ACTIVE VIBRATION CONTROL OF A ROTATING FLEXIBLE LINK USING FUZZY LOGIC

COMMANDE ACTIVE DES VIBRATIONS D'UNE MEMBRURE FLEXIBLE ET ROTATIVE À L'AIDE DE LA LOGIQUE FLOUE

A Thesis Submitted to the Division of Graduate Studies

of the Royal Military College of Canada

by

Andrew Edward Lebrun

Captain

In Partial Fulfillment of the Requirements for the Degree of

Master of Applied Science in Mechanical Engineering

June 2023

Acknowledgements

I would like to thank my thesis advisor, Dr. Amor Jnifene for his insight and support throughout the duration of the thesis. Dr. Jnifene was especially helpful at refining and developing the topic of research and providing insight into all facets of control systems.

I would also like to thank the technicians of the Mechanical Engineering department. In particular I would like to thank Dave Neumann for his help and expertise with the electrical components required for the research and Brendan Freeman for ensuring that I had the equipment required to complete my research.

Abstract

This research is about active vibration control of flexible structures. Many machine elements are overdesigned to be very rigid to avoid any damaging structural vibrations, but for certain machines like industrial robots, this overdesign often leads to heavy elements requiring more powerful motors at the joints. The overall weight becomes even more important when these structures are to be used in space or aviation applications. Such structures include the Space Station Remote Manipulation System (SSRMS), and the rotor blades in rotary wings to name a few. Structural vibration within these structures can cause unwanted results during use. For the Space-Station Remote Manipulator System high precision is crucial and in helicopter tail rotors vibration can be at best uncomfortable for the occupants and at worst dangerous. One way to reduce the weight of these structures and at the same time reduce any excessive structural vibration is through Active Vibration Control.

The objective of this research is to contribute to the area of Active Vibration Control of flexible structures through the design and implementation of an Active Vibration Control system to control the tip deflection of a single, flexible link on a rotating base. The proposed Active Vibration Control method is based on the theory of Fuzzy Logic. The Fuzzy Logic based controller has two inputs; the angular position error of the base angle and the end-point deflection. The first four modes of vibration were analyzed. The main advantage of the Fuzzy Logic Controller is its non-dependence on the dynamic model of the system under control.

The Fuzzy Logic Controller was tested using two models from MATLAB Simulink and MATLAB Simscape as well as experimentally. Using the MATLAB Simulink model, the Fuzzy Logic Controller was able to reduce the link's maximum tip deflection by 42.64%

with a $\pi/32$ desired angular position and was able to reduce the vibration in the link to less than 5 mm 17.99% faster than the link under proportional control and using the MATLAB Simscape model, the Fuzzy Logic Controller was able to reduce the link's maximum tip deflection by 61.64% with a $\pi/2$ desired angular position.

In the experimental verification, the vibration of the link was able to be reduced by up to 85.73%. This controller was then able to be applied to a second link with different dimensions without creating a model of the system and showed up to an 81.65% reduction in tip deflection with no steady-state error compared with a proportional controller's 0.0031 to 0.0092 rad steady state error.

Résumé

Cette recherche porte sur le commande active des vibrations dans les structures flexibles. De nombreux éléments de machine sont surdimensionnés pour être très rigides afin d'éviter les vibrations structurelles dommageables, mais pour certaines machines comme les robots industriels, cette surconception conduit souvent à des éléments lourds nécessitant des moteurs plus puissants au niveau des articulations. Le poids total devient encore plus important lorsque ces structures doivent être utilisées dans des applications spatiales or aéronautiques. Ces structures comprennent le Télémanipulateur (SSRMS en anglais, pour Space Station Remote Manipulator System) et les pales de rotor des voilures tournantes, pour n'en nommer que quelques-unes. Les vibrations au sein de ces structures spatiales peuvent entraîner des résultats indésirables lors de l'utilisation. Pour le télémanipulateur SSRMS une haute précision est cruciale et les rotors d'un hélicoptère, les vibrations peuvent être au mieux inconfortables pour les occupants et au pire dangereuses. Une façon de réduire le poids de ces structures et en même temps réduire toute vibration structurelle excessive consiste à utiliser une commande active des vibrations.

L'objectif de cette recherche est de contribuer au domaine de commande active des vibrations des structures flexibles à travers la conception et la mise en œuvre d'un système de commande pour contrôler la position de l'extrémité d'une membrure flexible sur une base rotative. La méthode proposée pour la commande active des vibrations est basée sur la théorie de la logique floue. Le contrôleur basé sur la logique floue a deux entrées; l'erreur de position angulaire de l'angle de base et la déviation du point final. Le principal avantage de la commande par logique floue est sa non-dépendance au modèle dynamique du système sous contrôle.

Le correcteur à logique floue a été vérifié à l'aide de deux modèles de MATLAB

Simulink et MATLAB Simscape ainsi qu'expérimentalement. À l'aide du modèle MATLAB Simulink, le correcteur à logique floue a pu réduire la déflexion maximale de la pointe de la membrure flexible de 42,64% avec une position angulaire souhaitée de $\pi/32$ et a pu réduire la vibration dans la membrure flexible à moins de 5 mm 17,99% plus rapide que la membrure flexible sous commande proportionnel et en utilisant le modèle MATLAB Simscape, le correcteur de logique floue a pu réduire la déviation maximale de la pointe de la membrure flexible de 61,64% avec une position angulaire souhaitée de $\pi/2$.

Lors de la vérification expérimentale, la vibration de la membrure flexible a pu être réduite jusqu'à 85,73%. Ce correcteur a ensuite pu être appliqué à une deuxième membrure flexible de dimensions différentes sans créer de modèle du système et a montré jusqu'à 81,65% de réduction de la déviation de la pointe sans erreur en régime permanent par rapport aux 0,0031 à 0,0092 radians d'erreur en régime permanent d'un correcteur proportionnel.

Contents

List of Figures

List of Tables

Nomenclature

Symbol	*Description*	*Units*
Δ	Logarithmic Decrement	-
δ	End Point Deflection	m
ζ	Damping Coefficient	-
θ	Base Angle	rad
θ_x	Non-Collocated Rigid-Body Angle	rad
ν	Membership Function	-
ρ	Mass Density	kg/m^3
τ	Torque	Nm
ω	Frequency of Vibration	rad/s
A	Cross Sectional Area	m^2
$[C_D]$	Damping Matrix	Ns/m
E	Young's Modulus	N/m^2
G	Shear Modulus	Pa
I	Moment of Inertia	m^4
$[I]$	Identity Matrix	-
J_{base}	Mass Moment of Inertia of Motor	kgm^2
$[K]$	Stiffness Matrix of Link	N/m
$[k]$	Stiffness Matrix of Link Element	N/m
K_p	Proportional Gain	-
$\mathcal{L}$	Lagrange	-
L	Total Length of Link	m
l	Length of Link Element	m
$[M]$	Mass Matrix of Link	kg

$[m]$	Mass Matrix of Link Element	kg
N	Number of Windows in Encoder Disk	-
n_{pulses}	Number of Pulses from Encoder	-
P	Potential Energy	J
Q	Generalized Force	N
q	Generalized Coordinates	-
U	Applied Torque Vector	Nm
u	Position On Universe of Discourse	-
V	Kinetic Energy	J
w	Vertical displacement	m
x_n	n^{th} Peak in Amplitude	m

1 Introduction

Vibration affects flexible links in use all over the world. These must be modelled as long and flexible as they are not inflexible. From manufacturing robots to the Space-Station Remote Manipulator System (SSRMS) seen in Figure 1.1, also known as the CANADARM2, attached to the International Space Station, high precision is crucial. This precision is compromised largely due to lateral vibration in the link. Other links also require minimized vibrations including helicopter tail rotors, where vibration can be at best uncomfortable for the occupants and at worst dangerous. The vibrations that occur within these links are largely caused by the flexible joint when the motor acts upon it and the flexible nature of the link itself as it is considerably longer than it is thick. This research aims to reduce the vibration within a flexible link using a Fuzzy Logic Controller.

