frac{m_p^2 l^2 r g}{J_t} & -\frac{b_r J_p}{J_t} & -\frac{m_p l r b_p}{J_t} \\ -\frac{k_m k_u m_p l r}{R_m J_t} & \frac{m_p g l J_r}{J_t} & -\frac{m_p l r b_r}{J_t} & -\frac{J_p b_p}{J_t} \end{bmatrix},\ B = \begin{bmatrix} 0 \\ 0 \\ \frac{J_p k_m}{J_t R_m} \\ -\frac{m_p r b_p k_m}{J_t R_m} \end{bmatrix},\ C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}\ D = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

$$x(t) = [\alpha(t)\ \theta(t)\ \dot{\alpha}(t)\ \dot{\theta}(t)]^T,\ \ u(t) = u,\ \ y(t) = [\alpha(t)\ \theta(t)]$$

Note that the controllability and obserbility matrix both are full rank.

## 5.2 State observer

With the observablility and controllability properties of the linear system at upright position in Equation 32 for any $T > 0$, it is possible to uniquely determine/reconstruct $x(t)$ for $t \in [0, T]$ based on knowledge of the input $u(t)$ and output $y(t)$ for $t \in [0, T]$, and further implement an observer-based output-feedback controller with the reconstructed state. The following methods have been tested to build the state observer. The terms ”observer”, ”estimator”, ”filter” are in this context used synonymously.

### 5.2.1 High-pass filters

We need a state observer since we directly measure only 2 of our 4 states($\alpha$ and $\theta$). In order to make a well-informed controller gain design, we need full state information. For this purpose, you can use an observer, a high pass filter, or a filter. They all serve to perform the same function. We use a high-pass filter to estimate the other 2 states($\dot{\alpha}$ and $\dot{\theta}$) [7]. The Quanser documentation recommends the use of the following high pass filter for both $\dot{\alpha}$ and $\dot{\theta}$ [8]:

$$\hat{\dot{\alpha}} = \frac{50s}{s+50}\alpha, \quad \hat{\dot{\theta}} = \frac{50s}{s+50}\theta \tag{33}$$

where $\hat{\dot{\alpha}}$ and $\hat{\dot{\theta}}$ are estimates of $\dot{\alpha}$ and $\dot{\theta}$ respectively. What this mathematically signifies is that we are differentiating the outputs w.r.t time, imposing a cut-off frequency on the result, and amplifying the result with a gain.

However, we experimentally found that the cut-off frequency is too low for our application and that a lot of unnecessary lower frequency signals were appearing in our estimated signals. A higher frequency cutoff leads to elimination of the lower frequency noise spectrum. In some applications, the important information in the signal is low-frequency, meaning that averaging recent measurements may be more accurate than just taking the most recent one. This is smoothing. Averaging recent measurements is a simple smoothing technique. In other applications, such as this one, the important information is high-frequency. A car airbag deploys at a sudden deceleration, what matters is the difference between recent speed and current speed, not the level of either one. A low-pass filter, or smoothing, would destroy the value of the airbag. A high pass filter could make it more sensitive and less susceptible to error.

Similarly in our case, we care about the difference between current position and recent position, and not either of the levels. Another important point to note is that both the parameters need not have exactly the same filter. $\dot{\alpha}$ and $\dot{\theta}$ are velocities, and are coupled with each other, but have different dynamics and even act along different planes, so they generally shouldn't have the same cut-off frequency. As for the filter gains, their value represents how much the estimated signal needs to be amplified to be a realistic estimation of the parameter. We proportionally vary the gain along with the cut-off frequency.

Figure 5.1 shows the comparison between the bode plots of the recommended high-pass filter and the filter that we are using. The difference in the amplitude of the filtered signal and the cut-off frequency is clearly visible. It should be noted that the system is very sensitive to a change in the HPF transfer function. The ideal value was tuned experimentally. The legend shows the different high pass filters that were plotted against each other. The official document recommendation was to use the HPF $50s/(s+50)$. However, on using this to estimate $\dot{\alpha}$ and $\dot{\theta}$, it was found that while the estimate of $\dot{\theta}$ was decent, the one for $\dot{\alpha}$ had a big offset. It was found that by varying the filter gain and cut-off frequency, a major effect on the estimation, and thus this offset was observed. It could be that lower frequency noise was contributing to the offset in our estimation. As we started increasing the cut-off frequencies, and thus increasing the frequency ranges to get filtered off, we got better readings of $\dot{\alpha}$, but $\dot{\theta}$ estimation deteriorated due to their coupling. So we had to vary both the filters in order to get a good estimation. Experimentally the filters that worked as the best estimators for $\dot{\alpha}$ and $\dot{\theta}$ were $\frac{140s}{s+140}$ and $\frac{180s}{s+180}$. We found these two estimations to be very sensitive to any changes, leading us to the conclusion that the frequency noises in this range cause the most trouble to the state estimation.

