Title
Sensor-Based Attitude Estimation of the Human Arm

Author
Kannanda, Vikas Chinnappa

Publication Date
2017

UNIVERSITY OF CALIFORNIA, SAN DIEGO

Sensor-Based Attitude Estimation of the Human Arm

A Thesis submitted in partial satisfaction of the
requirements for the degree Master of Science

in

Engineering Sciences (Mechanical Engineering)

by

Vikas Chinnappa Kannanda

Committee in charge:

Professor Mauricio de Oliveira,Co-Chair
Professor Robert Bitmead, Co-Chair
Professor Michael Tolley

2017

The Thesis of Vikas Chinnappa Kannanda is approved and it is acceptable in quality and form for publication on microfilm and electronically.

Co-Chair

Co-Chair

University of California, San Diego

2017

DEDICATION

This thesis is dedicated to my parents and grandparents, who all encouraged me to pursue a graduate degree. To my mom and dad, for encouraging me an interest in science and engineering from an early age and supporting through all my failings. To my brother for staying with me through thick and thin.

I would especially like to thank my advisor, Professor de Oliveira, for his guidance and patience, and for helping me to see the light.

TABLE OF CONTENTS

LIST OF FIGURES

ABSTRACT OF THE THESIS

Sensor-Based Attitude Estimation of the Human Arm

by

Vikas Chinnappa Kannanda

Master of Science in Engineering Sciences (Mechanical Engineering)

University of California, San Diego, 2017

Mauricio de Oliveira,Co-Chair
Robert Bitmead, Co-Chair

Wearable technology has grown rapidly in popularity in recent years. The advances in accessible micro-controllers, affordable sensors, and high-level interface design suites has made this possible. At the same time, interfaces for gestural control have become more attainable. A concurrent trend in DIY electronic instruments and controllers has evolved for similar reasons. In combining these phenomena, and in searching for greater control of live graphical interfaces, several people have developed wearable controllers in the form of exoskeletons and gloves. In general, the impetus behind wearable controllers, and more specifically optical tracking, is to impart the ability to control graphical interfaces through gestures.

Optimal state estimation theory is used in this thesis to build a linear model of the arm which by default has high degree of non-linearity. By using Kalman filtering, we can estimate the complex dynamics of a non-linear system with the simplicity of a linear system. First a model which contains the dynamics of the arm is introduced as the plant. After some definitions and derivations of filtering are established, the position is then estimated using an optimal filter. The use of a model that introduces the torques applied as a state variable instead of an input greatly improves the estimation.

The thesis is concluded with a design of a lightweight hardware architecture that can be used to implement the estimation. The estimation is found to deliver fast tracking of the highly non-linear system but breaks down at certain points due to the nature of Euler angles producing Gimbal lock. To overcome this the concept of quaternions algebra is studied and basic position calculation is done using quaternions.

Chapter 1

Introduction

The growth of wearable gadgets and accessories has been exponential in recent years. Among these wearable technologies are gestural controllers. The demand for such controllers has seen a steady rise in the recent past. Many commercially available controllers that use IMU’s to as the main sensor only employ gesture tracking and do not track the attitude of the entire arm. These kinds of controllers can only recognize certain gestures or movements, and cannot provide the exact location of the arm in virtual space. The Mi.Mu glove is probably most well-known or established of the gloves currently on the market. It was designed and is maintained and developed by a group of electronic musicians and researchers from the U.K.

The other kind controllers rely on optical tracking to control live graphic interfaces. These systems are ones currently used by large scale VR providers such as HTC Vive and Oculus. They use an array of infrared lights and cameras to track the controllers held by a user. These controllers are expensive and there is a loss of dexterity in using hand-held controllers. They also suffer from loss of tracking due to occlusion or lack of dynamic range of the camera.

This thesis discusses the development of the wearable tracking system that uses a set of IMU's mounted directly on the arm to estimate its attitude. Such a system mitigates the problems faced by traditional controllers. There is a higher degree of freedom compared to traditional gloves, and also greater range of motion compared to the optically tracked controllers.

The attitude of the human arm can be represented a set of pendulums that are connected end on end. These pendulums are driven at each joint by a torque produced by the muscles along the arm. The attitude of these links are represented using Euler angles which are easy to visualize, however have their own drawbacks. These angles are prone to gimbal lock; whenever the attitude is straight up or straight down, roll and heading become undefined.

To estimate the position of the links the concepts of Optimal State Estimation can be used. In this thesis, a Kalman filter is used in the estimation. The Kalman filter in its various forms is clearly established as a fundamental tool for analyzing and solving a broad class of estimation problems.

Chapter 2

Glove-based controller for Graphic Interfaces

The motivation to develop a controller so that a user can use his/her own hands for controlling live graphic interfaces as compared to using physical controllers. This led to looking at every aspect of an existing glove based controller and looking to improve it. Inspiration was taken from the Mi.Mu glove developed by researchers in the UK.

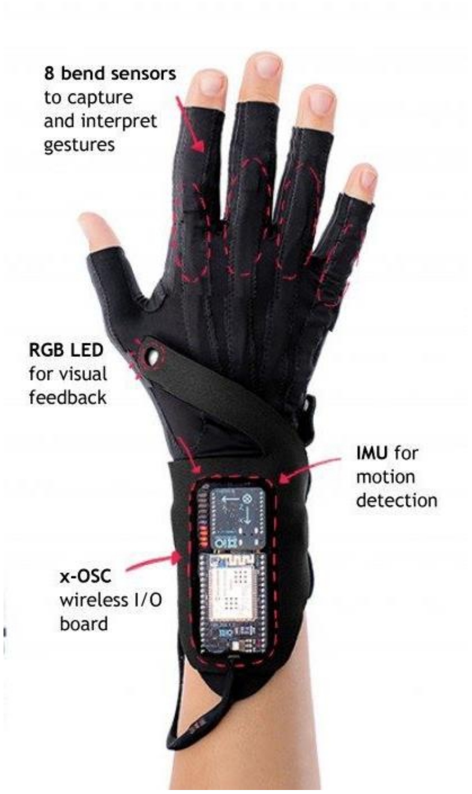
8 bend sensors to capture and interpret gestures
RGB LED for visual feedback
IMU for motion detection
x-OSC wireless I/O board

Figure 2.1 MiMu glove

The Mi.Mu glove as shown in Fig 2.1 uses 8 off the shelf flex sensors on the back of the fingers to determine the approximate bend angles of the fingers. It also houses a single IMU on the wrist for gesture control. In the following sections, we will discuss the steps taken to improve on these two aspects of the glove.

2.1 Soft Strain Sensors

The off the shelf flex sensors often used on glove based controllers have some limitations as they cannot change their length while bending. Due to this, the flex sensors need to have some degree of freedom to move around when placing it on the glove, and during continuous usage this will cause increased wear on the sensor.

To mitigate some of the reliability and durability concerns, we shall explore the use of sensors made of soft materials such as silicone. The sensors developed at Bioinspired Robotics and Design Lab at UCSD were trialed on the glove. These sensors use a thin layer of conductive cPDMS (12% MWCNT in Sylgard 184 PDMS) sandwiched between layers of silicone.

The basic design used on the glove is shown in Fig 2.3. The type of silicone used is PDMS. The first layer is spin-coated onto the base plate and left to cure partially. Next a masking tape is applied. The resistor is patterned on the masking tape using a laser cutter. The carbon grease is then spread evenly over the pattern and once finished the masking tape is removed. Finally, the top layer of PDMS is spin coated on the sensor. Once the Silicone is cured small cuts are made to the top layer to affix the leads. Once fabricated the sensors are glued onto the back of fingers of the glove using Silpoxy,a silicone based glue.

These sensors produce a very high resistance compared to general off the shelf bend sensors, in the order of 300-500 MOhm during the unstrained position. Once the sensor is strained or stretched, the resistor channels change shape and hence also changes its resistance. A small batch of these sensors were tested to check the performance.

During initial testing using a Wheatstone's bridge, the values are read using a Arduino. These sensors have a high hysteresis during operation and thus results in a slow change in the resistance. Such a sensor is not suitable for the glove as it requires a very fast sensor.

At this stage, the soft sensors even though they are very light and durable are not suitable for this application. The design will require much more development to meet the demands of fast movement. Thus for the rest of the thesis we will concentrate on the Attitude Estimation of the arm using IMU's.

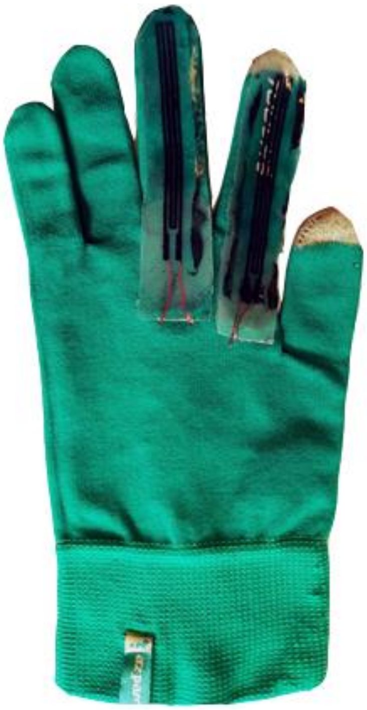

Figure 2.2 Soft Strain Sensors mounted on the glove

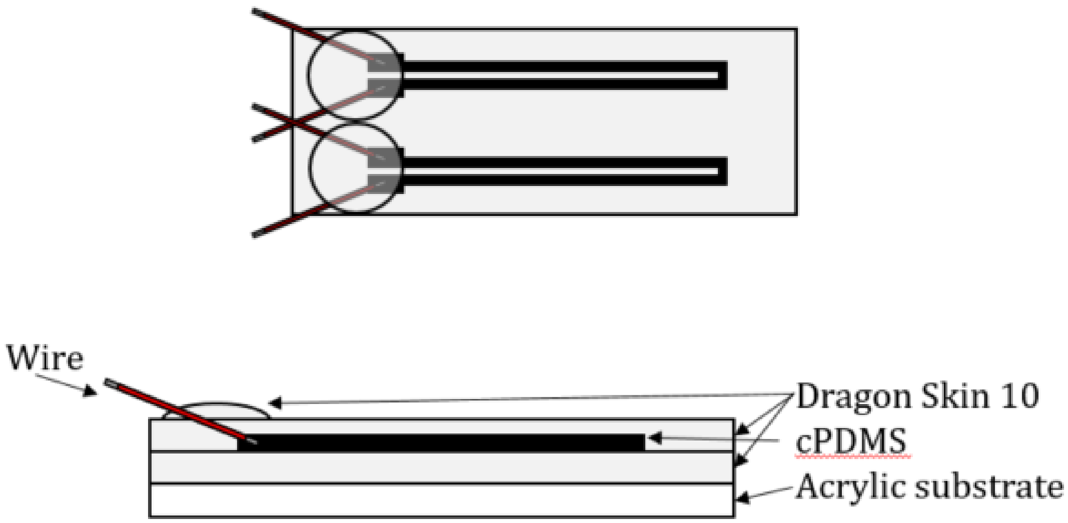

Figure 2.3 Soft Strain Sensor Schematic

Chapter 3

Sensor Configuration for Attitude estimation

3.1 IMU Postioning

In this setup, three quaternions are output form the IMU's using the DMP on them. The DMP uses proprietary sensor fusion algorithms developed by Invesense Corp. to convert the raw data from the accelerometer and gyroscope into quaternions.