The motivation for this research is that there are many types of links in use all over the world of different sizes and made of different materials. Conventional controllers that must be designed specifically for the individual link are not capable of being implemented in large scale capacities all over the world without engineers and technicians putting in considerable time to complete the analysis and design for every link in use. However, by developing a controller using Fuzzy Logic, it will greatly reduce the amount of design work required to implement Active Vibration Controllers across a wide range of flexible links.

The focus of this research is to develop an Active Vibration Controller to minimize the vibration of a flexible link based on tools of Fuzzy Logic that does not depend on the model of the system being calculated. This research is organised as follows: the second chapter is the literature review which will focus on explaining the general background of the topics that are required to be understood within this topic. The third chapter will show the derivation and verification of a dynamic model of the system that will be explored so that it may be

used to conduct testing within simulations. The fourth chapter will carry on with the actual vibration control of the system. This includes everything from the design to the simulation analysis of the Fuzzy Logic Controller with optimization of the controller. The fifth chapter will contain the experimental verification of the Fuzzy Logic Controller compared with a proportional controller. Finally, the sixth chapter will contain a summary of conclusions along with areas of potential future research related to this topic.

Figure 1.1: Space-Station Remote Manipulator System [1]

2 Literature Review

This chapter will cover the background information about the key topics explored within this research as well as previous research that has been conducted in that field. The topics that are discussed include how to develop a dynamic model of a flexible link, conventional controllers and their usage in controlling vibration in flexible links, Fuzzy Logic Control, and vibration analysis. Each of these topics will be important to understand prior to designing a Fuzzy Logic Controller for a flexible link.

2.1 Dynamic Modelling of Flexible Structures

One of the first steps in designing a controller for a dynamic system is to create a dynamic model of the system so that the controller can be tested, and its effectiveness verified through simulations before moving into physical testing. There are several methods of determining the mathematical representation for a flexible link.

Two of the most common methods used to derive the dynamic equations are the Finite Element Method (FEM) (also referred to as Finite Element Analysis (FEA)) and the Assumed Modes Method (AMM). The FEM involves sectioning the link into multiple elements connected at nodes that have their own shape functions and boundaries [2, 3]. This makes FEM especially useful when links are segmented with different materials, cross-sections, or any other key properties which make the segmentation very intuitive. This is normally used with a Lagrange approach to determine the equations of motion for the whole link [2, 3].

In contrast with FEM, the AMM is used by calculating the shape functions (also called mode functions) and the time function of the entire link with the boundary conditions considered [2–4]. This model, similar to FEM, is normally combined with the Lagrange

approach to determine the equations of motion of the system [2–5].

Simulations completed by both Reddy and Sudhudi [4, 5] have shown the effectiveness of the AMM method when combined with the Lagrange approach for modelling a flexible link with a flexible joint. Theodore [3] took this further and actively compared the effectiveness of models with FEM and AMM. They found that FEM is especially useful for links with changing cross-sections or systems of multiple links whereas AMM is better suited for single link systems with uniform cross-sections. AMM is also better suited where high precision is required due to the FEM being a more simplified method because it is a lower order system (finite order system versus continuous system) which can affect the precision of the model.

However, while the AMM is more accurate than the FEM, FEM requires fewer computations within the computer which makes it better suited for real-time controller implementations [3]. In order to ensure accuracy within FEM, the AMM can be used to calculate the natural frequencies of the link and then be used to verify that the natural frequencies of the FEM model are accurate.

2.2 Conventional Active Vibration Control

Controllers are used to ensure the output of a given system matches the desired output. There are two types of systems, an open-loop system where the output is not able to be measured meaning that there is no mechanism to eliminate the error of the desired output of the system, thus changing the output and a closed-loop system as seen in Figure 2.1. This research will focus on implementation of a controller within a closed-loop system. Closed-loop systems have a sensor that can determine what the actual output of the system is and inform the controller so that the controller is able to adjust the output of the system [6].

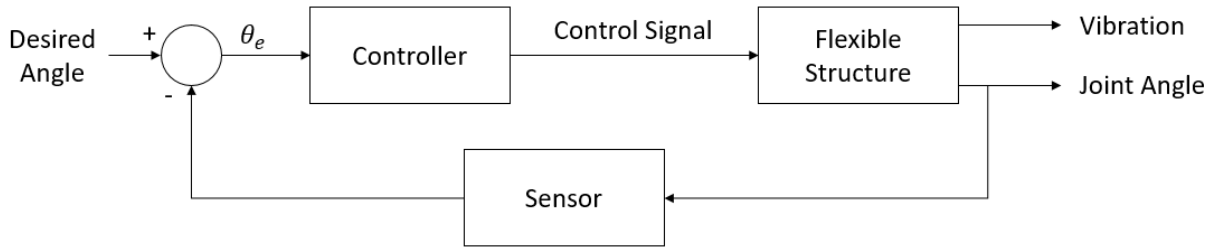

Figure 2.1: Closed Loop Flexible Link Control System

Conventional controllers used within these systems typically fall into several categories. Four of these are proportional (P) controllers, proportional-derivative (PD) controllers, proportional-integral (PI) controllers, and proportional-integral-derivative (PID) controllers [6]. While there are a wide variety of uses for these controllers, each has some limitations. The proportional term of the controller is used to move the output to the desired value of the system. The derivative term of the controllers are added to increase damping which is used to reduce the time required to reach the steady state response and minimize the transient response of the system. Finally, the integral term of the controller is used to eliminate the steady-state error in the system. However, these controllers have three gain values that need to be adjusted and optimized for each use. This means that a controller must be individually designed for each system and if that system is altered in any way, the controller must be redesigned as well. This can be time consuming to constantly have to redesign the controller [6].

Researchers have continued research on conventional controllers including Le [7] who used a PD controller and linear quadratic regulator state feedback to reduce vibration of a flexible cantilever link. They focused on controlling the first three mode shapes within the link. They achieved significant vibration reduction when the controller was added to the system. However, one of the key drawbacks of their research was that the frequency response of the physical system was not obtained through experiments and was estimated

to be employed within the analytical model. This was due to limitations with the sampling rate of the hardware they were using. This caused some minor error within the controller and in order to eliminate it, the frequency response of the physical system would have to be experimentally determined and implemented.

Manning et al. [8] utilized a linear Auto Regressive Moving Average eXogenous inputs (ARMAX) model for smart structures such as piezoceramics to adjust the closed-loop pole placement on links to enhance the damping effects on a cantilever link. This was done by attaching piezoactuators along a link, determining the desired closed-loop pole placement for the system, and applying a voltage to the piezoactuators to change the closed-loop transfer function of the link system, increasing damping. This was able to reduce vibration while reducing the required sampling rate, however this was prone to limit cycle issues and a limited number of modes able to be analyzed and controlled.

Some work that looked to control a flexible link with a rotating joint was produced by Jnifene and Fahim [9] where a time delay controller was used to minimize vibration at the end point of the link. They were able to accomplish this by using the angle of the joint and the delayed deflection of a point along the link. They found that they were unable to control the endpoint of the link when collocating the actuator with the joint sensor without the time delay. However, by adding the time delay and using a non-collocated sensor to measure the deflection partway along the link with a Computed Torque/Delayed Deflection controller, simulation results showed that the controller was able to effectively control the vibration of the end point of the link.

Much of the previous research though has focused on cantilever links and ignored any flexibility within the joint. Dong [2] was able to expand on the previous research by Jnifene and Fahim to develop some of the first work that took the flexible joint into consideration

with a flexible link and flexible joint time delay controller. They used the Assumed Mode Method with a Lagrange approach to develop a model of the system and then applied a time delayed feedback signal controller paired with a PD controller to affect the vibration. They were able to greatly reduce the vibration in both the joint and the link but noted that the higher the mode of the system, the more it was affected by the relationship between the link and the joint.