### 5.2.2 Luenberger observer

For the continuous-time state-space system at the upright position, the observed state $\hat{x}$ and observed output $\hat{y}$ are obtained by:

$$\begin{aligned}\dot{\hat{x}} &= A\hat{x} + Bu + L(y - \hat{y}) \\ \hat{y} &= C\hat{x} + Du\end{aligned}$$

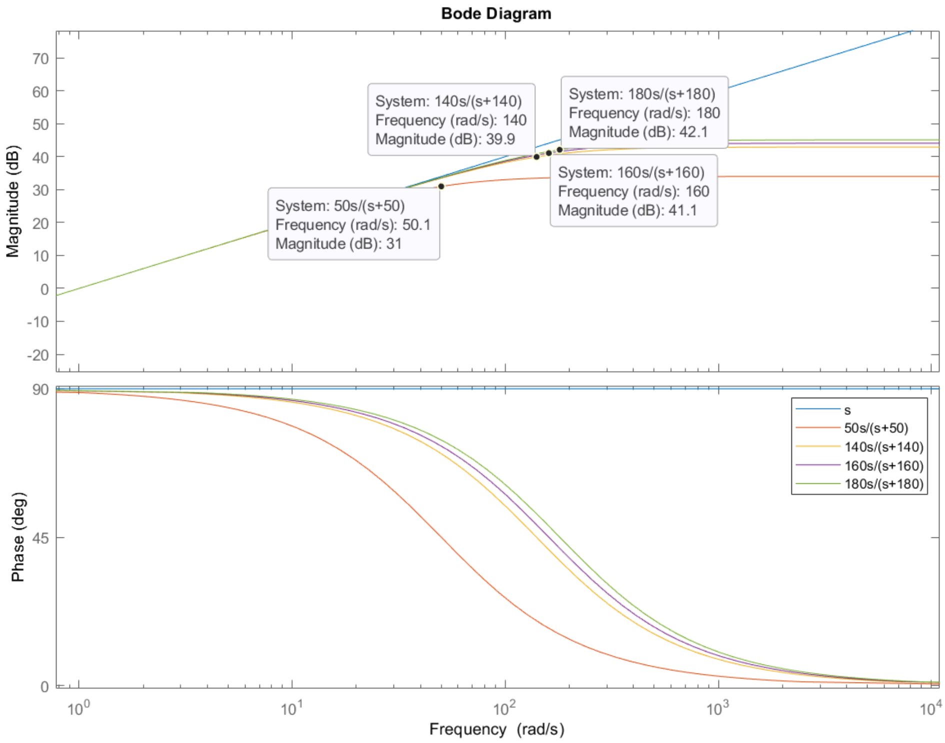

Figure 5.1: Frequency domain analysis of HPF cut-off frequencies

The observer error $e = x - \hat{x}$ satisfies the equation

$$\dot{e} = (A - LC)e$$

The observer is asymptotically stable if the observer error $e(k) = \hat{x}(k) - x(k)$ converges to zero when $k \to \infty$. For a Luenberger observer, the observer error satisfies $e(k+1) = (A - LC)e(k)$. The Luenberger observer for the continuous-time system is therefore asymptotically stable when the matrix $A - LC$ has all negative eigenvalues.

Due to the separation principle, the feedback gain $K$ and observer gain $L$ can be decided independently without harm to the overall stability of the systems [9]. As a rule of thumb, the poles of the observer $A - LC$ are usually chosen to converge 5 times faster than the poles of the system $A - BK$, which means we wish to use "fast" eigenvalues for the error dynamics $A - LC$. However, they should not be taken too fast since measurement noise might then be amplified. Moreover large observer gains can adversely influence robustness.

By replacing the unavailable $x$ by the available $\hat{x}$ for control, the observer-based output-feedback controller can be built and the control input is:

$$u(k) = -K\hat{x}(k) + k_r * r$$

where $k_r$ is the steady state gain (the inverse of the dc-gain of the closed-loop feedback system), and $r$ is the reference signal. In order to calculate the observer error, an extended state $\bar{x}(k) = \begin{bmatrix} x(k) \\ \hat{x}(k) \end{bmatrix}$ in the following Equation is required to use the Matlab function **lsim**:

$$\begin{aligned} \begin{bmatrix} x(k+1) \\ \hat{x}(k+1) \end{bmatrix} &= \begin{bmatrix} A & -BK \\ LC & A - LC - BK \end{bmatrix} \begin{bmatrix} x(k) \\ \hat{x}(k) \end{bmatrix} + \begin{bmatrix} k_r \\ k_r \end{bmatrix} r \\ \begin{bmatrix} y(k) \\ \hat{y}(k) \end{bmatrix} &= \begin{bmatrix} C & \mathbf{0} \\ \mathbf{0} & C \end{bmatrix} \begin{bmatrix} x(k) \\ \hat{x}(k) \end{bmatrix} \\ \bar{x}(0) &= \begin{bmatrix} x(0) \\ \hat{x}(0) \end{bmatrix} \end{aligned} \tag{34}$$

where $x(0) = \begin{bmatrix} 0 \\ \pi \\ 0 \\ 0 \end{bmatrix}$ (the upright position), and $\hat{x}(0)$ is chosen randomly.

In Figure 5.2 (left), the rotary arm consumes less time to be balanced back to the downright original position with the state feedback controller and the Luenberger observer, compared to the case with the high-pass filters. In Figure 5.2 (right), the state observer error $e$ between $x(t)$ (estimated with the high-pass filters) and $\hat{x}(t)$ (estimated with the luenberger observer) with different initial position $\hat{x}(0)$ are shown. One can notice that the four states observer errors converge to zero after 0.3 seconds.