The setup consists of three IMU's placed along the arm. The sensors are placed with the y-axis of the IMU pointing toward the next joint. Two are placed at the mid-points of the upper-arm and forearm, and the last one is on the back of the hand.

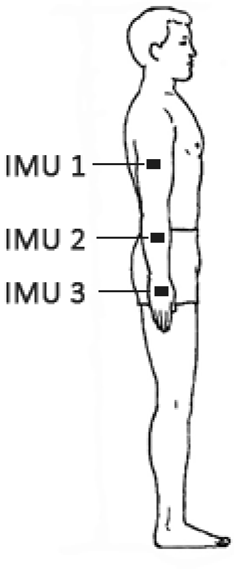

Figure 3.1 IMU Sensor positions

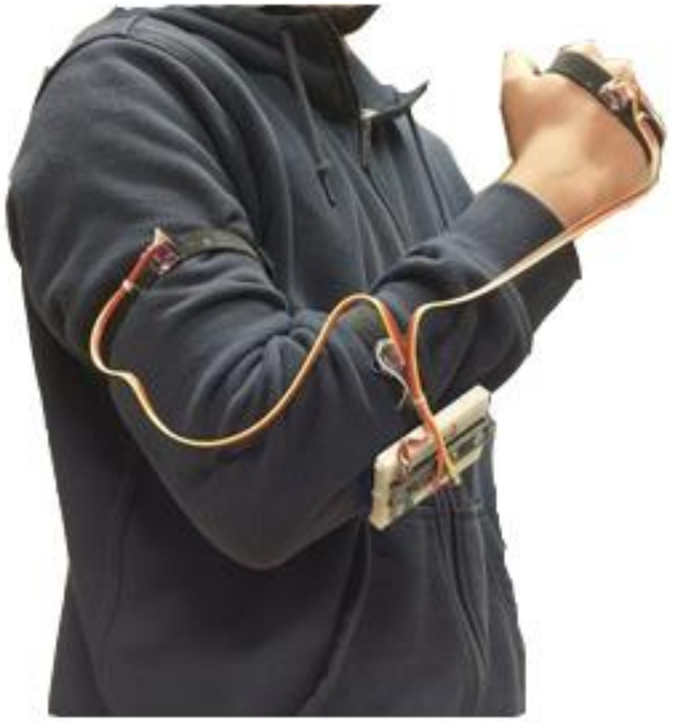

Figure 3.2 Mounting IMU using velcro

3.2 Hardware Architecture

A hardware used to implement the tracking consists of three IMU’s, specifically MPU-9250 which is a 9-axis sensor that contains an accelerometer, gyroscope and a magnetometer. The IMU’s are placed at the approximate midpoint of the links with the y-axis of all the sensors pointing toward the ground when the arm is pointing toward the ground.

Each IMU has an inbuilt low memory signal processor which uses sensor fusion algorithm to output the quaternions. The data i.e. quaternions from each of the IMU is read by a micro-controller (Arduino Micro) using an I^2C bus. Under normal operation the I^2C bus can handle only two addresses. Since we are dealing with 3 IMU’s, an I^2C multiplexer is incorporated to communicate will the IMU’s.

The quaternions are then processed on a computer running a Python script. The data is formatted and transferred to a PC using a Wi-Fi module (ESP-8266) over a local area network. The ESP-8266 module is programmed to communicate with the PC and the micro-controller. Special functions have been written to allow the user to remotely start and stop the micro-controller.

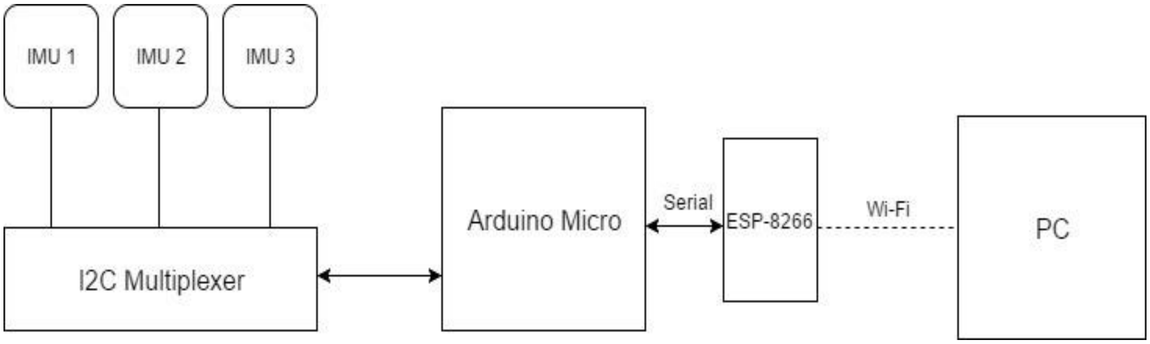

Figure 3.3 Hardware Architecture

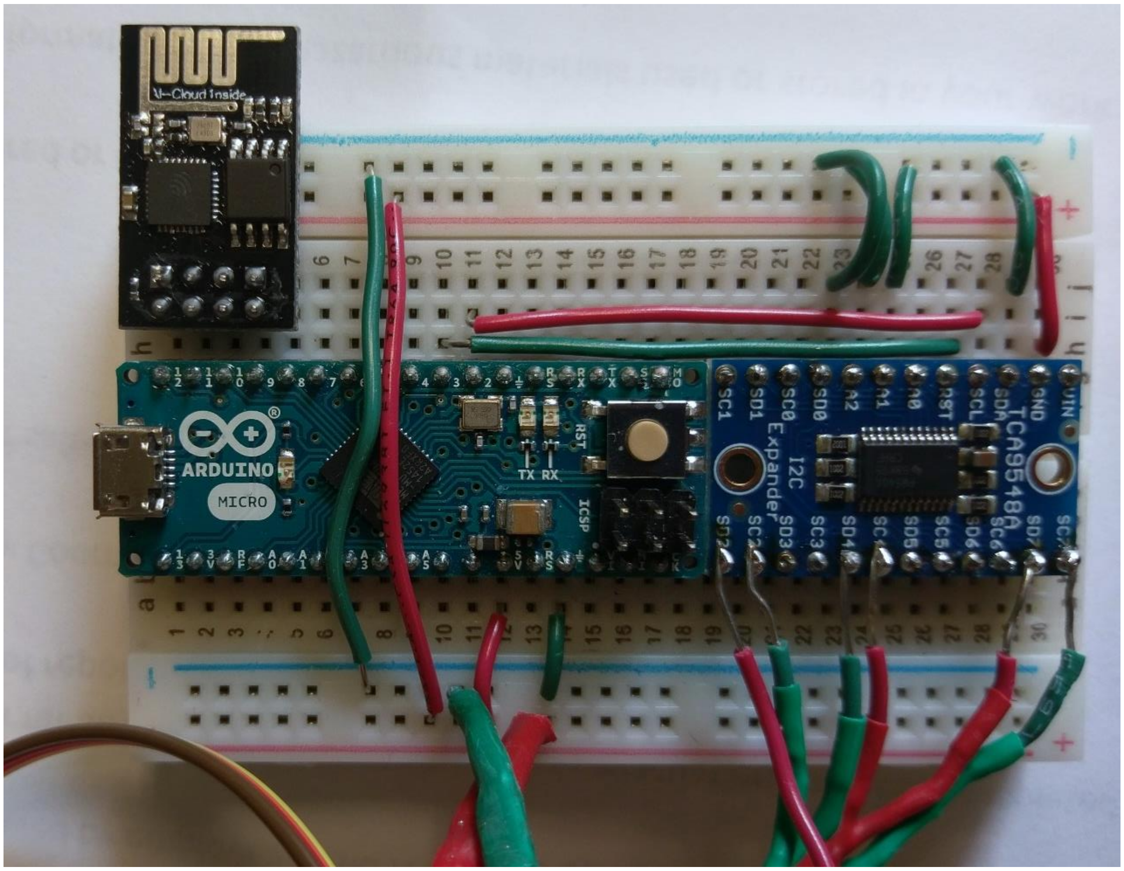

Figure 3.4 Data Transfer Electronics

3.3 Attitude tracking

Initial Attitude estimation is done using the quaternions read directly from the IMU’s. A brief introduction to quaternions is given in Section 3.4.

The co-ordinates of the shoulder are taken as (0,0,0). The co-ordinates of the elbow, wrist and orientation of the hand can be calculated using the quaternion from the respective IMU. The quaternions are first directly converted into rotation matrices using the relationships shown below:

$$R_{\bar{q}} = \begin{bmatrix} 1-2\left(q_2^2+q_3^2\right) & 2\left(q_1q_2-q_3q_4\right) & 2\left(q_1q_3+q_2q_4\right) \\ 2\left(q_1q_2+q_3q_4\right) & 1-2\left(q_1{}^2+q_3^2\right) & 2\left(q_2q_3-q_1q_4\right) \\ 2\left(q_1q_3-q_2q_4\right) & 2\left(q_2q_3+q_1q_4\right) & 1-2\left(q_2^2+q_1^2\right) \end{bmatrix} \tag{3.1}$$

By multiplying this rotation matrix with the length of the upper arm, forearm and wrist, the co-ordinates for the joints Elbow(E), Wrist(W) and Knuckle(K) can be obtained. But since the lengths of the humerus, ulna, and metacarpals differ for each person, it must be changed accordingly. There is almost a linear correlation between the height of a person and the lengths of the arms. Using anthropometric data collected of US army personnel, an approximate measure of the lengths can be found. Using only sex and height of the user as the initial data the lengths can be interpolated. The lengths are denoted by L_h, L_u, L_m for each of the corresponding links.

Using the lengths and the rotation matrix the co-ordinates of the joints can be calculated, with the shoulder assumed to be (0,0,0).

$$E = L_h \times R_{\bar{h}} \tag{3.2}$$

$$W = E + \left(L_u \times R_{\bar{u}} \right) \tag{3.3}$$

$$K = W + \left(L_m \times R_{\bar{m}} \right) \tag{3.4}$$

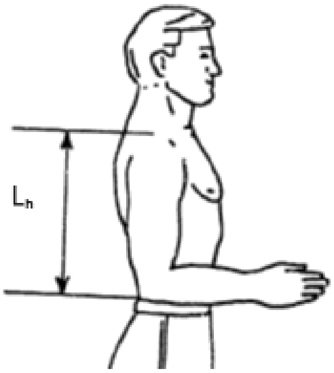

Figure 3.5 Length of Humerus

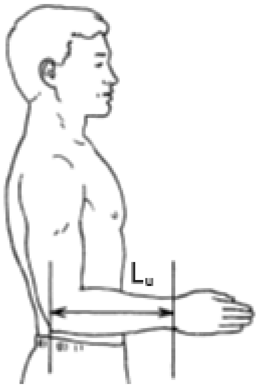

Figure 3.6 Length of Ulna

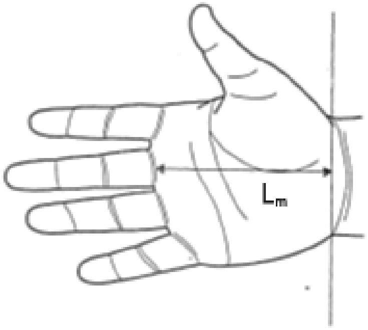

Figure 3.7 Length of Metacarpals

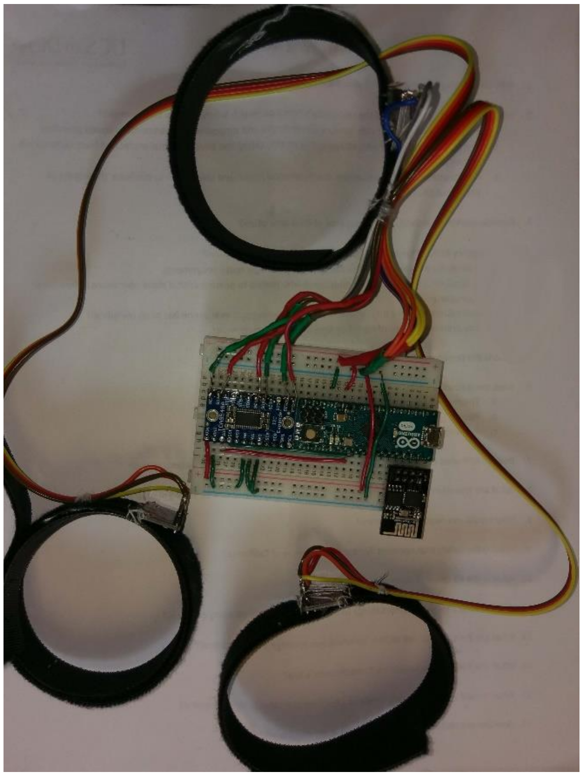

Figure 3.8 IMU's on velcro straps with Arduino and wi-fi module

3.4 Quaternion Algebra

A quaternion is an extension of complex numbers. Just as a complex number has a real part and an imaginary part i, a quaternion has a real part and three imaginary parts: i, j, and k. Notations vary between texts; this thesis will follow the JPL proposed standard convention [7], using the first three values as the imaginary part and the last value as the real part. Quaternions will be notated as a letter with an overbar, i.e. $\bar{q}$:

$$\bar{q} = \begin{bmatrix} q_1 i \\ q_2\ j \\ q_3 k \\ q_4 \end{bmatrix} \tag{3.5}$$

Much like complex numbers, the square of any of the imaginary parts of a quaternion is equal to -1. Multiplying two different imaginary parts is noncommutative; reversing order changes the sign. The imaginary combinations are:

$$\begin{array}{l} i^2 = -1 \ ij = -\mathrm{k} \ ji = k \\ j^2 = -1 \ jk = -\mathrm{i} \ kj = i \\ k^2 = -1 \ ki = -\mathrm{j} \ ik = j \end{array} \tag{3.6}$$

Note that this definition differs from that which Hamilton used. As the multiplication of imaginary parts of a quaternion is noncommutative, quaternion multiplication is also noncommutative. The product of two quaternions can be derived by multiplying each element of two quaternions:

$$\begin{aligned}\bar{q} \otimes \bar{p} = {} & \mathrm{q}_1\mathrm{p}_1 i^2 + \mathrm{q}_1\mathrm{p}_2 ij + \mathrm{q}_1\mathrm{p}_3 ik + \mathrm{q}_1\mathrm{p}_4 i \\ & + \mathrm{q}_2\mathrm{p}_1 ji + \mathrm{q}_2\mathrm{p}_2 j^2 + \mathrm{q}_2\mathrm{p}_3 jk + \mathrm{q}_2\mathrm{p}_4 j \\ & + \mathrm{q}_3\mathrm{p}_1 ki + \mathrm{q}_3\mathrm{p}_2 kj + \mathrm{q}_3\mathrm{p}_3 \mathrm{k}^2 + \mathrm{q}_3\mathrm{p}_4 k \\ & + \mathrm{q}_4\mathrm{p}_1 i + \mathrm{q}_4\mathrm{p}_2 j + \mathrm{q}_4\mathrm{p}_3 k + \mathrm{q}_4\mathrm{p}_4\end{aligned} \tag{3.7}$$

Substituting the definitions of individual complex products:

$$\begin{aligned}\bar{q} \otimes \bar{p} = {} & - \mathrm{q}_1\mathrm{p}_1 - \mathrm{q}_1\mathrm{p}_2 k + \mathrm{q}_1\mathrm{p}_3 j + \mathrm{q}_1\mathrm{p}_4 i \\ & + \mathrm{q}_2\mathrm{p}_1 k - \mathrm{q}_2\mathrm{p}_2 - \mathrm{q}_2\mathrm{p}_3 i + \mathrm{q}_2\mathrm{p}_4 j \\ & - \mathrm{q}_3\mathrm{p}_1 j + \mathrm{q}_3\mathrm{p}_2 i - \mathrm{q}_3\mathrm{p}_3 + \mathrm{q}_3\mathrm{p}_4 k \\ & + \mathrm{q}_4\mathrm{p}_1 i + \mathrm{q}_4\mathrm{p}_2 j + \mathrm{q}_4\mathrm{p}_3 k + \mathrm{q}_4\mathrm{p}_4\end{aligned} \tag{3.8}$$

Then grouping the individual complex components into a new quaternion:

$$\bar{q} \otimes \bar{p} = \begin{bmatrix} \left(+ \mathrm{q}_1\mathrm{p}_4 - \mathrm{q}_2\mathrm{p}_3 + \mathrm{q}_3\mathrm{p}_2 + \mathrm{q}_4\mathrm{p}_1\right)\mathrm{i} \\ \left(+\mathrm{q}_1\mathrm{p}_3 + \mathrm{q}_2\mathrm{p}_4 - \mathrm{q}_3\mathrm{p}_1 + \mathrm{q}_4\mathrm{p}_2\right)\mathrm{j} \\ \left(- \mathrm{q}_1\mathrm{p}_2 + \mathrm{q}_2\mathrm{p}_1 + \mathrm{q}_3\mathrm{p}_4 + \mathrm{q}_4\mathrm{p}_3\right)\mathrm{k} \\ - \mathrm{q}_1\mathrm{p}_1 - \mathrm{q}_2\mathrm{p}_2 - \mathrm{q}_3\mathrm{p}_3 + \mathrm{q}_4\mathrm{p}_4 \end{bmatrix} \tag{3.9}$$

The multiplication of the two quaternions is therefore equivalent to:

$$\bar{q} \otimes \bar{p} = \begin{bmatrix} q_4 & q_3 & -q_2 & q_1 \\ -q_3 & q_4 & q_1 & q_2 \\ q_2 & -q_1 & q_4 & q_3 \\ -q_1 & -q_2 & -q_3 & q_4 \end{bmatrix} \begin{bmatrix} p_1 \\ p_2 \\ p_3 \\ p_4 \end{bmatrix} \tag{3.10}$$

Chapter 4

Modelling the Arm

Modelling the human arm is quite complex. To create a model of the human arm we can assume it to be series on connected pendulums that driven at each joint. Imagining the arm in such a way will help visualize and formulate a reasonable model. Using the pendulum model, we can simulate the torques acting at each joint or node. And create an optimal filter to estimate the angles at the joints. We can start first with a simple model of a planar double pendulum.

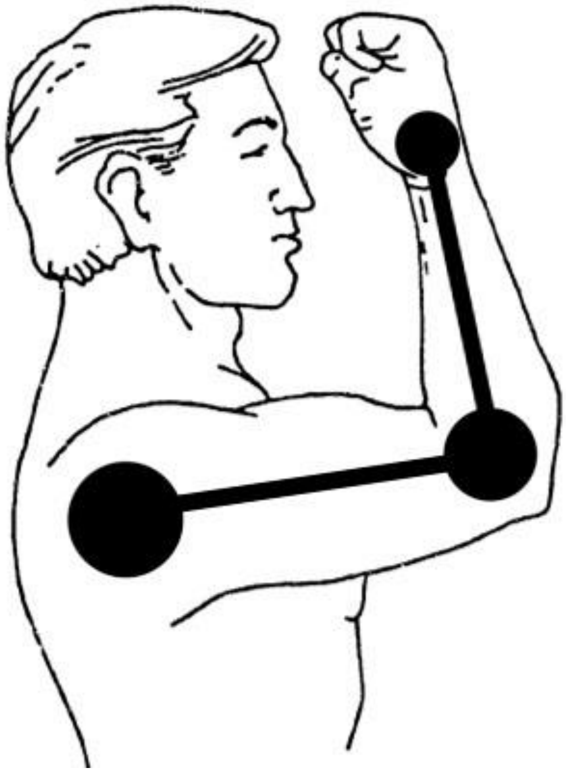

Figure 4.1 Pendulum approximation of the arm

4.1 Planar Double Pendulum

The first step in our problem will be to model the double pendulum. The first link will correspond to the upper arm and the second link to the forearm. The geometry of the planar double pendulum is as shown

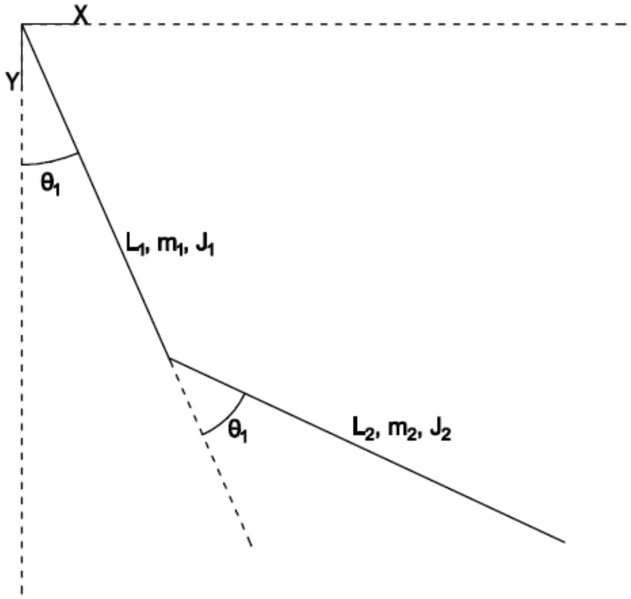

Figure 4.2 Double Pendulum Geometry

The links have masses m, length L and moment of inertia J. The two links are assumed to be uniform cylinders, so the vectors to for the center of masses is given by

$$r_1 = \begin{bmatrix} \frac{1}{2}\mathrm{L}_1 \sin(\theta_1) \\ -\frac{1}{2}\mathrm{L}_1 \cos(\theta_1) \end{bmatrix} \qquad r_2 = \begin{bmatrix} \mathrm{L}_1 \sin(\theta_1) + \frac{1}{2}\mathrm{L}_2 \sin(\theta_1 + \theta_2) \\ -\mathrm{L}_1 \cos(\theta_1) - \frac{1}{2}\mathrm{L}_2 \cos(\theta_1 + \theta_2) \end{bmatrix} \tag{4.1}$$

The velocities of each link is given by $v = \frac{dr}{dt}$

$$v_1 = \begin{bmatrix} \frac{1}{2}\mathrm{L}_1\dot{\theta}_1\cos(\theta_1) \\ \frac{1}{2}\mathrm{L}_1\dot{\theta}_1\sin(\theta_1) \end{bmatrix} \qquad v_2 = \begin{bmatrix} \mathrm{L}_1\dot{\theta}_1\cos(\theta_1)+\frac{1}{2}\mathrm{L}_2\left(\dot{\theta}_1+\dot{\theta}_2\right)\cos(\theta_1+\theta_2) \\ \mathrm{L}_1\dot{\theta}_1\sin(\theta_1)+\frac{1}{2}\mathrm{L}_2\left(\dot{\theta}_1+\dot{\theta}_2\right)\sin(\theta_1+\theta_2) \end{bmatrix} \tag{4.2}$$