2.3 Active Vibration Control Based in Fuzzy Logic

Fuzzy Logic is a mathematical framework that allows for the representation of uncertainty and imprecision in data and information. It was first proposed in 1965 by Lotfi Zadeh [10] in which they proposed the concept of Fuzzy Sets as a way to consider classes of objects that contain members that are not neatly described in the conventional, binary terms as either belonging to a class or not. They recognized that in some cases, the status of objects within particular classes was ambiguous but not due to lack of information, but rather due to the imprecision or vagueness in the language used [11]. In order to represent the uncertainty of these components of the set, a degree of membership which ranges from 0 to 1 is assigned to each member of the set. For example, if the fuzzy set is representing temperature, the variables in question could include "hot", "cold" or "warm" with each temperature having a value between 0 and 1 for each variable based on how well they fit each description. Of note, a temperature can have a non-zero value for more than one variable.

While some research into the theory behind Fuzzy Sets continued through the 1960s and into the early 1970s, it wasn't until 1974 when the concept of Fuzzy Logic was first able to be applied to a controller by Mamdani [12] who used Fuzzy Sets to control a small steam engine. Fuzzy Logic Controllers were still not widely used after this however until the 1980s when Japan started using Fuzzy Logic Controllers in commercial products which

began to see an increase in use of the Fuzzy Logic Controllers worldwide [11].

Fuzzy Logic Control is an if_then_else rule based system that can control the outputs of a system. A Fuzzy Logic Controller with all its components can be seen in Figure 2.2. Fuzzy Logic Control utilizes linguistic rules to control a system. This allows for the controller to adjust based on subjective differences in linguistic definitions. This is completed with its four main components: a Fuzzifier, a Rule Base, the Computational Unit, and the Defuzzifier [11]. The Fuzzy Logic Controller works by taking crisp inputs and using membership functions to convert them into values between 0 and 1 along distinct categories corresponding to a linguistic description of that value. The Rule Base of the system then identifies which inputs and outputs are being affected within the system. Next the Computational Unit calculates, based on the values of the input membership functions and the Rules Base, where on the output membership functions the output lies as part of a Fuzzy Set. Finally, the Defuzzifier converts that output Fuzzy Set value back into a crisp value as the output of the controller that feeds into the system. There is overlap between each membership function to account for the ambiguity of language. For example, if describing the temperature outside, 25 degrees Celsius may have a value of 0.7 warm and a value of 0.2 hot. This is because some people would describe this temperature as warm whereas others would describe it as hot and the difference in linguistics is accounted for. It is important to note that there is no requirement for the sums of the affected membership functions to add to any specific number. It is only required that the values of each individual membership function range from 0 to 1. The Fuzzy Logic Controller then uses a Rule Base that is built using the linguistic descriptions. For example, if designing a temperature controller on an air conditioner where increasing the power will make the air conditioner work more, a rule may be "if the temperature is hot, then increase power to high". There will be rules for every input possibility and as the crisp inputs will likely cover multiple membership functions due to the overlap, the Computational Unit is then able to analyse all activated rules to determine how the overlap

affects the required output. Once the Computational Unit has finished analysing the inputs, the output passes through a Defuzzifier, turning the output back into a crisp value. This allows much smoother transitions as the controller is not simply on or off, but rather can have a range of how much to affect the system. These controllers are also typically very robust as they are not dependant on the mathematical representation of the system [13].

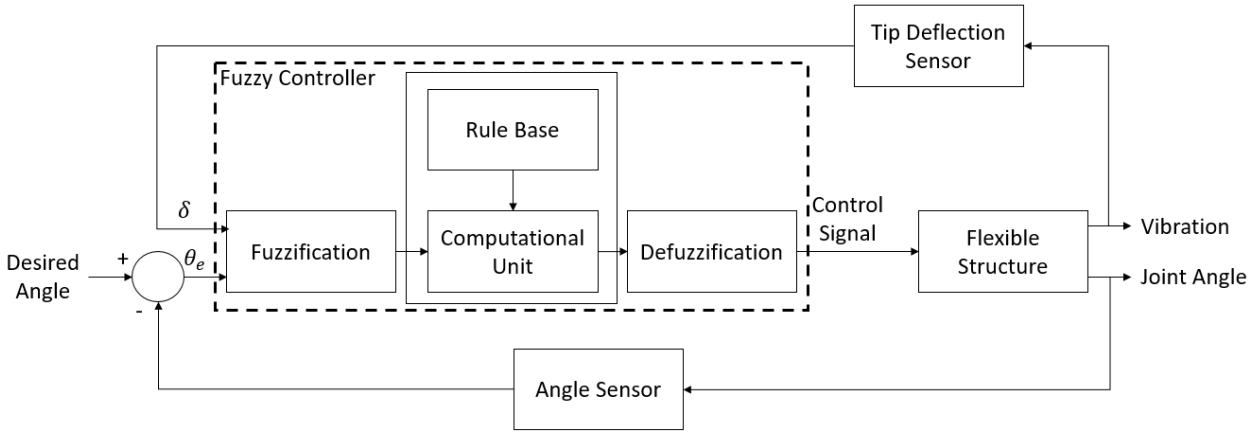

Figure 2.2: Fuzzy Logic Controller

Research in controlling the vibration of flexible links using Fuzzy Logic was conducted by Andrews [14]. They created a Fuzzy Logic Controller for dampening the vibration in a one degree of freedom flexible link. They were able to minimize the vibration of the end point of the link. While this controller was effective in its goal of minimizing vibration, it did not effectively minimize the percent overshoot of the rigid body motion. It also failed to eliminate the steady state error of the system. Andrews and Jnifene [15] later built upon their own work to further the research of end point control of Fuzzy Logic Controllers. They were able to track and control the vibrations by using strain gauges to track the end point deflection as well.

This research was then expanded upon by Fricker [13]. Fricker took Andrews' work and utilized a Fuzzy Model Reference Self Tuning Control algorithm to scale the gains of the Fuzzy Logic Controller automatically in order to track end point position while

maintaining the vibration suppression. This allowed for the controller to automatically adapt to changes in the system while minimizing the end point vibration as well. While this was an improvement from previous Fuzzy Logic Controllers, this system struggled with having a steady state error caused by the strain gauges reference position being vertical. Once the link began to move, the strain gauges would experience some strain caused by gravity acting on the link which created a steady state error. Also, the controller appeared to be less effective as the frequency and amplitude of the oscillations increased.

More recently, Sun et al. [16] continued the research in this field by comparing a Fuzzy Neural Network with a conventional PD controller to suppress the vibration of a flexible link. They first analysed the system mathematically and designed the PD and neural networks using a Lyapunov function candidate to ensure stability of the system when the controllers are implemented. Simulations were then completed that showed that while the PD controller was effective, it had a large overshoot. However, when the Fuzzy Neural Network was used, the link was controlled in less time and the overshoot was eliminated.

2.4 Vibration Analysis

In order to reduce the vibration of a flexible structure, the system must be turned into a smart structure. A smart structure is a non-biological system that can sense, control, and actuate on its own [17]. Therefore, a sensor must be used to determine how the link is vibrating. This allows the controller to determine how to reduce the vibration and then activates an actuator to reduce the vibration. The sensor is a key part of this system as having an effective sensor is critical for the controller to be able to work properly. However, adding more sensors increases the overall cost and complexity of the system, therefore, one of the goals of the sensor is to minimize the number required. This can be achieved by only focusing on the first few modes of vibration as they will have a significantly larger effect on the displacement of points along the link.

Li and Chen [18] recognized the importance of understanding the amount of end point deflection of a flexible link and attempted to determine the most effective ways to track its movement. They used a position sensor and accelerometer at the tip and an encoder at the base to measure the joint angle. They then estimated the position and velocity of the end point using two different methods: the Finite-Difference Method and the Observer Method using a Kalman Filtering Algorithm. They found that the Kalman Filtering Algorithm was considerably more effective at tracking the end point of the link as the Finite-Difference Method was unable to effectively account for noise within the sensors which produced additional error.