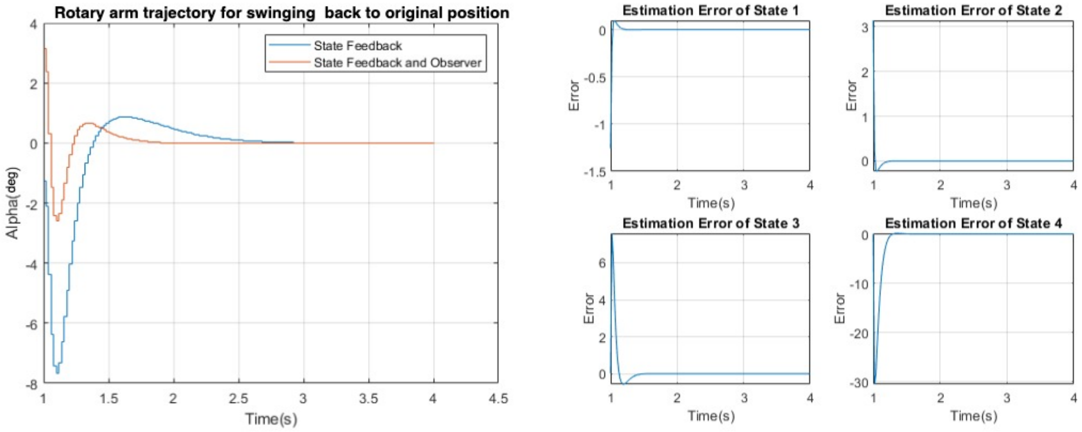

Figure 5.2: Luenberger observer reference tracking $\alpha$ trajectory (left) and state error (right)

### 5.2.3 Kalman filter

During the experiments, it is noticed that the velocity state values observed from the Luenberger observer are slightly noisy, and thus a time-invariant Kalman filter observer is used. The Kalman filter uses the system model and input and output measurements to solve the Kalman gain K by minimizing the error covariance P, where the Kalman filter algorithm is summarized in Equation 35 (for prediction) and 36 (for update) and the detailed derivation of the algorithm and all the symbol meanings are explained in [10]:

$$\begin{aligned} \hat{x}_k^- &= A\hat{x}_{k-1} + Bu_k \\ P_k^- &= AP_{k-1}A^T + Q \end{aligned} \tag{35}$$

$$\begin{aligned} &K_k = \frac{P_k^- C^T}{CP_k^- C^T + R} \\ &\hat{x}_k = \hat{x}_k^- + K_k\left(y_k - C\hat{x}_k^-\right) \\ &P_k = \left(I - K_k C\right) P_k^- = AP_{k-1}A^\top + Q - K_k\left(CP_{k-1}C^\top + R\right)K_k^\top \end{aligned} \tag{36}$$

where the first two terms for $P_k$ on the right represent the natural evolution of the uncertainty, the last term shows how much uncertainty the Kalman filter removes.

The process noise covariance matrix Q has the same dimension as the A matrix, and the numerical value of $diag([0, 0, q_1, q_2])$ is applied with the assumption that the process noise effects the angular velocity only. The measurement noise covariance matrix R is assumed to be time-invariant as well. The Kalman filter gives an unbiased estimate, i.e.,

$$\mathrm{E}[\hat{x}(k \mid k)] = \mathrm{E}[\hat{x}(k \mid k-1)] = \mathrm{E}[x(k)]$$

If the noise is uncorrelated with $x(0)$, then the Kalman filter is optimal, i.e., no other linear filter gives a smaller variance of the estimation error.(For non-Gaussian assumptions, nonlinear filters, particle filters or moving-horizon estimators do a much better job.) By applying the estimated state from

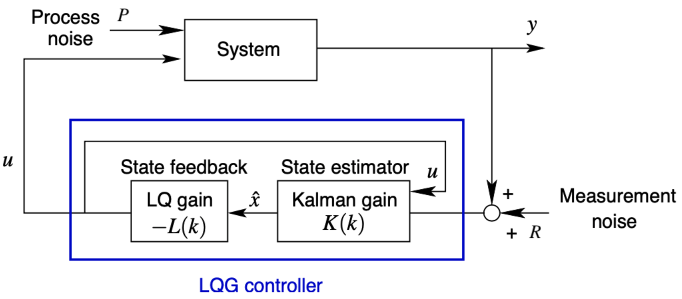

Figure 5.3: Linear quadratic Gaussian control diagram

Kalman filter to the LQR controller, the Linear Quadratic Gaussian controller is for optimal control of linear systems with additive Gaussian noise using quadratic state and control costs, as shown in Figure 5.3. As such, LQG controllers can be used for systems which explicitly model measurement noise in the output, while LQR control cannot.

### 5.2.4 Observer performance comparison and discussion

In order to choose the best state reconstruction method that works well with the balance-up controller to be discussed in the next section, the reconstructed states $\dot{\alpha}$ and $\dot{\theta}$ from four methods with sine wave $\alpha$ reference signal (amplitude=±15°, time period=10s): derivative of angular measurement, high-pass filters, Luenberger observer and Kalman filter, are compared and shown in Figure 5.4. Different than the simple control requirement performed in Figure 5.2, the Luenberger observer and Kalman filter both return unrealistic values and thus fail to balance up, compared to the case with high-pass filter that succeed to balance up. One can observe that the Kalman filter estimated state has a sine wave pattern, which could result from the sine wave reference signal. Compared to derivative and high-pass filter, the Luenberger observer and Kalman filter always return positive velocity values. There are several potential reasons discussed below