To calculate the Lagrangian L, first we need the Potential $\mathrm{V_g}$ and Kinetic T energies of the system.

$$T = \frac{1}{2}\left(\mathrm{J}_1\dot{\theta}_1^{\,2}+\mathrm{J}_2\left(\dot{\theta}_1+\dot{\theta}_2\right)^2+\mathrm{m}_1\mathrm{v}_1.\mathrm{v}_1+\mathrm{m}_2\mathrm{v}_2.\mathrm{v}_2\right) \tag{4.3}$$

$$T = \frac{1}{2}\begin{pmatrix} \mathrm{J}_1\dot{\theta}_1^{\,2}+\mathrm{J}_2\left(\dot{\theta}_1+\dot{\theta}_2\right)^2+\mathrm{m}_1\left(\frac{1}{4}\mathrm{L}_1^{\,2}\dot{\theta}_1^{\,2}\sin^2(\theta_1)+\frac{1}{4}\mathrm{L}_1^{\,2}\dot{\theta}_1^{\,2}\cos^2(\theta_1)\right) \\ +\mathrm{m}_2\begin{pmatrix} \left(\mathrm{L}_1\dot{\theta}_1\sin(\theta_1)+\frac{1}{2}\mathrm{L}_2\left(\dot{\theta}_1+\dot{\theta}_2\right)\sin(\theta_1+\theta_2)\right)^2 \\ +\left(\mathrm{L}_1\dot{\theta}_1\cos(\theta_1)+\frac{1}{2}\mathrm{L}_2\left(\dot{\theta}_1+\dot{\theta}_2\right)\cos(\theta_1+\theta_2)\right)^2 \end{pmatrix} \end{pmatrix} \tag{4.4}$$

$$V_g = g\mathrm{m}_1\mathrm{r}_1\begin{pmatrix} 0 \\ 1 \end{pmatrix}+g\mathrm{m}_2\mathrm{r}_2\begin{pmatrix} 0 \\ 1 \end{pmatrix} \tag{4.5}$$

$$V_g = g\mathrm{m}_2\left(-\mathrm{L}_1\cos(\theta_1)\text{-}\frac{1}{2}\mathrm{L}_2\cos(\theta_1+\theta_2)\right)-\frac{1}{2}g\mathrm{L}_1\mathrm{m}_1\cos(\theta_1) \tag{4.6}$$

The Lagrangian is calculated using

$$L = T - V_g \tag{4.7}$$

Now, to calculate the equations of motion

$$\left(\frac{d}{dt}\frac{\partial L}{\partial\dot{\theta}_1}\right)-\frac{\partial L}{\partial\theta_1}+\mathrm{b}\dot{\theta}_1+\mathrm{b}\dot{\theta}_2=\tau_1\text{-}\tau_2 \tag{4.8}$$

$$\left(\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}_2}\right)-\frac{\partial L}{\partial \theta_2}+\mathrm{b}\dot{\theta}_2=\tau_2 \tag{4.9}$$

The equations of motion can be written as Non-Linear state space system with states X and input u

$$X=\begin{bmatrix}\theta_1\\ \theta_2\\ \dot{\theta}_1\\ \dot{\theta}_2\end{bmatrix}=\begin{bmatrix}x_1\\ x_2\\ x_3\\ x_4\end{bmatrix}\quad u=\begin{bmatrix}\tau_1\\ \tau_2\end{bmatrix} \tag{4.10}$$

$$\dot{X}=\begin{bmatrix} x_3\\ x_4\\ \frac{-1}{\left(4J_2+L_2{}^2m_2\right)\left(4J_1+L_1{}^2\left(m_1+4m_2\right)\right)-...}\left(-\frac{1}{2}4\mathrm{J}_2+2\mathrm{L}_1\mathrm{L}_2\mathrm{m}_2\cos(\mathrm{x}_2)+\mathrm{L}_2{}^2\mathrm{m}_2\left(g\mathrm{L}_2\mathrm{m}_2\sin(\mathrm{x}_1...\right)\right.\\ \frac{-1}{\left(4\mathrm{J}_2+\mathrm{L}_2{}^2\mathrm{m}_2\right)\left(4\mathrm{J}_1+\mathrm{L}_1{}^2(\mathrm{m}_1+4\mathrm{m}_2)\right)-...}\left(2\left(-4g\mathrm{J}_2\mathrm{L}_1\mathrm{m}_1\sin(\mathrm{x}_1)-8g\mathrm{J}_2\mathrm{L}_1\mathrm{m}_2\sin(\mathrm{x}_1)-g\mathrm{L}_1\mathrm{L}_2{}^2\mathrm{m}_1...\right)\right.\end{bmatrix}$$

$$(4.11)$$

From this Non-Linear model, we can easily calculate a Linear model around a suitable equilibrium. The equilibrium points for this system are

$$\begin{pmatrix}\theta_1\\ \theta_2\end{pmatrix}=\begin{pmatrix}0\\ 0\end{pmatrix},\begin{pmatrix}0\\ \pi\end{pmatrix},\begin{pmatrix}\pi\\ \pi\end{pmatrix}$$

Linearizing about the point (0,0) we get

$$\begin{aligned}\dot{X}&=AX+Bu\\ Y&=CX+Du\end{aligned} \tag{4.12}$$

Substituting values for all constants $L_1=L_2=m_1=m_2=b_1=b_2=1, g=9.81$

We get,

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ \text{-6.2210} & 0.9571 & \text{-0.4878} & 0.1951 \\ 4.7854 & \text{-5.2639} & 0.6829 & 1.0732 \end{bmatrix}$$

$$B = \begin{bmatrix} 0 & 0 \\ 0 & 0 \\ 0.4878 & -1.1707 \\ -0.6829 & 2.4390 \end{bmatrix} \qquad C = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \tag{4.13}$$

4.2 Spherical Pendulum

The next step in our modelling problem is to move from a planar case to a 3-dimensional geometry. We can start by modelling a single spherical pendulum and discussing its dynamics. The geometry of the Spherical pendulum is as shown in Fig.

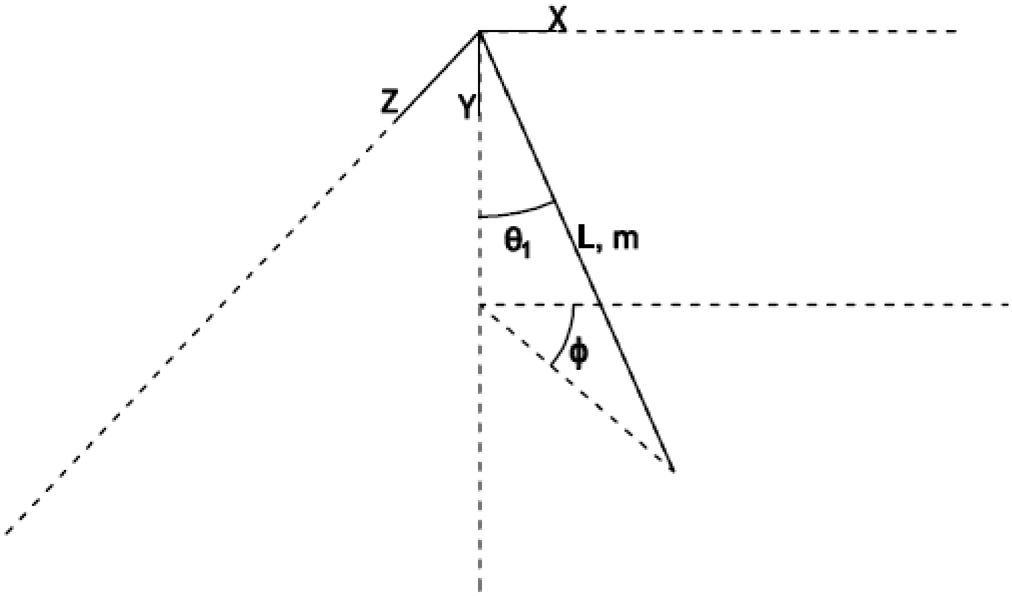

Figure 4.3 Spherical Pendulum Geometry

The same procedure is followed as earlier in calculating the Lagrangian and formulating the equations of motion. The vector of the center of mass is given by

$$r = \begin{bmatrix} \frac{1}{2} L \sin(\theta)\cos(\phi) \\ \frac{1}{2} L \cos(\theta) \\ \frac{1}{2} L \sin(\theta)\sin(\phi) \end{bmatrix} \tag{4.14}$$

The kinetic energy of the system is given by

$$T = \frac{1}{2} m v^2 + \frac{1}{2} J \vec{\omega}.\vec{\omega} \tag{4.15}$$

To simplify further calculations the inertial energy term is ignored due to the difficulty in determining the angular velocity $\vec{\omega}$ in the frame used. The Langrangian is given by

$$V_g = mgr\begin{bmatrix} 0 & 1 & 0 \end{bmatrix} \tag{4.16}$$

$$L = T - V_g \tag{4.17}$$

The equations of motion for the system is calculated by using the Lagrangian formula

$$\left(\frac{d}{dt} \frac{\partial L}{\partial \dot{\theta}} \right) - \frac{\partial L}{\partial \theta} + \mathrm{b}\dot{\theta} = \tau_1 \tag{4.18}$$

$$\left(\frac{d}{dt} \frac{\partial L}{\partial \dot{\phi}} \right) - \frac{\partial L}{\partial \phi} + \mathrm{b}\dot{\phi} = \tau_2 \tag{4.19}$$

The equations of motion are written in Non-Linear State space representation using the state X

$$X = \begin{bmatrix} \theta \\ \phi \\ \dot{\theta} \\ \dot{\phi} \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \tag{4.20}$$

$$\dot{X} = \begin{bmatrix} x_3 \\ x_4 \\ \dfrac{8b\mathrm{x}_2 - Lm\left(4g\sin(\mathrm{x}_1) + L\mathrm{x}_4{}^2\sin(2\mathrm{x}_1)\right) - 8\tau_1}{2L^2m} \\ -\dfrac{\csc^2(\mathrm{x}_1)\left(x_4\left(4b + L^2mx_2\sin\left(2x_1\right)\right) - 4\tau_2\right)}{L^2m} \end{bmatrix} \tag{4.21}$$

This Non-Linear model is linearized around a suitable equilibrium point. The equilibrium points of this system are $\theta = \left(0, \pi\right)$. It is observed that there is no value of ϕ in the equilibrium condition. This is due to ϕ not being defined when the pendulum is vertical, i.e. it is either pointing straight down or up. This phenomenon can be observed in the Non-Linear system shown in Eq(3.25) as the $\dot{x}_4$ term is not defined at $x_1 = \theta = \left(0, \pi\right)$.

The linearization is done by substituting $\theta = 0$ in the equations of motion calculated in Eq(3.22) and Eq(3.23). At this equilibrium point x4 is unobservable and is thus removed from the state. The resulting linear model will consist of only three states, namely $X = (\theta, \phi, \dot{\theta})$ with input $u = (\tau_1, \tau_2)$.