In order to measure the modes of vibration of a flexible link, Jehan [19] conducted an analysis of sensor placement on a flexible link. They noted that it was key in sensor placement to avoid any of the nodes of the modes that are being analysed. The general rules that they found was to determine the highest mode of vibration that is to be analysed, identify where the first maxima from the base is for the highest mode and place the sensor at or near this position while avoiding the nodes of lower modes. However, one of the key limitations of this research was that this placement is only the optimization of a single sensor placement on the link. This does still provide insight into additional sensor placement as many key factors studied here can continue to be applied, such as avoiding nodes where no vibration occurs for individual modes.

2.5 Summary

It can be seen from this literature review that while the control of flexible structures is extremely important, there is still areas of Active Vibration Control in flexible structures that require more research. While most control systems are designed utilizing conventional controllers which are effective for individual systems, effective Fuzzy Logic Controllers would

be not only efficient to implement on the system, but would also be easier to adjust for use in other flexible link systems. Therefore, there is a need to continue the research into Fuzzy Logic Controllers in flexible link systems.

3 Dynamic Model of a Single Flexible Link

This chapter focuses on the development and implementation of a dynamic model to be used in simulations to design an Active Vibration Controller. The first section within this chapter focuses on determining the natural frequencies of the system using the AMM followed by a section on FEA to determine the mass and stiffness matrices with enough elements that the natural frequencies of the model match the AMM natural frequencies. Section 3.2 then uses these matrices combined with a Lagrangian approach to develop the equations of motion of the system which can then be modelled within simulation as a state-space model. In Section 3.3 this model is verified by comparing it with the natural frequencies of the physical link that will be used for physical testing. Finally, in Section 3.4, a secondary model was designed using Simscape to be used in conjunction with the Simulink model.

3.1 Derivation of Mass and Stiffness Matrices

The system that is being analysed is a pinned-free link which rotates about its pinned joint similar to the system that can be seen in Figure 3.1. The properties of this link can be seen in Table 1.

The link is made of aluminum and is very thin compared to its length, ensuring flexibility within the link. While a Fuzzy Logic Controller is a non-model based controller, the controller will need to be verified through simulation before moving to a physical system to ensure that the design is appropriate. Therefore, the mathematical model of this link was developed so that simulation results can be obtained.

There are two common methods for deriving the dynamic equations of a flexible link system. They are the AMM and the FEA method. The key differences between the two are that the AMM calculates the natural frequencies of the link as a continuous system

whereas the FEA discretizes the link making it less accurate but easier to model.

Due to the non-model based nature of the Fuzzy Logic Controller, some accuracy is worth sacrificing for computational efficiency. However, the model should still accurately represent the system. Therefore, the AMM was used to determine the theoretical natural frequencies of the link and then the FEA method was used with increasing numbers of elements until the first four natural frequencies matched the AMM within a reasonable amount of error.

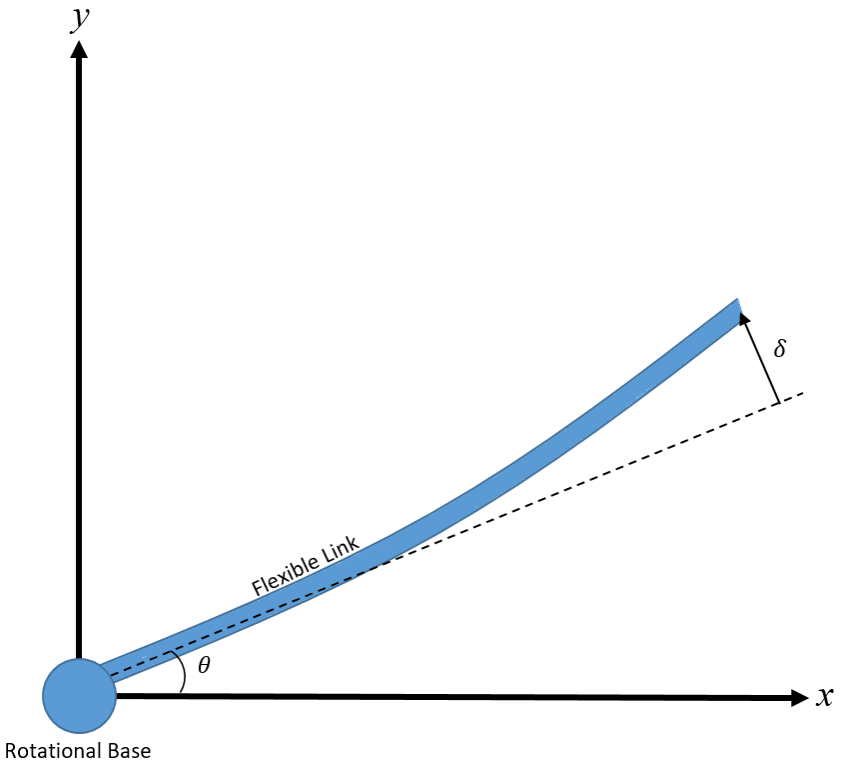

Figure 3.1: Diagram of Link System

Table 1: Properties of the Link

Length of link, L (m)	1.12
Width of link (m)	0.0508
Thickness of link (m)	0.00228
Mass density, ρ (kg/m^3)	2700
Moment of Inertia, I (m^4)	5.018*10^{-11}
Mass of link, m (kg)	0.350
Young's Modulus of Elasticity, E (Pa)	7x10^{10}

3.1.1 Assumed Mode Method

The AMM is typically the most accurate method of determining the natural frequencies of a link. This is because it treats the link as a continuous system. However, while this creates very accurate results, it is considerably more complex and requires more computational power than other methods such as FEA [3].

One way that it can be used though is to determine what the theoretical natural frequencies of the system are as it is easy to calculate the natural frequencies without developing the overall equation of the system. These frequencies can then be compared with methods that are easier to model to ensure that they are within an acceptable level of accuracy to be able to properly test the controller.

The following section shows the derivation of the system's natural frequencies using the AMM. From the Euler-Bernoulli link theory, the lateral vibration of a flexible link is governed by Equation 3.1 [20]:

$$EI\frac{\partial^4 w(x,t)}{\partial x^4} + \rho A\frac{\partial^2 w(x,t)}{\partial t^2} = f(x,t) \tag{3.1}$$

where E is the Young's Modulus, I is the moment of inertia, $w(x,t)$ is displacement at

any point along the link at any time, ρ is the density of the material, A is the cross-sectional area of the link, and $f(x,t)$ is the external forcing function acting on the link. In order to determine the natural frequencies of the link, the free vibration response of the link must be analyzed. This is done by setting the forcing function equal to zero in Equation 3.1. Rearranging Equation 3.1 gives:

$$c^2 \frac{\partial^4 w(x,t)}{\partial x^4} + \frac{\partial^2 w(x,t)}{\partial t^2} = 0 \tag{3.2}$$

where:

$$c = \sqrt{\frac{EI}{\rho A}}$$

The solution of Equation 3.2 can be obtained using the method of separation of variables as [20]

$$w(x,t) = W(x)T(t) \tag{3.3}$$

where $W(x)$ is the mode function (also called the normal function) of the link and represents all the modes of the link and $T(t)$ is the time function of the link. By substituting Equation 3.3 into Equation 3.2 gives

$$c^2 \frac{d^4 W(x)}{dx^4} T(t) + \frac{d^2 T(t)}{dt^2} W(x) = 0 \tag{3.4}$$

Collecting all terms that are with respect to x on one side of Equation 3.4 and all terms with respect to t on the other side gives

$$\frac{c^2}{W(x)} \frac{d^4 W(x)}{dx^4} = -\frac{1}{T(t)} \frac{d^2 T(t)}{dt^2} \tag{3.5}$$