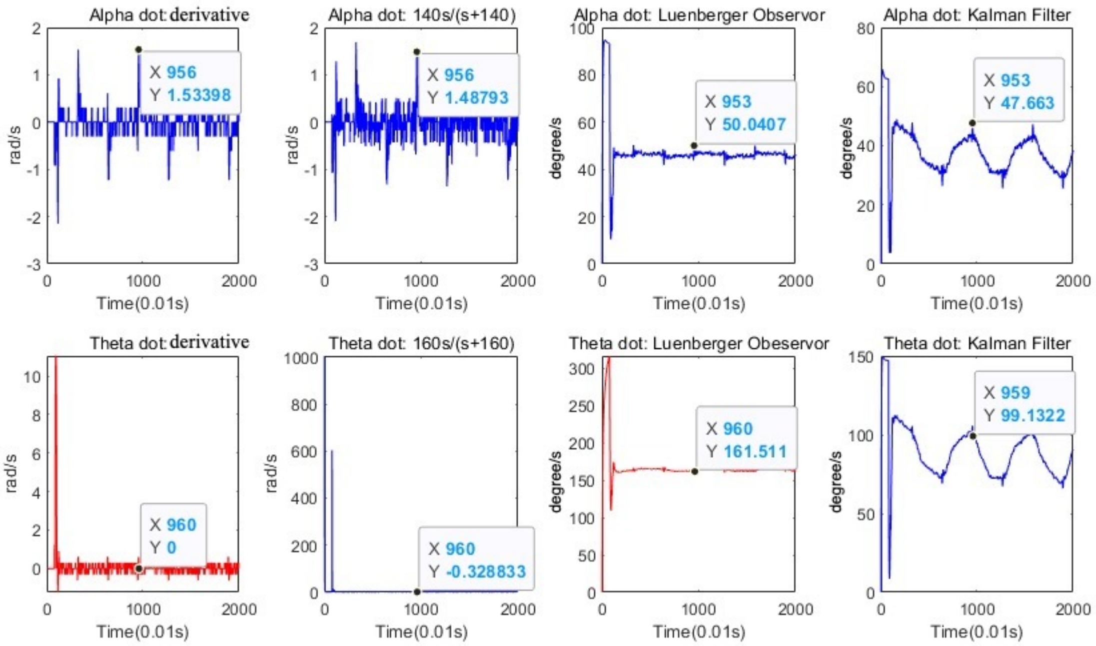

Figure 5.4: Time domain state estimation comparison of HPF (rad/s), Luenberger Observer (degree/s) and Kalman Filter (degree/s)

- The linearized state-space representation at upright position may not be partially incorrect that the two observers are not accurately estimating $\dot{\alpha}$ and $\dot{\theta}$. However, the downright state-space representation is identified correctly in the previous section, thus the Luenberger returns correct velocity states value similar to high-pass filter as shown in Figure 5.2 (right). Therefore, a closed-loop identification should be performed, and will be discussed in the last section.

- The parameters for the Luenberger observer (observer gain $L$) and Kalman filter (process noise covariance $Q$ and measurement noise covariance $R$) are not tuned correctly.

## 5.3 Balancing controllers

Till now, we have studied the model of the system, parameter estimation, optimisation and state estimation. The ultimate purpose of what we have done is to replicate the system as effectively as possible, so that we can design controller gains good enough to satisfy our objective. In this section, our objective is to design a controller such that our inverted pendulum can balance in the upright position, and our rotary arm can track a reference angle simultaneously.

### 5.3.1 Pole placement controller

We want to track our reference with minimum overshoot, settling time, rise time and offset. We can directly correlate these parameters with the damping ratio and natural frequency of a second order system.
The relation between second order system parameters and control system characterisitics are [11]:

$$t_s = \frac{4}{\zeta\omega_d},\ \%Overshoot = exp(\frac{-\zeta\pi}{\sqrt{(1-\zeta^2)}}) * 100\%,\ t_r = \frac{\pi}{2\omega_d}$$

where $t_s$ is the 2% settling time, $\zeta$ is the damping ratio, $\omega_d$ is the damped frequency, $t_r$ is the 100% rise time. We choose our requirements as $t_s = 2s, t_r = 0.5s, \%Overshoot = 5\%$.
The pole placement controller works on the concept that our state space matrices (A, B) are controllable and we know the exact locations we want the poles of our system to have. The **place** function in MATLAB calculates a feedback gain K, such that $u = -Kx$, so that our closed-loop poles of the system can be at the desired locations. Since we are also providing a reference input, our input to the system is not $u = -Kx$, but it is $u = -Kx + Lr$, where r is the reference input and $M$ is the reference gain.
Therefore our system now becomes,

$$\dot{x} = (A - BK)x + BMr \tag{37}$$

The new poles of our system are now the eigen values of $A - BK$ in the closed loop. Now the challenge is to find the desired pole locations for our system such that we can successfully track the reference as well as keep the pendulum upright.
Using our rise time ,settling time and overshoot requirements, we compute the damping ratio $\zeta$ and the damped frequency $\omega_d$. The result is $\zeta \approx 0.7$ and $\omega_d \approx 2.856$. We take 2 complex conjugate dominant poles which are chosen to satisfy the natural frequency and damping ratio requirements. The conjugate poles according to second order under damped system specifications are

$$p_1 = -\sigma + jw_d$$

$$p_1 = -\sigma - jw_d$$

where $\sigma = \zeta w_n$ and $w_d = w_n\sqrt{(1-\zeta^2)}$. The remaining closed-loop poles are placed along the real axis to the left of the dominant poles. We take the 2 real poles to be -30 and -40 so that the system response is defined by the two other dominating poles. Therefore our state feedback gain K=place(A, B, $[-2.8 + j2.856, -2.8 - j2.856, -30, -40]$).