The linearized model can be written as

$$\begin{aligned} \dot{X} &= AX + Bu \\ Y &= CX \end{aligned} \tag{4.22}$$

Where

$$\left[\begin{array}{c|c} A & B \\ \hline C & D \end{array}\right] = \left[\begin{array}{ccc|cc} 0 & 1 & 0 & 0 & 0 \\ \frac{2g}{L} & \frac{-4b}{L^2 m} & 0 & \frac{4}{L^2 m} & 0 \\ 0 & 0 & 0 & 0 & \frac{1}{b} \\ \hline 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{array}\right] \tag{4.23}$$

4.3 Spherical Double Pendulum

This final model is a combination of the two previous models, combining the double pendulum and the Spherical Pendulum. An additional degree of freedom is implemented in this model. The roll angle α about link 1 is also incorporated. The geometry is as shown below in Fig----

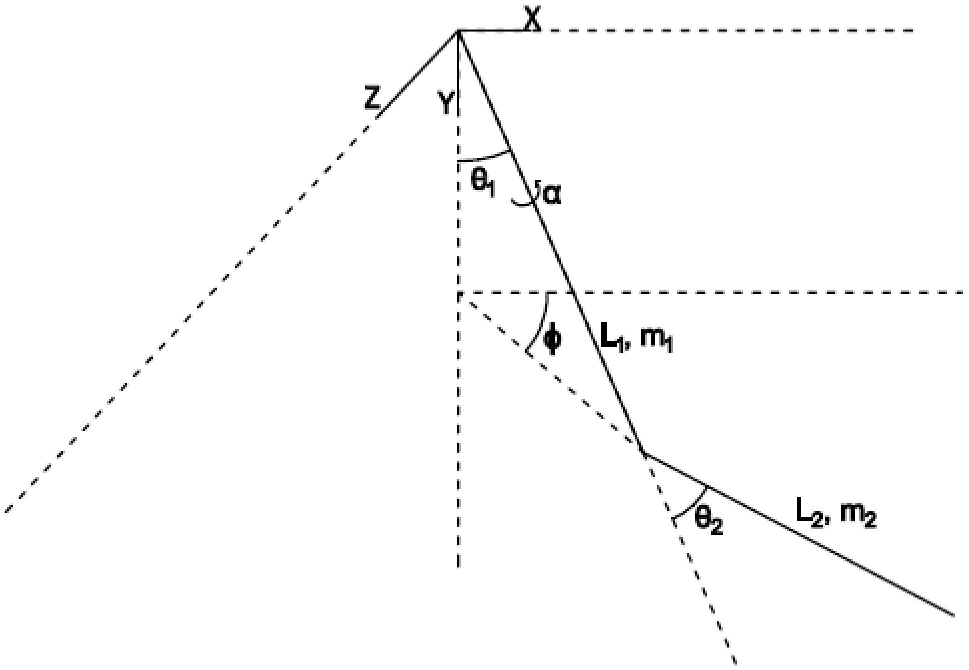

Figure 4.4 Spherical Double Pendulum Geometry

The procedure to find the equations of motion for the system is the same as the previous models. Fist consider the planar double pendulum system, where z_1 *and* z_2 are the unit vectors to each link.

$$z_1 = \begin{bmatrix} \sin(\theta_1) \\ \cos(\theta_1) \\ 0 \end{bmatrix} \qquad z_2 = \begin{bmatrix} \sin(\theta_1+\theta_2) \\ \cos(\theta_1+\theta_2) \\ 0 \end{bmatrix} \tag{4.24}$$

A Rotation Matrix $R_\theta\left(a\right)$ is defined that will rotate a vector by an angle θ about vector $a = \left[x, y, z\right]$.

$$R_\theta(a) = \begin{bmatrix} \cos(\theta) + x^2(1-\cos(\theta)) & xy(1-\cos(\theta)) - z\sin(\theta) & xz(1-\cos(\theta)) + y\sin(\theta) \\ xy(1-\cos(\theta)) + z\sin(\theta) & \cos(\theta) + y^2(1-\cos(\theta)) & yz(1-\cos(\theta)) - x\sin(\theta) \\ xz(1-\cos(\theta)) - y\sin(\theta) & x\sin(\theta) + yz(1-\cos(\theta)) & \cos(\theta) + z^2(1-\cos(\theta)) \end{bmatrix} \tag{4.25}$$

To formulate the vectors for the Three-dimensional model we need to rotate the planar pendulum by angle ϕ about the y-axis, and link 2 is rotated by and angle α about link 1. Using these two rotations the vectors for center of masses of the two links are given by

$$\begin{aligned} r_1 &= \frac{L_1}{2} R_\phi\left(0,1,0\right).z_1 \\ r_2 &= 2r_1 + \frac{L_2}{2} R_\phi\left(0,1,0\right).z_1\, R_\alpha\left(z_1\right).z_2 \end{aligned} \tag{4.26}$$

The Lagrangian is formulated as follows

$$T = \frac{1}{2} m_1 v_1^2 + \frac{1}{2} m_2 v_2^2 \tag{4.27}$$

$$V_g = m_1 g r_1 \left[0 \quad 1 \quad 0\right]' + m_2 g r_2 \left[0 \quad 1 \quad 0\right]' \tag{4.28}$$

$$L = T - V_g \tag{4.29}$$

The Equations of motion are calculated using the Lagrangian formula

$$\left(\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}_1}\right)-\frac{\partial L}{\partial \theta_1}+\mathrm{b}\dot{\theta}_1=\tau_1-\tau_2\cos(\alpha) \tag{4.30}$$

$$\left(\frac{d}{dt}\frac{\partial L}{\partial \dot{\theta}_2}\right)-\frac{\partial L}{\partial \theta_2}+\mathrm{b}\dot{\theta}_2=\tau_2 \tag{4.31}$$

$$\left(\frac{d}{dt}\frac{\partial L}{\partial \dot{\phi}}\right)-\frac{\partial L}{\partial \phi}+\mathrm{b}\dot{\phi}=\tau_3-\tau_2\sin(\alpha) \tag{4.32}$$

$$\left(\frac{d}{dt}\frac{\partial L}{\partial \dot{\alpha}}\right)-\frac{\partial L}{\partial \alpha}+\mathrm{b}\dot{\alpha}=\tau_4 \tag{4.33}$$

The equations of motion are written in Non-Linear State space representation using the state X

$$X=\begin{bmatrix}\theta_1\\ \theta_2\\ \dot{\theta}_1\\ \dot{\theta}_2\\ \phi\\ \alpha\\ \dot{\phi}\\ \dot{\alpha}\end{bmatrix}=\begin{bmatrix}x_1\\ x_2\\ x_3\\ x_4\\ x_5\\ x_6\\ x_7\\ x_8\end{bmatrix} \tag{4.34}$$

$$\dot{X}=g\left(x_1,x_2,x_3,x_4,x_5,x_6,x_7,x_8\right) \tag{4.35}$$

It is very difficult to linearize this system about the equilibrium point $\begin{pmatrix}\theta_1\\ \theta_2\end{pmatrix}=\begin{pmatrix}0\\ 0\end{pmatrix}$ due to the appearance of singularities in the equations. Like the previous system the terms for ϕ and α will not be defined at the equilibrium point. To overcome this the non-linear system is rewritten as follows.

$$M\ddot{Z} = f \tag{4.36}$$

Where $M\left(\theta_1, \theta_2, \phi, \alpha\right), f\left(\theta_1, \theta_2, \dot{\theta}_1, \dot{\theta}_2, \phi, \alpha, \dot{\phi}, \dot{\alpha}\right)$ and $Z = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \phi \\ \alpha \end{bmatrix}$.

Like the previous problem the states of $\dot{\phi}$ and $\dot{\alpha}$ become unobservable in the linearized system. So, we can eliminate these states from the formulation and the linearized system is given by new state $\hat{X}$.

The linearized system is built as shown below,

$$E\dot{\hat{X}} = A\hat{X} + Bu \tag{4.37}$$

Where

$$E = \begin{bmatrix} I & 0 & 0 \\ 0 & E_1 & E_2 \end{bmatrix} \qquad E_1 = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \qquad E_2 = \begin{bmatrix} \dfrac{\partial f}{\partial \dot{\phi}} & \dfrac{\partial f}{\partial \dot{\alpha}} \end{bmatrix} \tag{4.38}$$

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ \dfrac{\partial f}{\partial \theta_1} & \dfrac{\partial f}{\partial \theta_2} & \dfrac{\partial f}{\partial \dot{\theta}_1} & \dfrac{\partial f}{\partial \dot{\theta}_2} & \dfrac{\partial f}{\partial \phi} & \dfrac{\partial f}{\partial \alpha} \end{bmatrix} \qquad B = \begin{bmatrix} 0 & 0 & 0 & 0 \\ \dfrac{\partial f}{\partial \tau_1} & \dfrac{\partial f}{\partial \tau_2} & \dfrac{\partial f}{\partial \tau_3} & \dfrac{\partial f}{\partial \tau_4} \end{bmatrix} \tag{4.39}$$

Then we have the Linearized system as

$$\begin{aligned} \dot{X} &= AX + Bu \\ Y &= CX \end{aligned} \qquad (4.40)$$

Where $\hat{A} = E^{-1}A\,, \hat{B} = E^{-1}B\,,\ \hat{X} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \dot{\theta}_1 \\ \dot{\theta}_2 \\ \phi \\ \alpha \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix},\ u = \begin{bmatrix} \tau_1 \\ \tau_2 \\ \tau_3 \\ \tau_4 \end{bmatrix}$. (4.41)

Chapter 5

Attitude Estimation

5.1 Optimal State Estimation

The states of a system are those variables that provide a complete representation of the internal condition or status of the system at a given instant of time. State estimation is applicable to virtually all areas of engineering and science. Any discipline that is concerned with the mathematical modeling of its systems is a likely (perhaps inevitable) candidate for state estimation.

Consider the LTI system

$$\begin{aligned} \dot{x}(t) &= Ax(t) + B_w w(t) \\ y(t) &= C_y x(t) + D_{yw} w(t) \\ z(t) &= C_z x(t) \end{aligned} \tag{5.1}$$

The problem here is to compute $\left(\hat{A}, F\right)$ such that the output of the state estimator

$$\begin{aligned} \dot{\hat{x}}(t) &= \hat{A}\hat{x}(t) - Fy(t) \\ \hat{z}(t) &= C_z \hat{x}(t) \end{aligned} \tag{5.2}$$

stabilizes the state estimation error and minimizes the cost function

$$J := \lim_{t \to \infty} E\left[\left(z(t) - \hat{z}(t)\right)^T \left(z(t) - \hat{z}(t)\right)\right] \tag{5.3}$$

The error function is defined as $e(t) := x(t) - \hat{x}(t)$ so that $z(t) - \hat{z}(t) = C_z e(t)$

The dynamics of the estimation error can be written as

$$\begin{aligned} \dot{e}(t) &= Ax(t) - \hat{A}\hat{x}(t) + Fy(t) + B_w w(t) \\ &= \left(A - \hat{A} + FC_y\right)x(t) + \hat{A}e(t) + \left(B + FD_{yw}\right)w(t) \end{aligned} \tag{5.4}$$

It can be observed that if A is unstable then both x(t) and e(t) are unbounded with only one exception

$$\hat{A} = A + FC_y \Rightarrow A - \hat{A} + FC_y = 0$$

So, we get

$$\dot{e}(t) = \left(A + FC_y\right)e(t) + \left(B + FD_{yw}\right)w(t) \tag{5.5}$$

The dynamics of the error e(t) and the state x(t) have been decoupled. And this choice of $\hat{A}$ produces the state observer

$$\begin{aligned} \dot{\hat{x}} &= A\hat{x} + F\left(\hat{y} - y\right) \\ \hat{y} &= C_y \hat{x} \\ \hat{z} &= C_z \hat{x} \end{aligned} \tag{5.6}$$

Now the problem has been simplified to find the stabilizing state estimation gain F to minimize the cost function

$$J := \lim_{t \to \infty} E\left[e(t)^T C_z^T C_z e(t)\right] \tag{5.7}$$

Where e(t) is the state of the LTI system

$$\dot{e}(t) = \left(A + FC_y\right)e(t) + \left(B_w + FD_{yw}\right)w(t) \tag{5.8}$$

The solution to this problem is to Find an F that minimizes the cost function

$$J = trace\left[X\left(B_w + FD_{yw}\right)W\left(B_w + FD_{yw}\right)^T \right] \tag{5.9}$$

Where X is the solution in the Lyapunov equation

$$\left(A + FC_y\right)^T X + X\left(A + FC_y\right) + C_z^T C_z = 0 \tag{5.10}$$

However, in this form F appears in both the cost function and the Lyapunov equation and cannot be solved simultaneously. The Duality property is used to solve this problem.