Due to one side of Equation 3.5 only changing with respect to distance and the other side only changing with respect to time but with them being equal to each other, the only

possible solution for this is if both sides are equal to a constant which shall be referred to as ω^2 which gives [20]

$$\frac{c^2}{W(x)}\frac{d^4W(x)}{dx^4} = -\frac{1}{T(t)}\frac{d^2T(t)}{dt^2} = \omega^2 \tag{3.6}$$

The mode function and time functions can then be examined independently from each other where [20]

$$\frac{d^4W(x)}{dx^4} - \beta^4 W(x) = 0 \tag{3.7}$$

and

$$\frac{d^2T(t)}{dt^2} + \omega^2 T(t) = 0 \tag{3.8}$$

where

$$\beta^4 = \frac{\omega^2}{c^2} = \frac{\rho A \omega^2}{EI} \tag{3.9}$$

The solution to Equation 3.7 is given by [20]

$$W(x) = C_1 cos\beta x + C_2 sin\beta x + C_3 cosh\beta x + C_4 sinh\beta x \tag{3.10}$$

where C_1, C_2, C_3, and C_4 are constants that are determined from the boundary conditions of the link. The solution of Equation 3.8 is [20]:

$$T(t) = Acos\omega t + Bsin\omega t \tag{3.11}$$

where A and B are obtained through the initial conditions of the link. This portion of the equation of motion of the link will change with different initial conditions and will not affect the mode shapes of the link or its natural frequencies and therefore does not need to

be solved at this time.

In order to finish solving Equation 3.10, the boundary conditions at the joint can be considered as a pinned end. Therefore, the conditions that must be satisfied in the equation are that the deflection at x=0 must be equal to 0 for any time and the bending moment at x=0 must be equal to 0 for any time which is shown in Equations 3.12 and 3.13 respectively [20].

Pinned end:

$$w(0,t) = W(0)T(t) = 0 = W(0) \tag{3.12}$$

$$EI\frac{\partial^2 w(0,t)}{\partial x^2} = EI\frac{d^2W(0)}{dx^2}T(t) = 0 = EI\frac{d^2W(0)}{dx^2} \tag{3.13}$$

The boundary conditions of the free end that must be satisfied in Equation 3.10 are that the bending moment at x=L must be equal to 0 for any time and the shear force at x=L must be equal to 0 for any time which is shown in Equations 3.14 and 3.15 respectively [20].

Free end:

$$EI\frac{\partial^2 w(L,t)}{\partial x^2} = EI\frac{d^2W(L)}{dx^2}T(t) = 0 = EI\frac{d^2W(L)}{dx^2} \tag{3.14}$$

$$\frac{\partial}{\partial x}\left(EI\frac{\partial^2 w(L,t)}{\partial x^2}\right) = EI\frac{d^3W(L)}{dx^3}T(t) = 0 = EI\frac{d^3W(L)}{dx^3} \tag{3.15}$$

Substituting Equation 3.12 into Equation 3.10 gives

$$C_1 + C_3 = 0 \tag{3.16}$$

Substituting Equation 3.13 into Equation 3.10 gives

$$EI\frac{d^2W(x)}{dx^2} = EI\beta^2[-C_1cos\beta x - C_2sin\beta x + C_3cosh\beta x + C_4sinh\beta x]_{x=0} = 0 \tag{3.17}$$

Due to the pinned end being at x=0, Equation 3.17 is simplified as

$$\beta^2 EI[-C_1 + C_3] = 0 \tag{3.18}$$

As $\beta^2 EI$ is a constant term, Equation 3.18 can finally be simplified as

$$-C_1 + C_3 = 0 \tag{3.19}$$

Combining Equations 3.16 and 3.19 gives C_1=C_3=0 and therefore Equation 3.10 becomes:

$$W(x) = C_2sin\beta x + C_4sinh\beta x \tag{3.20}$$

Next, substituting Equation 3.14 into Equation 3.20 gives

$$EI\frac{d^2W(x)}{dx^2} = EI\beta^2[-C_2sin\beta x + C_4sinh\beta x)]_{x=L} = 0 \tag{3.21}$$

which is simplified as

$$-C_2sin\beta L + C_4sinh\beta L = 0 \tag{3.22}$$

Substituting Equation 3.15 into Equation 3.20 gives

$$EI\frac{d^3W(x)}{dx^3} = EI\beta^3[-C_2cos\beta x + C_4cosh\beta x)]_{x=L} = 0 \tag{3.23}$$

which is simplified as

$$-C_2 cos\beta L + C_4 cosh\beta L = 0 \tag{3.24}$$

For the non-trivial solution of C_2 and C_4 the determinant of the coefficients of Equations 3.22 and 3.24 are set to zero such that:

$$\begin{vmatrix} -sin\beta L & sinh\beta L \\ -cos\beta L & cosh\beta L \end{vmatrix} = 0 \tag{3.25}$$

Solving the determinant of Equation 3.25 gives

$$(-sin\beta L)(cosh\beta L) - (-cos\beta L)(sinh\beta L) = 0 \tag{3.26}$$

which can be rewritten as

$$-sin\beta L cosh\beta L + cos\beta L sinh\beta L = 0 \tag{3.27}$$

Knowing that $\frac{sin(x)}{cos(x)} = tan(x)$ and $\frac{sinh(x)}{cosh(x)} = tanh(x)$ Equation 3.27 can be simplified to:

$$tan(\beta L) - tanh(\beta L) = 0 \tag{3.28}$$

provided that $\beta L \neq \frac{\pi}{2} + k\pi$ where $k = 1, 2, 3...$

Equation 3.28 has multiple solutions, with each subsequent solution corresponding to the next mode of vibration. The first five values of βL are shown in Table 2.

Table 2: Solutions of βL

$\beta_n L$	**Value**
0	0
1	3.926602
2	7.068583
3	10.210176
4	13.351768

These modal values can then be used with Equation 3.9 in order to determine the natural frequencies of the flexible link as

$$\omega = (\beta_n L)^2 \sqrt{\frac{EI}{\rho A L^4}} \tag{3.29}$$

in rad/s.

For the link with the properties shown in Table 1

$$\omega = 2.6718(\beta_n L)^2 \tag{3.30}$$

By combining the values of $\beta_n L$ shown in Table 2 with Equation 3.30, the first five natural frequencies (f) of the link including the rigid body motion were calculated as seen in Table 3 where $f = \frac{\omega}{2\pi}$.

Table 3: Natural Frequencies of Link

Mode Number	**Frequency (Hz)**
0	0
1	6.56
2	21.25
3	44.33
4	75.81

Now that the first five natural frequencies of this system (including rigid body motion) have been mathematically calculated, the accuracy of the natural frequencies of the FEA mass and stiffness matrices can be determined with a varying number of elements. Lagrange methods and FEA with different numbers of elements can be used to build a model of the link. The frequencies obtained using FEA are compared with these natural frequencies in order to confirm the model is an acceptable representation of the system while minimizing the required computing power.

3.1.2 Finite Element Analysis

Another method for determining the natural frequencies of the flexible link is the FEA method. This is done by deriving the mass and stiffness matrices of the system and using those to determine the eigenvalues of the system. The mass and stiffness matrices will be able to be incorporated within software such as MATLAB to create a model of the link. However, the number of elements that will be used in the model needs to be determined first. As the number of elements in the model increases, the accuracy of the model will increase as well but this will come at the cost of increased computational power being required. In order to maximize the effectiveness of this model, two elements were used to calculate the natural frequencies and then they were compared with the results from the AMM. More elements were then added, one at a time, with the natural frequencies being recalculated and compared again until an appropriate level of accuracy was found.

Using the Euler-Bernoulli link theory, a link element, i, that is part of a link as seen in Figure 3.2 has node displacements of $w_i(t)$, $\theta_i(t)$, $w_{i+1}(t)$, and $\theta_{i+1}(t)$. The link element has a length of l and the transverse displacement at any point along the element at a position of x is given by $w(x,t)$. The transverse displacement within the element is assumed to be a cubic equation as seen in Equation 3.31 [20].