### 5.3.2 LQR controller

The second type of controller that we explore in order to balance-up is the Linear Quadratic Regulator or LQR. This controller also returns a state-feedback gain, but using a different method as compared to pole placement. LQR uses 2 matrices(Q,R) which are responsible for imposing relative penalties on the states and inputs respectively. Mathematically put, these Q and R are used in an optimisation energy function resulting in a controller gain K. This gain K is applied to the system in the same way as the pole placement gain from the previous section. The diagonal values in Q and the value of R need to be tuned experimentally to determine the appropriate weights/penalties to each state and input. After a lot of tuning, our Q and R matrices are found to be

$$Q = \begin{bmatrix} 5*10^9 & 0 & 0 & 0 \\ 0 & 10^3 & 0 & 0 \\ 0 & 0 & 10^6 & 0 \\ 0 & 0 & 0 & 10^2 \end{bmatrix}, \; R = 5*10^{12}$$

Using the MATLAB syntax K=lqr(A, B, Q, R), we obtain the state feedback gain K which is then used in the system. The way LQR works is that it tries to optimise the energy cost function(assuming no terminal cost) by taking u=-Kx.

$$J(u) = \int_0^\infty (x^T Q x + u^T R u) dt \tag{38}$$

subject to

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

K is derived from the solution of the closed-loop Riccati Equation. As we can see from the Equation 38, the optimization depends on the weights Q and R, but it does not depend on the absolute values of the weights, it depends on their balance with respect to each other. So while tuning we observed that in order to stabilize the position at upright, the voltage needs to be extremely low, within 0.1V, hence the weight attached to it had to be of a very high order. Similarly, we observed that there were plenty of oscillations and errors in the rotary arm, and thus we used higher weights to impose more precision on the gains corresponding to $\alpha$ and $\dot{\alpha}$.

### 5.3.3 Balance-Up Control Strategy

To implement this, we use a control switching technique. We define an angular range(for example 5°) on either side about the position where $\theta = \pi$. We design our Simulink system such that when the pendulum angle is within this range, the feedback loop is activated and the controller provides an actuation input to the system. If the pendulum is out of this range, no controller is active and it just falls. So to use this balance up control system, the pendulum needs to be manually placed at $\theta = \pi$. The reference tracking for $\alpha$ is straightforward. We provide a reference signal, like a sine or square wave and the controller is responsible for ensuring that the reference is accurately followed. Thus, the controller gains have to be designed such that the $\alpha$ reference value is tracked and such that $\theta$ is stabilized at the upright position. Mathematically put,

$$u = \begin{cases} u_{balance} & \frac{175\pi}{180} \leq \theta \leq \frac{185\pi}{180} \\ 0 & otherwise \end{cases}$$

### 5.3.4 Performance discussion

In Figure 5.5, the closed-loop system performance with two designed controllers for balance-up, and rotary arm square reference signal (amplitude=±30°, time period=30s) tracking, and disturbance rejection are compared. At around 23 seconds, the rotary arm is disturbed manually with a deviation of around $-10°$, and is rejected successfully. One can observe that the resolution of the pendulum link measurement $\theta$ is seen from the constant vibration of the pendulum link position. The vibration

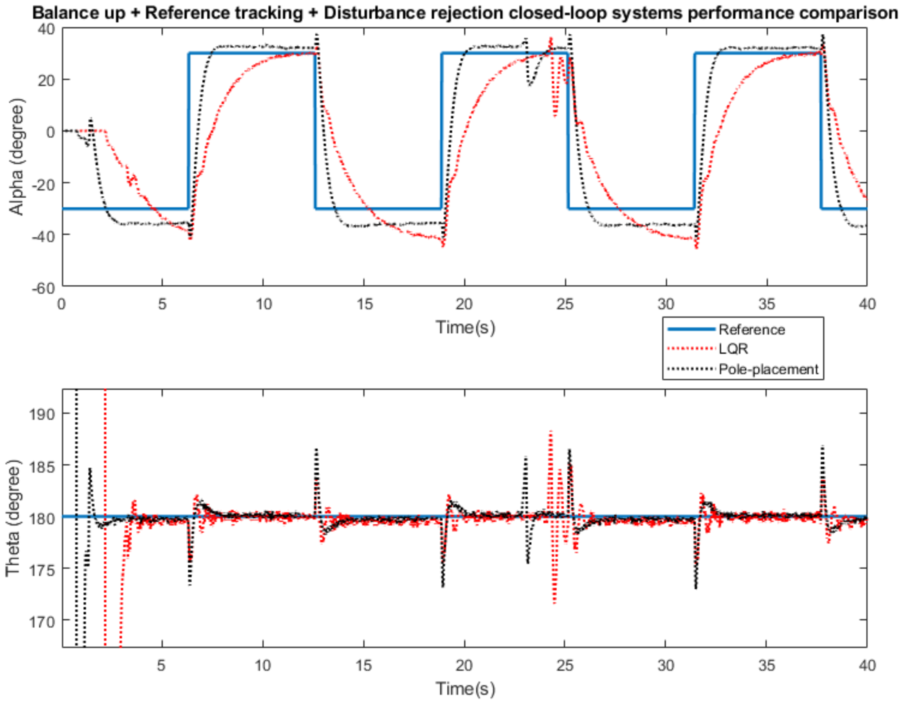

Figure 5.5: Balance up + reference tracking ($\alpha$ square singal) + disturbance rejection (at 23 seconds ) performances of two controllers

is also due to balancing movements of the DC motor, which stabilize the pendulum link to the upright position.