Consider the cost function

$$J = trace\left(XBWB^T\right) \tag{5.11}$$

Where X is the solution to the Lyapunov Equation

$$A^T X + XA + C^T C = 0 \tag{5.12}$$

Substituting the gramian solution of X in the cost function

$$\begin{aligned} J &= trace\left(XBWB^T\right) \\ &= trace\left(\left[\int_0^\infty e^{A^t} C^T C e^{A^t}\, dt\right] BWB^T\right) \end{aligned}$$

$$
\begin{aligned}
&= trace\left(\left[\int_{0}^{\infty} e^{A^t} C^T C e^{A^t}\, dt\right] BWB^T\right)\\
&= trace\left(C\left[\int_{0}^{\infty} e^{A^t} BWB^T e^{A^t}\, dt\right] C^T\right)
\end{aligned}
\qquad (5.13)
$$

$$J = trace\left(CYC^T\right), \qquad (5.14)$$

Where Y is the solution to the Lyapunov equation

$$AY + YA^T + BWB^T = 0 \qquad (5.15)$$

Using this new dual formulation in the LTI system,

$$J = trace\left(C_z Y C_z^T\right) \qquad (5.16)$$

Where the Lyapunov equation is given by

$$\left(A + FC_y\right)Y + Y\left(A + FC_y\right)^T + \left(B_w + FD_{yw}\right)W\left(B_w + FD_{yw}\right)^T = 0 \qquad (5.17)$$

If $B_w W D_{yw}^T = 0$, $D_{yw} W D_{yw}^T > 0$

The above Lyapunov equation becomes

$$\left(A + FC_y\right)Y + Y\left(A + FC_y\right)^T + B_w W B_w^T + FD_{yw} W D_{yw}^T F^T = 0 \qquad (5.18)$$

By completing the squares to obtain the ARE

$$AY + YA^T - YC_y^T\left(D_{yw} W D_{yw}^T\right)^{-1} C_y Y + B_w W B_w^T = 0 \qquad (5.19)$$

And the associated optimal gain

$$F = -YC_y^T \left(D_{yw} W D_{yw}^T \right)^{-1} \tag{5.20}$$

5.2 Double Pendulum

We have the linear state space system from Eq (4.12)

$$\begin{aligned} \dot{X} &= AX + Bu \\ Y &= CX \end{aligned} \tag{5.21}$$

The process noise variances and the measurement noise variances are structured in a Diagonal matrix W_a.

$$w = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \end{bmatrix} \qquad W_a = \begin{bmatrix} w_1 & 0 & 0 & 0 \\ 0 & w_2 & 0 & 0 \\ 0 & 0 & w_3 & 0 \\ 0 & 0 & 0 & w_4 \end{bmatrix} \tag{5.22}$$

These variances are augmented to the Linear system in Eq (5.21), so we get the noise augmented system

$$\begin{aligned} \dot{X} &= AX + B_w w \\ Y &= C_w X + D_w w \end{aligned} \tag{5.23}$$

We can now calculate the optimal state feedback gain F $(u = -Fx)$, by minimizing the cost function

$$J = \int_0^\infty x^T Qx + u^T Ru \tag{5.24}$$

Where $Q = B_w W_a B_w^T$ and $R = D_w W_a D_w^T$. (5.25)

The new filtered system can now be formulated as

$$\begin{aligned} \dot{\hat{X}} &= A_f \hat{X} + B_f w \\ \hat{Y} &= C_f \hat{X} + D_f w \end{aligned} \tag{5.26}$$

Where

$$\begin{aligned} A_f &= A - F^T C_w \\ B_f &= F^T \\ C_f &= C_w \\ D_f &= 0 \end{aligned} \tag{5.27}$$

The filtered linear system is now simulated on Simulink with varying input torques on the two links. Additional measurement noise is also added in the model.

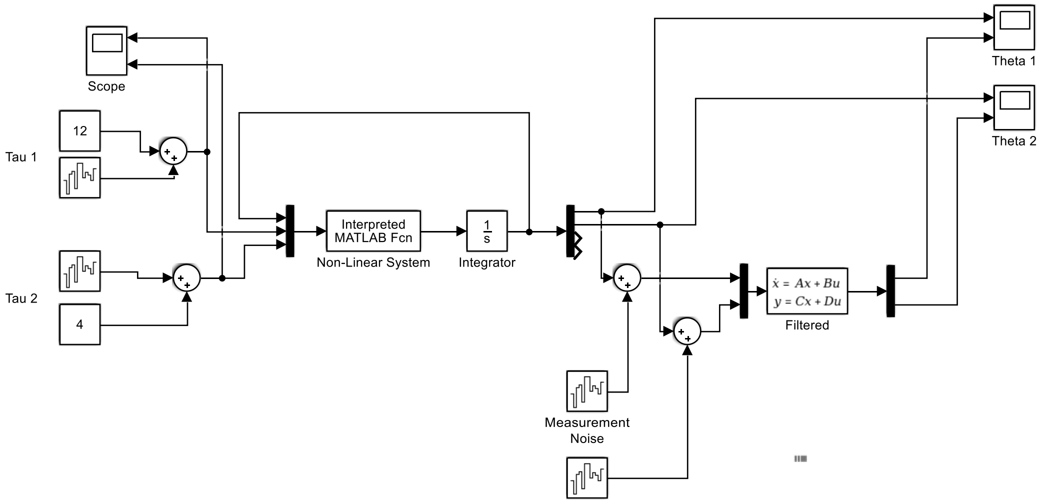

Figure 5.1 Double Pendulum Simulation

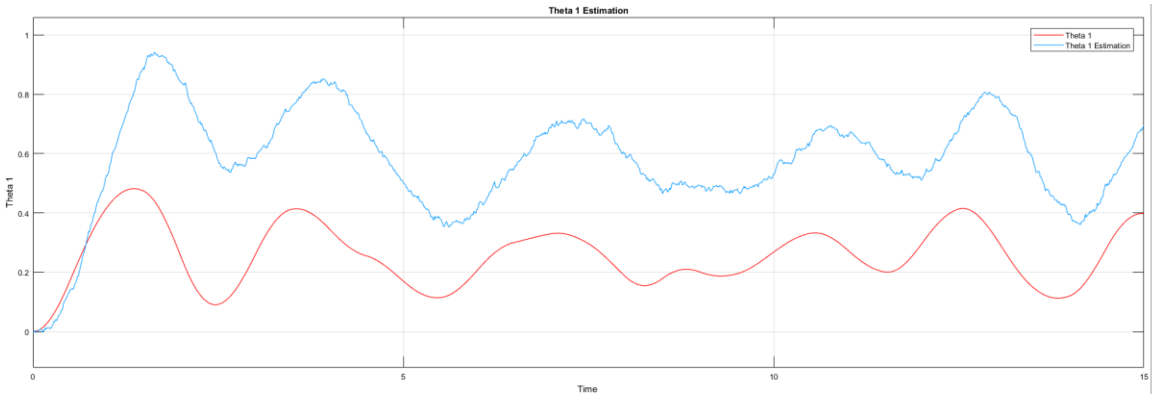

Figure 5.2 Theta 1 Estimation

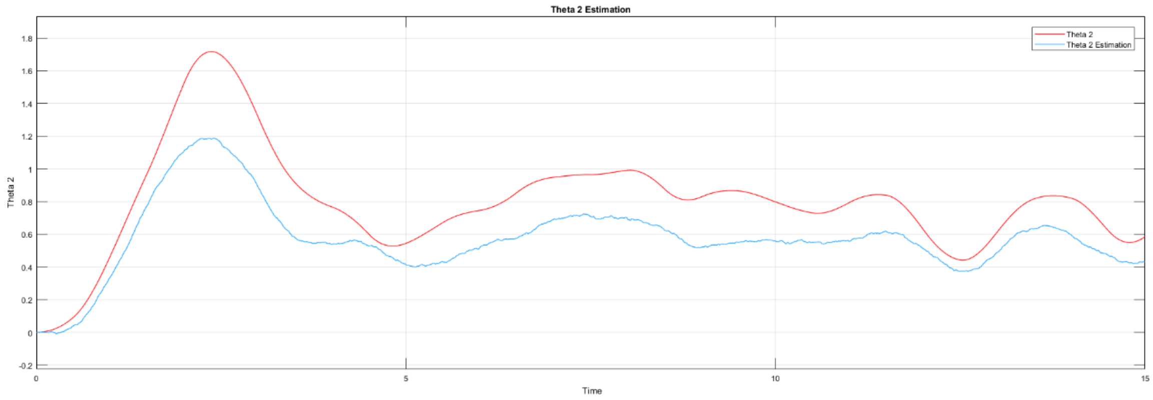

Figure 5.3 Theta 2 Estimation

From the simulation result shown in Fig--- we can see that the estimated values of θ_1 and θ_2 do not accurately track the values of the non-linear plant. This is due to the model having no information of the torque applied on the system.

The next step would be to include the torques acting on the system as part of the state. But the problem here is that we have no way of predicting the dynamics of the torque applied on the links since it is a completely random function. The best way to look at this problem would be to observe that the rate of change of torque is very slow compared to the other states namely the angle and velocity. Considering this the torques can be modelled as a constant torque acting on the links in the linear system.