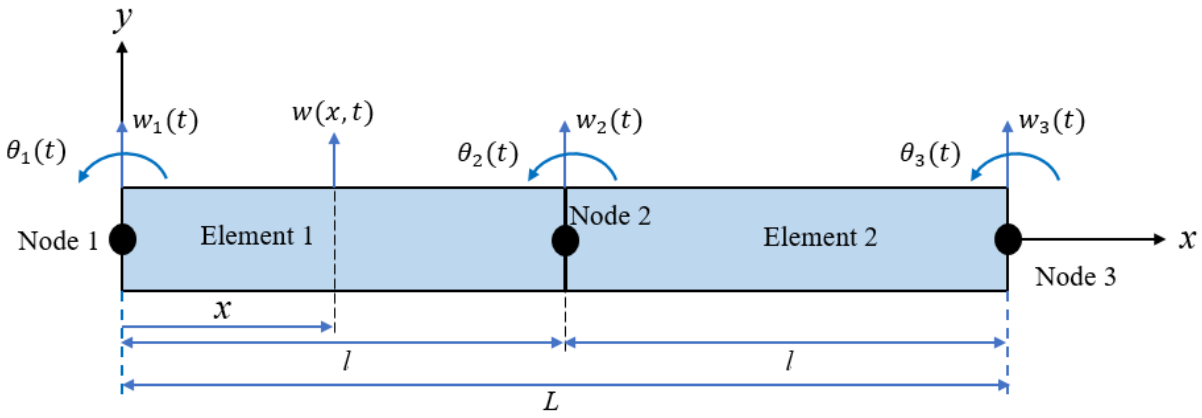

Figure 3.2: Finite Element Analysis of 2 Element Link

$$w(x,t) = a(t) + b(t)x + c(t)x^2 + d(t)x^3 \tag{3.31}$$

The displacements at each node must satisfy the following conditions [20]:

$$w(0,t) = w_i(t) \tag{3.32}$$

$$\frac{dw(0,t)}{dx} = \theta_i(t) \tag{3.33}$$

$$w(l,t) = w_{i+1}(t) \tag{3.34}$$

$$\frac{dw(l,t)}{dx} = \theta_{i+1}(t) \tag{3.35}$$

Equations 3.32 through 3.35 are then substituted into Equation 3.31 in order to determine the coefficients $a(t)$ through $d(t)$ as:

$$a(t) = w_i(t) \tag{3.36}$$

$$b(t) = \theta_i(t) \tag{3.37}$$

$$c(t) = \frac{1}{l^2}\Big(- 3w_i(t) - 2\theta_i(t)l + 3w_{i+1}(t) - \theta_{i+1}(t)l \Big) \tag{3.38}$$

$$d(t) = \frac{1}{l^3}\Big(2w_i(t) + \theta_i(t)l - 2w_{i+1}(t) + \theta_{i+1}(t)l\Big) \tag{3.39}$$

By substituting Equations 3.36 through 3.39 into Equation 3.31, Equation 3.31 can be rewritten as:

$$\begin{aligned} w(x,t) = w_i(t) + \theta_i(t)x + \frac{1}{l^2}\Big(-3w_i(t) - 2\theta_i(t)l + 3w_{i+1}(t) - \theta_{i+1}(t)l\Big)x^2 \\ + \frac{1}{l^3}\Big(2w_i(t) + \theta_i(t)l - 2w_{i+1}(t) + \theta_{i+1}(t)l\Big)x^3 \end{aligned} \tag{3.40}$$

which can be simplified as:

$$\begin{aligned} w(x,t) = \left(1 - 3\frac{x^2}{l^2} + 2\frac{x^3}{l^3}\right)w_i(t) + \left(\frac{x}{l} - 2\frac{x^2}{l^2} + \frac{x^3}{l^3}\right)l\theta_i(t) \\ + \left(3\frac{x^2}{l^2} - 2\frac{x^3}{l^3}\right)w_{i+1}(t) + \left(-\frac{x^2}{l^2} + \frac{x^3}{l^3}\right)l\theta_{i+1}(t) \end{aligned} \tag{3.41}$$

Equation 3.41 can then be rewritten as

$$w(x,t) = N_1(x)w_i(t) + N_2(x)\theta_i(t) + N_3(x)w_{i+1}(t) + N_4(x)\theta_{i+1}(t) \tag{3.42}$$

where N_1 through N_4 are the shape functions of the system and are equal to:

$$N_1(x) = 1 - 3\frac{x^2}{l^2} + 2\frac{x^3}{l^3} \tag{3.43}$$

$$N_2(x) = x - 2\frac{x^2}{l} + \frac{x^3}{l^2} \tag{3.44}$$

$$N_3(x) = 3\frac{x^2}{l^2} - 2\frac{x^3}{l^3} \tag{3.45}$$

$$N_4(x) = -\frac{x^2}{l} + \frac{x^3}{l^2} \tag{3.46}$$

Next, the kinetic and bending strain energy are examined for a link element which are expressed as [20]

$$T_i(t) = \frac{1}{2}\int_0^l \rho A\left(\frac{\partial w(x,t)}{\partial t}\right)^2 dx \tag{3.47}$$

$$V_i(t) = \frac{1}{2}\int_0^l EI\left(\frac{\partial^2 w(x,t)}{\partial x^2}\right)^2 dx \tag{3.48}$$

which can be rewritten in quadratic form as [20]

$$T_i(t) = \frac{1}{2}\dot{q}_i(t)^T[m]\dot{q}_i(t) \tag{3.49}$$

$$V_i(t) = \frac{1}{2}q_i(t)^T[k]q_i(t) \tag{3.50}$$

where $T_i(t)$ is the kinetic energy of the element, $V_i(t)$ is the potential energy of the element caused by the bending strain, [m] is the mass matrix of the element, [k] is the stiffness matrix of the element, ρ is the density of the element, A is the cross sectional area, E is Young's modulus of the element, I is the moment of inertia of the element's cross section, $q_i(t)$ and $\dot{q}_i(t)$ are the vector containing the coordinates of the link element and the derivative with respect to time of the vector respectively which is given by [20]

$$q_i(t) = \begin{Bmatrix} w_i(t) \\ \theta_i(t) \\ w_{i+1}(t) \\ \theta_{i+1}(t) \end{Bmatrix} \tag{3.51}$$

Finally, Equation 3.41 is substituted into Equations 3.47 through 3.50 and the integrations are completed in order to derive the mass and stiffness matrices of the link element, i, as [20]

$$[m]_i = \frac{\rho A l}{420}\begin{bmatrix} 156 & 22l & 54 & -13l \\ 22l & 4l^2 & 13l & -3l^2 \\ 54 & 13l & 156 & -22l \\ -13l & -3l^2 & -22l & 4l^2 \end{bmatrix} \tag{3.52}$$

$$[k]_i = \frac{EI}{l^3}\begin{bmatrix} 12 & 6l & -12 & 6l \\ 6l & 4l^2 & -6l & 2l^2 \\ -12 & -6l & 12 & -6l \\ 6l & 2l^2 & -6l & 4l^2 \end{bmatrix} \tag{3.53}$$

These matrices for each element are then able to be added together to create the mass and stiffness matrices for the overall link with each element of the matrices corresponding to a displacement or rotation of a node. An example of an mxm mass and stiffness matrix addition can be seen in Equations 3.54 and 3.55. Finally, due to the boundary conditions of the link being a pinned joint, no vertical displacement is possible at the first node. Therefore the first column and row of the overall mass and stiffness matrices can be eliminated to simplify computations.