Both controllers can achieve reference tracking while reject the disturbance. The Pole-placement controller performs better than the LQR controller in terms of less rise time ($t_r(pole) = 1.2s$, $t_r(LQR) = 3.6s$), less settling time ($t_s(pole) = 1.8s$, $t_s(LQR) = 4.3s$) and smaller steady state error at $\alpha = -30°$ (Offset(pole)=$-7.4°$, Offset(LQR)=$-10°$). One can note that the offset of both controllers at $\alpha = 30°$ is much smaller than at $\alpha = -30°$. The potential reason is that as discussed in section 4.4.1 and Table 4.5, the optimal parameter values for CW (in this case, 30°)and CCW ($-30°$)directions are different and thus the setup is not symmetric.

The parallel-connected integrating controller is applied to compensate this offset, but the selection of the integration gain is time-consuming, and in the worse scenario, could cause more oscillating response and even instability of the system. The performance could also be improved by optimizing the system parameters via closed-loop identification, which is achieved in the last section.

## 5.4 Hybrid Swing-Up Controller

Our final controller discussion of this report is the swingup controller. Swingup controller means that the pendulum should start swinging up on its own from the downright position, and then balance itself at the upright position, while the rotary arm tracks a reference. The strategy we use is as shown in Figure 5.6. We use the balance up controller, same as in the previous section when the pendulum arm is at a certain range about the upright position. While previously, we did not have a controller outside this range, now we design a swing up controller, whose objective is to provide enough energy to the pendulum to swing up to the balance up controller range. At this range the swing up controller switches to the balance up controller and we can stabilise the pendulum at the upright position. Mathematically put,

$$u = \begin{cases} u_{balance} & \frac{175\pi}{180} \leq \theta \leq \frac{185\pi}{180} \\ u_{swingup} & otherwise \end{cases}$$

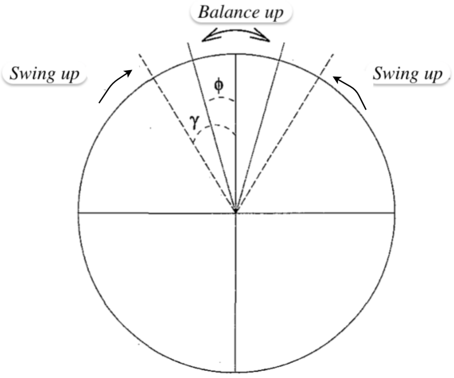

Figure 5.6: Switching strategy between balance-up ($\phi = \pm 5°$) and swing-up ($\gamma = \pm 5°$) controller

### 5.4.1 Controller Design

We again focus on the physical model of the pendulum to understand how to design the swingup controller for the pendulum. We will model our controller based on the principle of conservation of energy. We take the 0 potential reference to be at the COM of the pendulum when its at the downright position. Therefore, at any angle $\theta$, the potential energy is

$$V = mgl(1 - cos\theta) \tag{39}$$

and the kinetic energy is

$$K = \frac{J_p\dot{\theta}^2}{2} \tag{40}$$

At the upright position(when $\theta = \pi$), the potential energy is $V = 2mgl$. The total energy of our system is

$$E = V + K = mgl(1 - cos\theta) + \frac{J_p\dot{\theta}^2}{2} \tag{41}$$

and

$$\dot{E} = \frac{dE}{dt} = m_pgl\dot{\theta}sin\theta + J_p\dot{\theta}\ddot{\theta} \tag{42}$$

We also know the non-linear equation of motion of our pendulum, which is

$$J_p\ddot{\theta} + m_pglsin\theta + m_placos\alpha = 0 \tag{43}$$

We are ignoring the dissipating friction term $b_p$ here since the energy conversation rule will not hold true with them. a is the linear acceleration of the pendulum caused by the movement of the rotary arm, which in term is caused by the motor, which in turn is caused by the input voltage. In Equation 43, we solve for $J_p\ddot{\theta}$ and substitute it in Equation 42. We get

$$\dot{E} = -m_pal\dot{\theta}cos\theta \tag{44}$$

We can see that the energy is directly proportional to $a$. Since we know that the linear acceleration of the pendulum is directly proportional to the motor torque, which in turn is proportional to the voltage as we have seen earlier, we can write

$$a = (E_r - E)\dot{\theta}cos\theta \tag{45}$$

We also know that $V_m = \frac{R_mrm_r}{k_m}a$. The control in equation 45 will ensure that the $E_r - E$ term is driven to 0. If we set the reference energy($E_r$) as the energy at the upright position($E_r = 2m_pgl$), the controller will try to minimise the energy difference by swinging the pendulum up to the upright position. To further amplify this value we can add a gain($k_e$) and to ensure that the control signal does not attain too high a value, we can clamp it(saturation). Therefore, our final control law looks like

$$a = sat_{(a_{max}, -a_{max})}(k_e(E_r - E)sgn(\dot{\theta}cos\theta)) \tag{46}$$

where the signum function is used for switching the signs of the control input quickly. So our final control law is

$$V_m = \frac{R_m r m_r}{k_m} sat_{(a_{max}, -a_{max})}(k_e(E_r - E)sgn(\dot{\theta}cos\theta)) \tag{47}$$

To summarise, this control law will ensure that a voltage is provided such that the pendulum arm can reach its energy reference value. In our case, we want the reference value to be at the upright position where there is only potential energy and no kinetic energy. We use a tunable proportional gain as well as a sing switching mechanism to ensure that the controller works as desired.
When the pendulum arm reaches a certain angle, our hybrid controller structure will switch from this controller to our balance up controllers(either LQR or Pole Placement, which is covered in the previous section) and then the controller will behave just like it did previously.