So the new linear system is given by

$$\begin{aligned} \dot{\tilde{X}} &= A\tilde{X} + B_w w \\ Y &= C_w \tilde{X} + D_w w \end{aligned} \tag{5.28}$$

Where

$$\tilde{X} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \dot{\theta}_1 \\ \dot{\theta}_2 \\ \tau_1 \\ \tau_2 \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \\ x_5 \\ x_6 \end{bmatrix} \tag{5.29}$$

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ -6.2210 & 0.9571 & -0.4878 & 0.1951 & 0.4878 & -1.1707 \\ 4.7854 & -5.2639 & 0.6829 & 1.0732 & -0.6829 & 2.4390 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix}$$

$$B_w = \begin{bmatrix} 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix} \quad C_w = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix} \quad D_w = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.30}$$

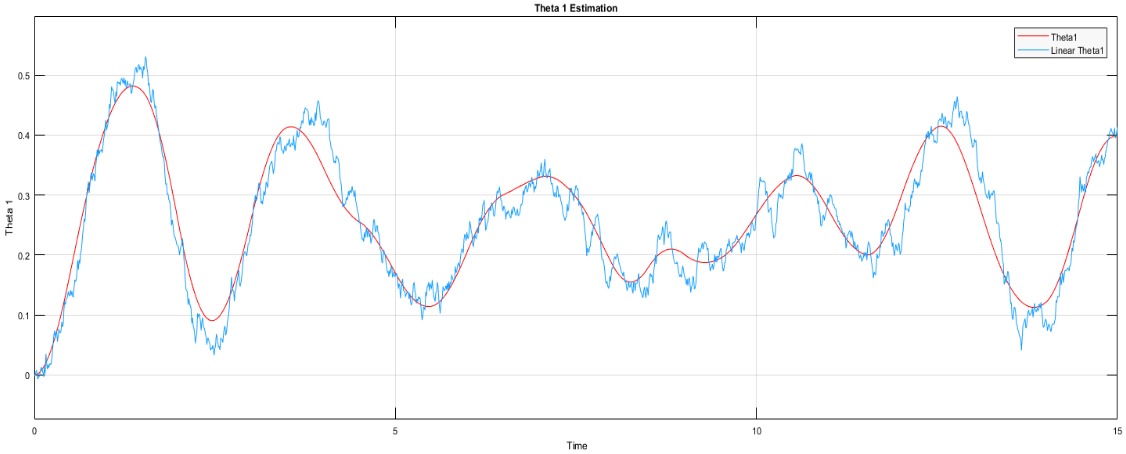

Figure 5.4 Theta 1 Estimation with Torque model

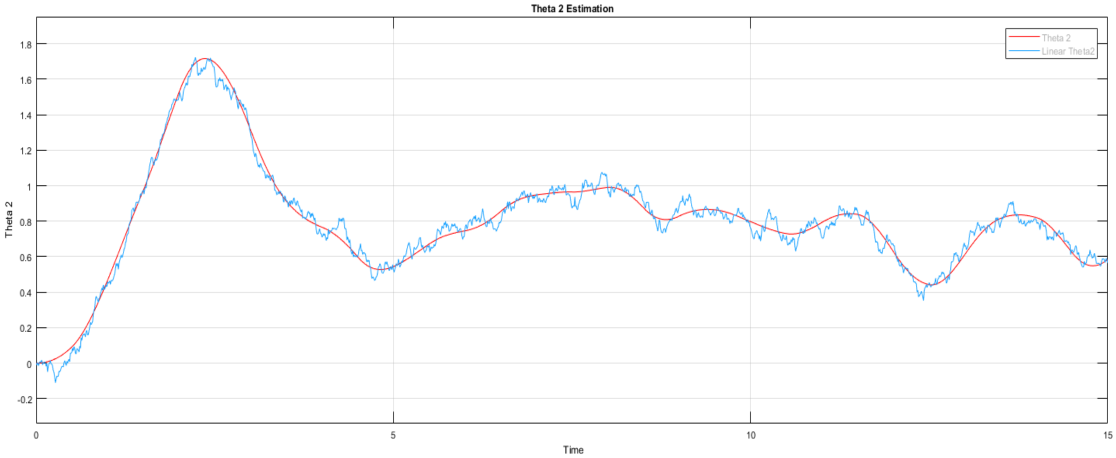

Figure 5.5 Theta 2 Estimation with Torque model

5.3 Spherical Pendulum

We have the linear state space system from Eq (4.22)

$$\begin{aligned} \dot{X} &= AX + Bu \\ Y &= CX \end{aligned} \tag{5.31}$$

The noise variances and the measurement noise variances are structured in a Diagonal matrix W$_a$.

$$w = \begin{bmatrix} w_1 \\ w_2 \\ w_3 \\ w_4 \end{bmatrix} \qquad W_a = \begin{bmatrix} w_1 & 0 & 0 & 0 \\ 0 & w_2 & 0 & 0 \\ 0 & 0 & w_3 & 0 \\ 0 & 0 & 0 & w_4 \end{bmatrix} \tag{5.32}$$

These variances are augmented to the Linear system in Eq (5.31), so we get the noise augmented system

$$\begin{aligned} \dot{X} &= AX + B_w w \\ Y &= C_w X + D_w w \end{aligned} \tag{5.33}$$

This linear model is used to estimate the state of the non-linear model. But first the torques must be augmented into the state. By including the torques in the state the states can be tracked for a non-zero mean torque input. So, the new system is given by

$$\begin{aligned} \dot{\tilde{X}} &= A\tilde{X} + B_w w \\ Y &= C_w \tilde{X} + D_w w \end{aligned} \tag{5.34}$$

Where

$$\tilde{X} = \begin{bmatrix} \theta \\ \dot{\theta} \\ \phi \\ \tau_1 \\ \tau_2 \end{bmatrix} \tag{5.35}$$

Similar procedure as the previous section is followed to find an optimal gain F. The filtered system is formulated using the matrices A_f, B_f, C_f, D_f. The system is then simulated on Simulink with varying input torques on the two directions.

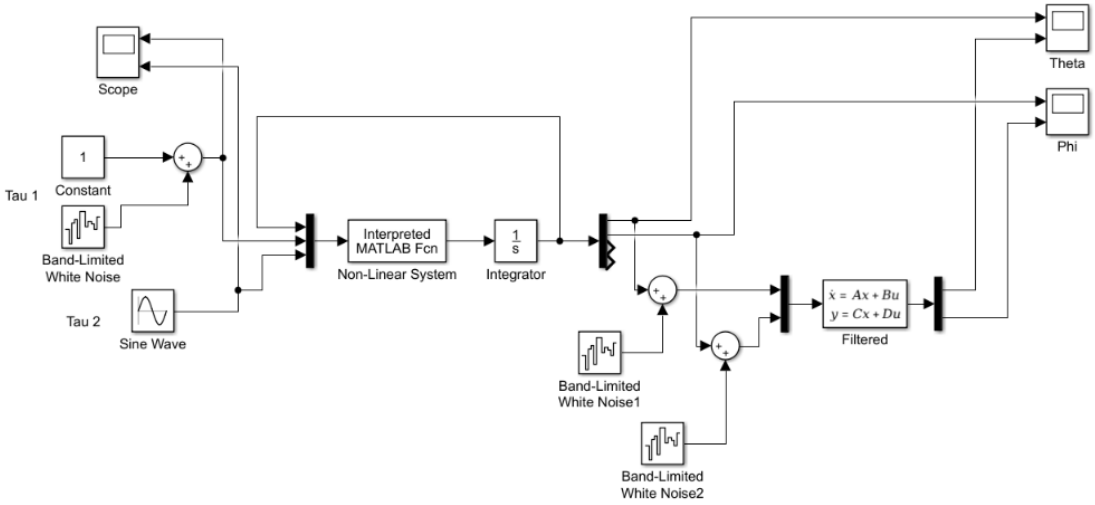

Figure 5.6 Spherical Pendulum Simulation

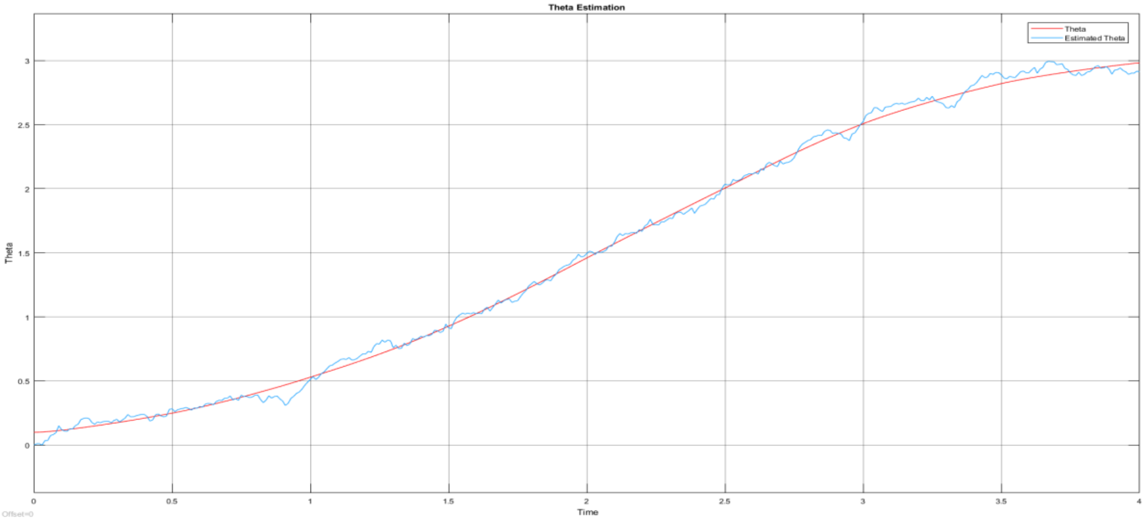

Figure 5.7 Theta Estimation

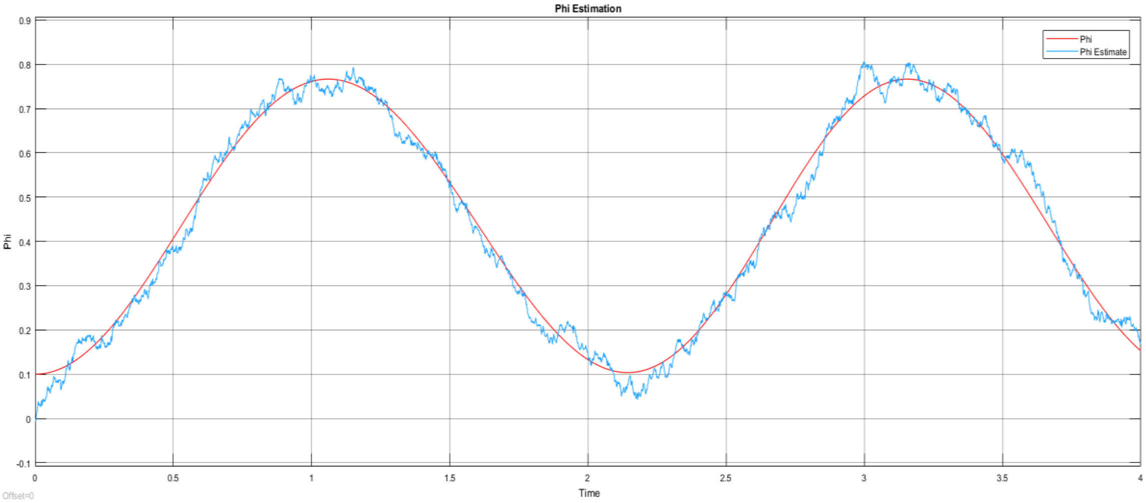

Figure 5.8 Phi Estimation

5.4 Spherical Double Pendulum

We have the linear state space system from Eq (4.40)

$$\begin{aligned}\dot{X} &= AX + Bu \\ Y &= CX\end{aligned} \tag{5.36}$$

The noise variances and the measurement noise variances are structured in a Diagonal matrix W$_a$.

$$w = \begin{bmatrix} w_1 \\ w_2 \\ \vdots \\ w_8 \end{bmatrix} \qquad W_a = \begin{bmatrix} w_1 & 0 & 0 & 0 \\ 0 & w_2 & 0 & 0 \\ 0 & 0 & \ddots & 0 \\ 0 & 0 & 0 & w_8 \end{bmatrix} \tag{5.37}$$

These variances are augmented to the Linear system in Eq (5.36), so we get the noise augmented system

$$\begin{aligned}\dot{X} &= AX + B_w w \\ Y &= C_w X + D_w w\end{aligned} \tag{5.38}$$

The performance of the system can be improved by augmenting the driving torques in the state of the system. So, the new system will have 10 states. To calculate the optimal filter the $\hat{B}$ matrix is augmented by the noise variances w$_1$ through w$_8$. The new system is denoted as

$$\begin{aligned}\dot{\tilde{X}} &= A\tilde{X} + B_w w \\ Y &= C_w \tilde{X} + D_w w\end{aligned} \tag{5.39}$$

Where

$$\tilde{X} = \begin{bmatrix} \theta_1 \\ \theta_2 \\ \dot{\theta}_1 \\ \dot{\theta}_2 \\ \phi \\ \alpha \\ \tau_1 \\ \tau_2 \\ \tau_3 \\ \tau_4 \end{bmatrix} \tag{5.40}$$

Similar procedure as the previous section is followed to find an optimal gain F. The filtered system is formulated using the matrices A_f, B_f, C_f, D_f. The system is then simulated on Simulink with input torques on the three directions.