$$[M] = \frac{\rho A l}{420}\begin{bmatrix} m_{11}^{(1)} & m_{12}^{(1)} & m_{13}^{(1)} & m_{14}^{(1)} & 0 & 0 & ... & 0 & 0 \\ m_{21}^{(1)} & m_{22}^{(1)} & m_{23}^{(1)} & m_{24}^{(1)} & 0 & 0 & ... & 0 & 0 \\ m_{31}^{(1)} & m_{32}^{(1)} & (m_{33}^{(1)} + m_{11}^{(2)}) & (m_{34}^{(1)} + m_{12}^{(2)}) & m_{13}^{(2)} & m_{14}^{(2)} & ... & 0 & 0 \\ m_{41}^{(1)} & m_{42}^{(1)} & (m_{43}^{(1)} + m_{21}^{(2)}) & (m_{44}^{(1)} + m_{22}^{(2)}) & m_{23}^{(2)} & m_{24}^{(2)} & ... & 0 & 0 \\ & & & . & & & & & \\ & & & . & & & & & \\ & & & . & & & & & \\ 0 & 0 & 0 & 0 & 0 & 0 & ... & m_{33}^{(m)} & m_{34}^{(m)} \\ 0 & 0 & 0 & 0 & 0 & 0 & ... & m_{43}^{(m)} & m_{44}^{(m)} \end{bmatrix} \tag{3.54}$$

$$[K] = \frac{EI}{l^3}\begin{bmatrix} k_{11}^{(1)} & k_{12}^{(1)} & k_{13}^{(1)} & k_{14}^{(1)} & 0 & 0 & ... & 0 & 0 \\ k_{21}^{(1)} & k_{22}^{(1)} & k_{23}^{(1)} & k_{24}^{(1)} & 0 & 0 & ... & 0 & 0 \\ k_{31}^{(1)} & k_{32}^{(1)} & (k_{33}^{(1)} + k_{11}^{(2)}) & (k_{34}^{(1)} + k_{12}^{(2)}) & k_{13}^{(2)} & k_{14}^{(2)} & ... & 0 & 0 \\ k_{41}^{(1)} & k_{42}^{(1)} & (k_{43}^{(1)} + k_{21}^{(2)}) & (k_{44}^{(1)} + k_{22}^{(2)}) & k_{23}^{(2)} & k_{24}^{(2)} & ... & 0 & 0 \\ & & & . & & & & & \\ & & & . & & & & & \\ & & & . & & & & & \\ 0 & 0 & 0 & 0 & 0 & 0 & ... & k_{33}^{(m)} & k_{34}^{(m)} \\ 0 & 0 & 0 & 0 & 0 & 0 & ... & k_{43}^{(m)} & k_{44}^{(m)} \end{bmatrix} \tag{3.55}$$

Upon computation of the mass and stiffness matrices, the natural frequencies of the models were calculated and compared with the AMM results to ensure accuracy. This was done by solving for the eigenvalues of the system. The eigenvalue problem seeks to solve the equation [20]:

$$\big[[K] - \omega^2[M]\big]\overrightarrow{X} = \overrightarrow{0} \tag{3.56}$$

Where [K] is the link's overall stiffness matrix, [M] is the link's overall mass matrix, ω are the natural frequencies of the link, and $\overrightarrow{X}$ are the mode shapes of the link. By rearranging Equation 3.56, the natural frequencies of the system in rad/sec can be solved by analysing where the determinant is equal to zero with the equation [20]:

$$|\omega^2[I] - [K][M]^{-1}| = 0 \tag{3.57}$$

More elements were added if the model was not accurate enough until the frequency of the modes of vibration were within acceptable ranges. As seen in Table 4 and 5, a six element model was determined to be accurate enough for use within this research, having a

maximum error of 1.26% on the forth mode of vibration. Higher modes of vibration do not have large enough amplitudes to create significant changes in the link's response.

The overall mass and stiffness matrices can be found in Appendix A.

Table 4: FEA Results

	2 Element	3 Element	4 Element	5 Element	6 Element	AMM
Rigid Body Motion (Hz)	0	0	0	0	0	0
First Mode (Hz)	6.60	6.57	6.56	6.56	6.56	6.56
Second Mode (Hz)	23.74	21.52	21.36	21.30	21.27	21.25
Third Mode (Hz)	57.65	49.19	45.09	44.74	44.55	44.33
Forth Mode (Hz)	105.99	90.28	83.81	77.36	76.77	75.81
Fifth Mode (Hz)	-	159.93	133.17	127.62	118.31	-
Sixth Mode (Hz)	-	231.80	207.72	185.42	180.63	-
Seventh Mode (Hz)	-	-	309.60	266.73	246.78	-
Eighth Mode (Hz)	-	-	408.70	376.97	335.94	-
Ninth Mode (Hz)	-	-	-	503.22	451.60	-
Tenth Mode (Hz)	-	-	-	637.12	596.35	-
Eleventh Mode (Hz)	-	-	-	-	739.52	-
Twelfth Mode (Hz)	-	-	-	-	916.80	-

Table 5: FEA Percent Error Compared to AMM

	2 Elements	3 Elements	4 Elements	5 Elements	6 Elements
Rigid Body Motion	0 %	0 %	0 %	0 %	0 %
First Mode	0.61%	0.15%	0%	0%	0%
Second Mode	11.72%	1.27%	0.52%	0.24%	0.09%
Third Mode	30.05%	10.96%	1.71%	0.92%	0.50%
Forth Mode	39.81%	19.09%	10.55%	2.04%	1.27%

3.2 Dynamic Model

Now that the mass and stiffness matrices have been created, they can be used to create the equations of motion for the flexible link. This was done by using a Lagrangian approach with the matrices to develop the equation of motion of the system and then applied to a State-Space representation of the model within MATLAB/Simulink.

3.2.1 Lagrangian Model

Lagrange is an energy based approach, relying on the principle of conservation of energy, to determine the equations of motion of dynamic systems. One of the key differences between the Lagrange method for determining the equations of motion when compared with a Newtonian method of determining the equations of motion are that Lagrangian Dynamics uses generalized coordinates which enables it to work equally well in any coordinate system [21]. The Lagrangian approach to obtain the equation of motion is:

$$\frac{d}{dt}\left(\frac{\partial \mathcal{L}}{\partial \dot{q}_i}\right) - \frac{\partial \mathcal{L}}{\partial q_i} = Q_i \tag{3.58}$$

where:

$$\mathcal{L} = T - V \tag{3.59}$$

where $\mathcal{L}$ represents the Lagrangian function, T is the total kinetic energy of the system and V is the total potential energy of the system. Q_i represents the generalized forces including the internal damping and the applied torque. Finally, q_i represents each of the generalized coordinates which are each of the states in the vector q shown in Equation 3.60 and in Figure 3.3.

$$q = \begin{Bmatrix} \theta_1 \\ w_2 \\ \theta_2 \\ w_3 \\ \theta_3 \\ w_4 \\ \theta_4 \\ w_5 \\ \theta_5 \\ w_6 \\ \theta_6 \\ w_7 \\ \theta_7 \end{Bmatrix} \tag{3.60}$$

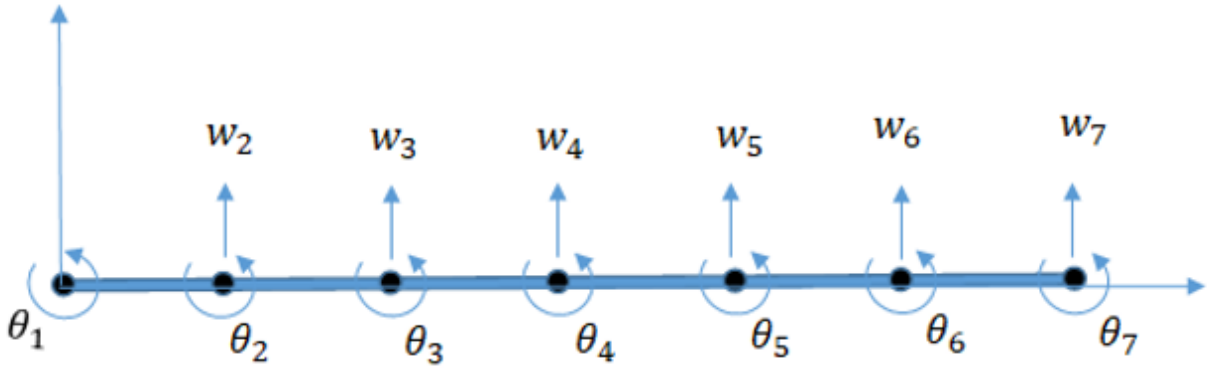

Figure 3.3: Six Element Representation of Flexible Link

The total kinetic energy of the system is represented in a quadratic form by:

$$T = \frac{1}{2}\dot{q}^T[M]\dot{q} \tag{3.61}$$

The total potential energy of the system is represented in a quadratic form by:

$$V = \frac{1}{2} q^T [K] q \tag{3.62}$$

The generalized forces in this system include the external forces and/or torque at each q_i and, for a damped system, all forces and/or torques created by the non-conservative forces such as those created by viscous damping. The generalized force vector is therefore given by:

$$Q = \tau - [C_D]\dot{q} \tag{3.63}$$

where τ is the applied torque vector given by

$$\tau = \begin{Bmatrix} \tau_1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{Bmatrix} \tag{3.64}$$

and $[C_D]$ is the damping matrix. If there is no damping in the system, the $[C_D]$ term will go to zero.