### 5.4.2 Performance Discussion

In Figure 5.7, the swing up performances under different $a_{max}$ are compared, where one can notice that a max drive of $a_{max} = 5\ cm/s^2$ is enough to swing up the pendulum in a reasonable number of swings (± 9). With larger values, it will lead to less number of swings and a shorter swing up time, and vice versa. At around 7 seconds, the pendulum is manually disturbed and the controller takes only 3 seconds (for $a_{max} = 20$) and 6 seconds (for $a_{max} = 10$) to swing up again.

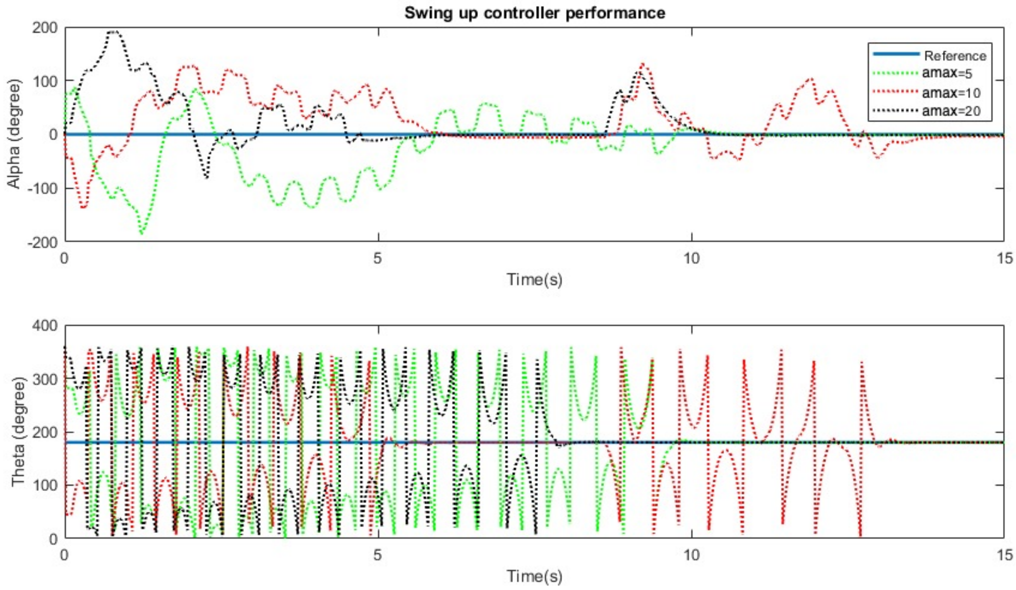

Figure 5.7: Swing up controller performances under different $a_{max}$

# 6 Closed-loop identification

In order to improve the closed-loop reference tracking performance, the linearized system at the upright position with pole-placement feedback controller and high-pass filter state observer is identified via the linear grey box identification algorithm. Different than the downright open-loop system identification, the $A$ matrix in the grey box for the closed-loop identification becomes $A - B * K$, where K is the feedback gain. The objective parameter for closed-loop identification are $b_r$ and $k_g$. The initial state condition is chosen as $x(0) = [-0.08; \pi; 0; 0]$ (the upright position), and the simulated response comparison is shown in Figure 6.1. The optimal value $b_r$ does not change much but the optimal value of $k_p$ increase to 113, which is reasonable because the balance-up requires larger voltage to the motor to provide larger torque force. The prediction rate is much lower ($P_r$ for $\theta = 5.43\%$, and for $\theta = 13.542\%$)than the open-loop identification, but the grey box model can balance up the pendulum and swing within the same $\alpha = \pm 0.1 rad$ around the initial position as the collected data from the setup.

With optimal parameters for the upright linearized system, the closed-loop performances of two controllers for swing up +balance up + reference tracking (sine wave signal with amplitude=±30°, time period=10s) + disturbance rejection are compared again as shown in Figure 6.2. It takes 4 seconds (Pole-placement) and 9 seconds (LQR) to swing up the pendulum. At 26 seconds, the manual disturbance with a deviation of $\alpha = 13°$ is rejected by both controllers successfully. The offset of both controllers decrease significantly to $\pm 5$ *degree*.