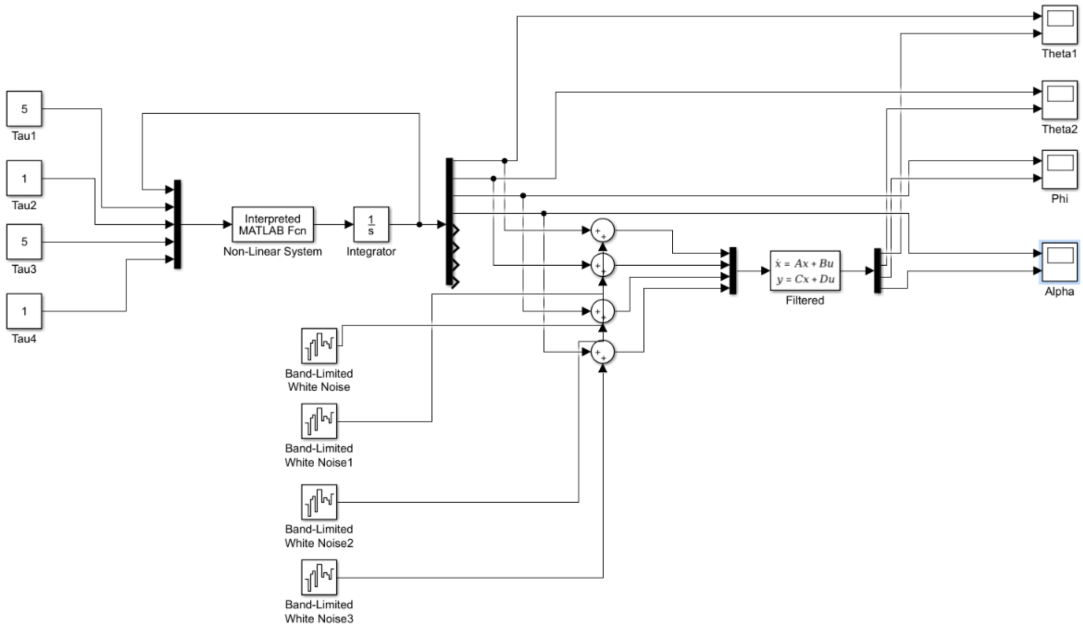

Figure 5.9 Spherical Double Pendulum Simulation

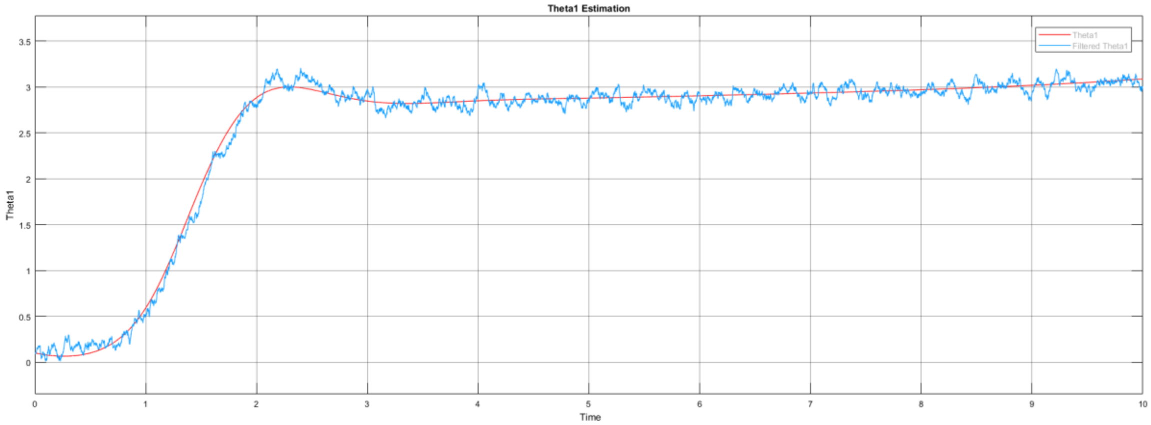

Figure 5.10 Theta 1 Estimation

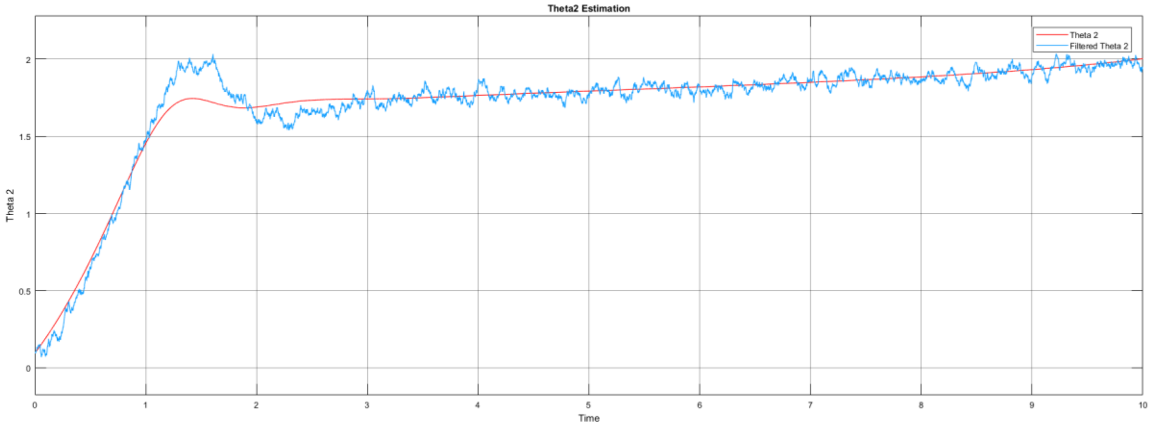

Figure 5.11 Theta 2 Estimation

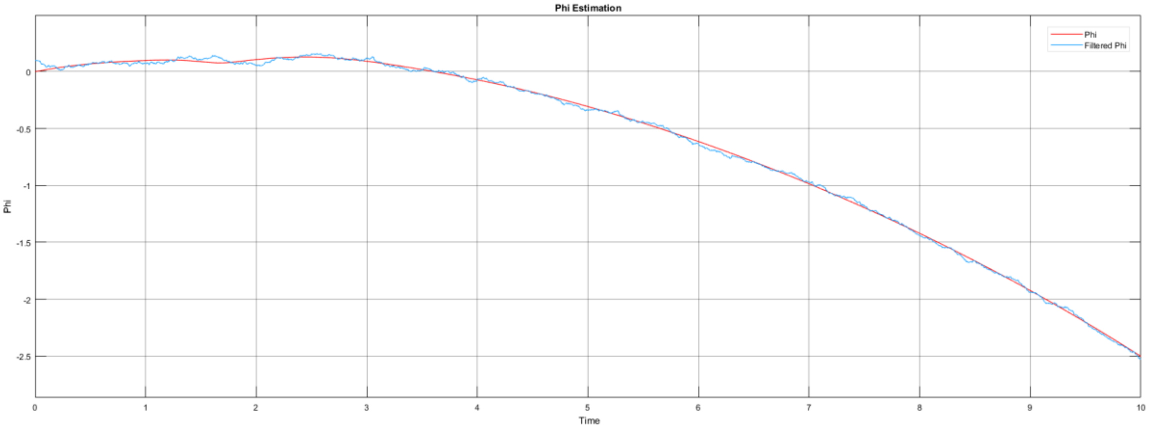

Figure 5.12 Phi Estimation

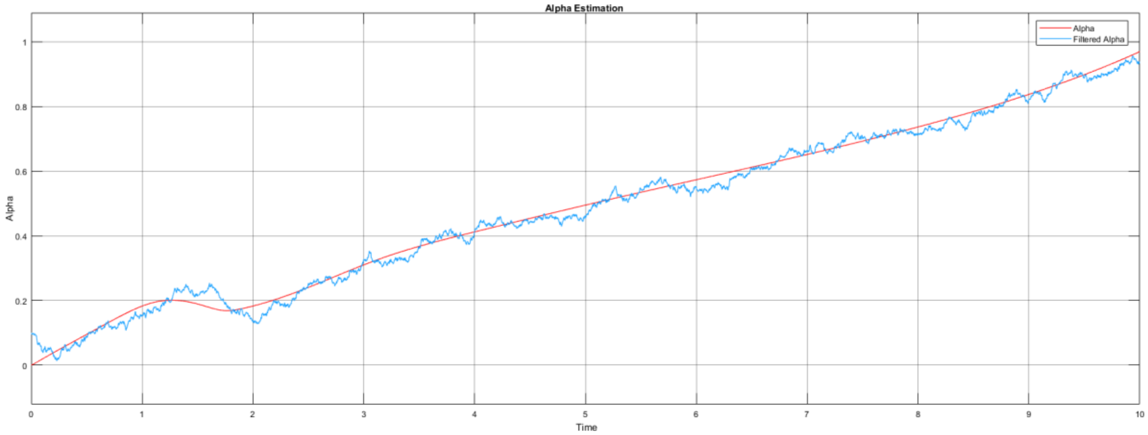

Figure 5.13 Alpha Estimation

Chapter 6

Conclusion

The torques applied to move and rotate the arm as is moves cannot be measured. So by adding the torque as a state variable and modelling it as a constant with a small variance, the linearized model has some information of the torques. This approximation is good for a low frequency change in torque, and helps build an accurate estimator.

The Kalman filter is a powerful tool in state estimation. The estimation of the attitude of the human arm using Euler angles is quite accurate for conditions of the arm that are away from the equilibrium positions. But due to the phenomenon of gimbal lock at equilibrium, it would be advantageous to use quaternions to define the vectors.

The hardware architecture developed provides a light weight and cheaper alternative to existing tracking systems. The programs written for the Arduino and ESP-8266 allow for remote control of the hardware through a Python script running on a PC. The ability of this system to be usable in any environment without the setup of cameras makes it a very capable tool. The change to low energy Bluetooth can improve the power consumption of the system as compared to Wi-Fi.

Bibliography

[1] Dan Simon, “Optimal State Estimation”, 2006.

[2] Brian D.O Anderson and John B Moore, “Optimal Filtering” 1979

[3] Breckenridge, W.G., “Quaternions - Proposed Standard Conventions," JPL, INTEROFFICE MEMORANDUM IOM 343-79-1199, 1999.

[4] Sheng Shen, He Wang, Romit Roy Choudhury, “I am a Smartwatch and I can Track my User’s Arm”, MobiSys’16, Singapore

[5] Zhang Y., Ma L., Zhang T., Xu F. (2011) Tracking of Human Arm Based on MEMS Sensors. In: Li D., Liu Y., Chen Y. (eds) Computer and Computing Technologies in Agriculture IV. CCTA 2010. IFIP Advances in Information and Communication Technology, vol 345. Springer, Berlin, Heidelberg