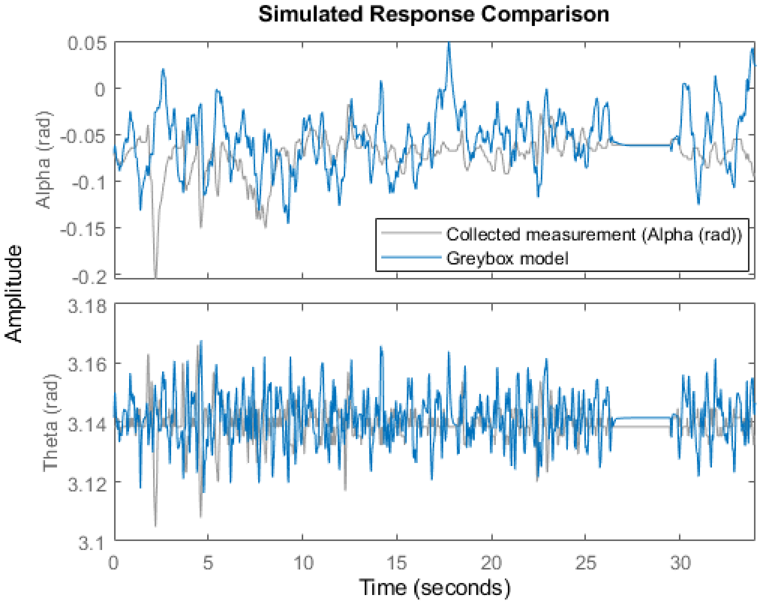

Figure 6.1: Closed-Loop Identification Result Comparison

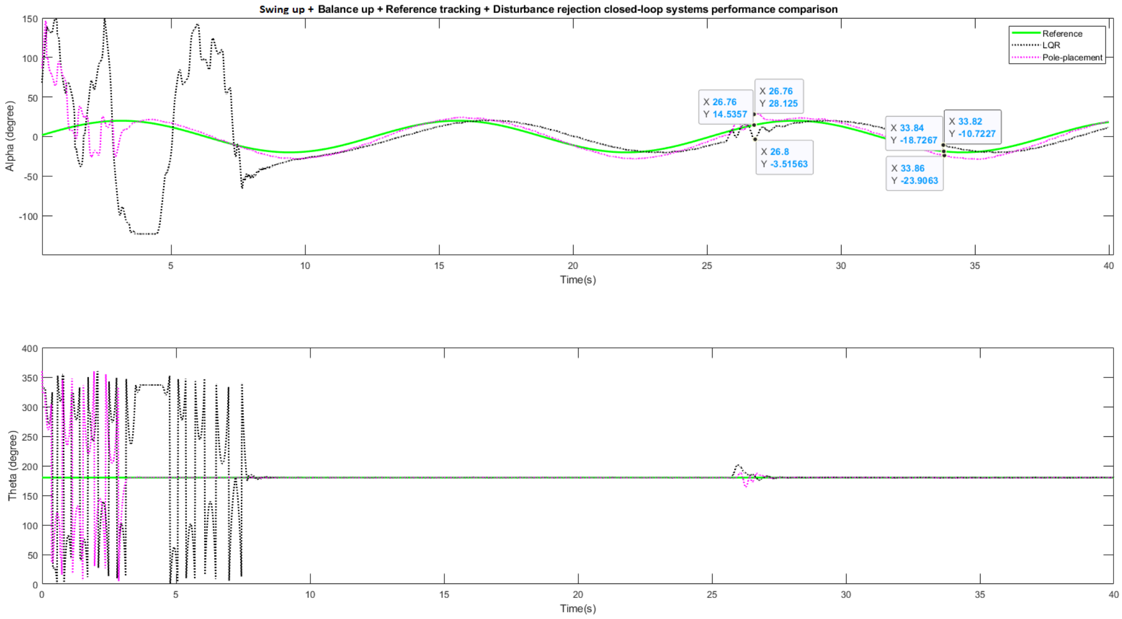

Figure 6.2: Controller Performance Comparison after the parameter estimation of closed-loop system

# 7 Conclusion

Through this report, we were able to control the Quanser Qube-2 Servo Inverted Pendulum System using a LQR and a Pole Placement controller, both for swing up and balance up applications. To summarize:

- While we achieve good results with both these controllers, there is scope for improvement in our results. The main disadvantage of LQR is that it requests a good knowledge about the state of the problem which is not always possible.

- The disadvantage of the pole placement method is that location of the closed loop poles on the complex plain is quiet difficult, and requires high level of skills in establishing relationship between poles and dynamic performances of the closed loop control system, and often the heuristic setting of poles is implemented.

- A better upgrade in terms of controller could be Model Predictive Control or MPC. It is also an optimization based controller like LQR, but it is usually finite-time and receding horizon optimisation based. While a Model Predictive Controller often looks at fixed length, often graduatingly weighted sets of error functions, the Linear Quadratic Regulator looks at all linear system inputs and provides the transfer function that will reduce the total error across the frequency spectrum, trading off state error against input frequency.

- LQR has better global stability properties, but MPC often has more locally optimal and complex performance because with MPC a new solution is computed often whereas LQR uses the same single (optimal) solution for the whole time horizon.

- Other avenues that can be explored for control are LQG, which we briefly covered in our report, and Adaptive Control, in which must adapt to a controlled system with parameters which vary, or are initially uncertain.

# References

[1] M. Arnolds, "Identification and control of the rotary inverted pendulum," 2003.

[2] M. Gevers, "Identification for control: From the early achievements to the revival of experiment design," *European journal of control*, vol. 11, no. 4-5, pp. 335–352, 2005.

[3] H. A. Nielsen, H. Madsen, *et al.*, *Predicting the heat consumption in district heating systems using meteorological forecasts.* Citeseer, 2000.

[4] H. A. Nielsen and H. Madsen, "Modelling the heat consumption in district heating systems using a grey-box approach," *Energy and buildings*, vol. 38, no. 1, pp. 63–71, 2006.

[5] "https://www.mathworks.com/help/ident/ref/greyestoptions.html," MatLab documentation: gretestOptions.

[6] "https://www.mathworks.com/help/ident/ug/model-quality-metrics.html," MatLab documentation: Loss Function and Model Quality Metrics.

[7] T. Räsänen and V.-P. Pyrhönen, "State feedback control of a rotary inverted pendulum," in *Automaatiopäivät 23: 15-16.5. 2019, Oulu*, Suomen Automaatioseura, 2019.

[8] M. L. Jacob Apkarian, *QUBE-Servo Experiment for Matlab/Simulink Users Student WorkBook.* ABET Press, 2017.

[9] P. N. Paraskevopoulos, *Modern control engineering.* CRC Press, 2017.

[10] M. Verhaegen and V. Verdult, *Filtering and system identification: a least squares approach.* Cambridge university press, 2007.

[11] W. Bolton, *Control Systems.* Elsevier Ltd, 2002.