By substituting Equation 3.59 into Equation 3.58 and replacing the Q by the one described in Equation 3.63, the Lagrangian dynamics are written as:

$$\frac{d}{dt}\left(\frac{\partial T}{\partial \dot{q}}\right) + \frac{\partial V}{\partial q} = \tau - [C_D]\dot{q} \tag{3.65}$$

which can be reduced to

$$[M]\ddot{q} + [K]q = \tau - [C_D]\dot{q} \tag{3.66}$$

Rearranging Equation 3.66 gives:

$$[M]\ddot{q} + [C_D]\dot{q} + [K]q = \tau \tag{3.67}$$

3.2.2 Computer Simulation of the Dynamic Response

In order to simulate the dynamic response of the system in Equation 3.67, a State-Space model of the system was created. The system's states are described by a set of variables and its behavior over time is described by a series of equations written as matrices. The model consists of two components: a state equation that describes how the internal state variables change over time and an observation equation that maps the state variables to the observed outputs [22, 23].

The State-Space representation of the system was created using MATLAB/Simulink to track the position of the link over time. The State-Space representation of the system can be given by the equation:

$$\dot{X} = AX + Bu$$

and

$$Y = CX + Du$$

where:

X is the state vector containing 26 states. The first 13 are θ_1, w_2, θ_2, w_3, continuing through to θ_7 in order with the final 13 being the time derivative of each state as $\dot{\theta}_2$ through $\dot{\theta}_7$;

$\dot{X}$ is the time derivative of the state vector which describes the states of the system;

u is the scalar representing the applied torque to the joint;

A is the state matrix which represents the system;

B is the input matrix which represents the input;

Y is the output vector $= \begin{pmatrix} \theta \\ \delta \end{pmatrix} = \begin{pmatrix} \theta_1 \\ w_7 \end{pmatrix}$;

C is the output matrix that identifies which states are observed; and

D is the feed through matrix which represents the effect of a feed-forward loop of which none is present.

For the model without any damping added to the system, the resulting state matrices are:

$$A = \begin{bmatrix} [0]_{13X13} & [I]_{13X13} \\ -[M]^{-1}[K] & [0]_{13X13} \end{bmatrix} \tag{3.68}$$

$$B = \begin{bmatrix} [0]_{13X1} \\ [M]^{-1}[B^*] \end{bmatrix} \tag{3.69}$$

where $[B^*]$ is

$$[B^*] = \begin{Bmatrix} 1 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{Bmatrix} \tag{3.70}$$

$$\text{C}=\begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \end{bmatrix} \tag{3.71}$$

$$D = \begin{bmatrix} 0 \\ 0 \end{bmatrix} \tag{3.72}$$

Simulink was then used to create the model seen in Figure 3.4. By applying an initial tip displacement of 0.08 m to the system, the response of the base angle, θ (which is equal to θ_1) and the tip displacement (δ) can be seen in Figures 3.5 and 3.6 respectively. These results match with the expected results as the link oscillates with the theoretical natural frequencies as seen from the Fast Fourier Transform analysis shown in Figure 3.7.

When given a step-input, θ begins to trend upwards which is expected within this

simulation as there is no energy loss to limit the motion, there is a step-input continuously being applied to the link, and the model does not understand that after 2π radians, the link will be back in its starting position. It was also noted that the δ response requires the rigid body motion of the link subtracted from it as without this, the model reads the absolute position of the tip displacement. However, with the adjustment of subtracting θ multiplied by the length of the link, which is seen in Figure 3.4, the relative tip deflection is obtained.

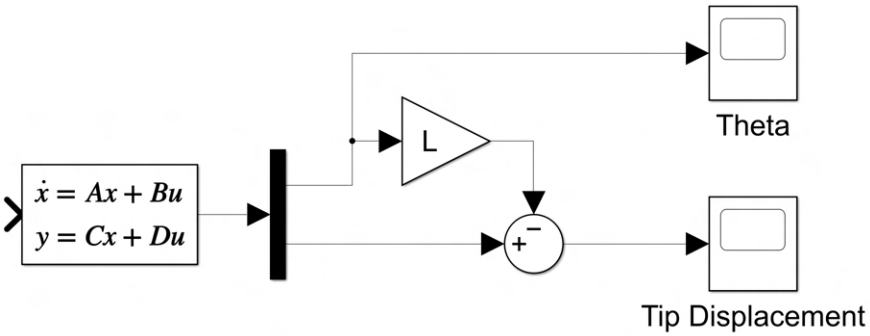

Figure 3.4: Link Model Implemented in Simulink

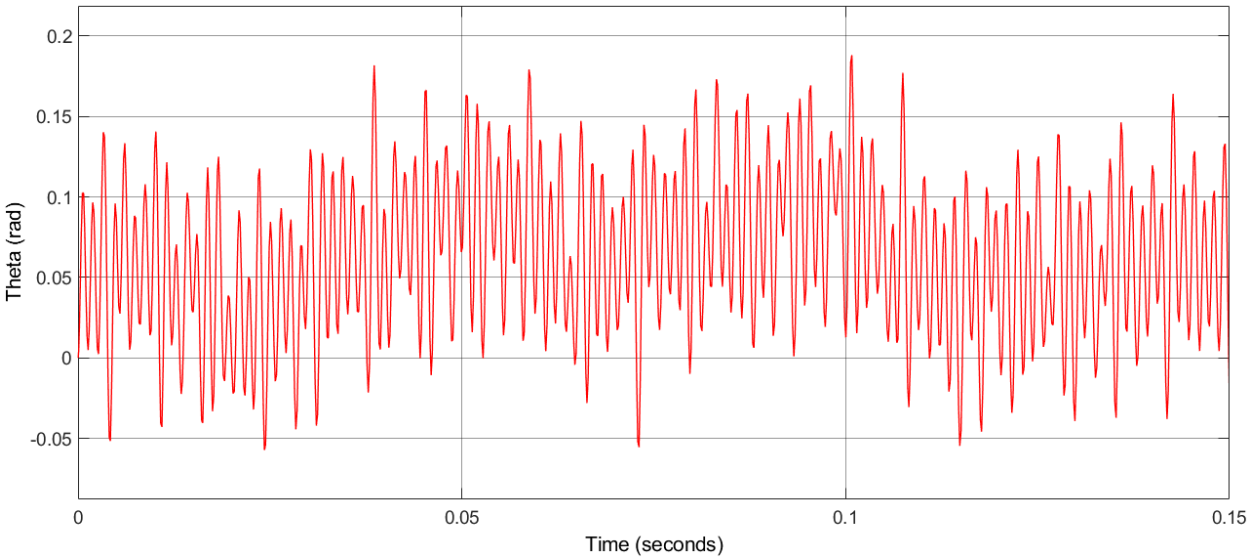

Figure 3.5: Free Vibration θ Response Due to 0.08 m Initial Tip Deflection

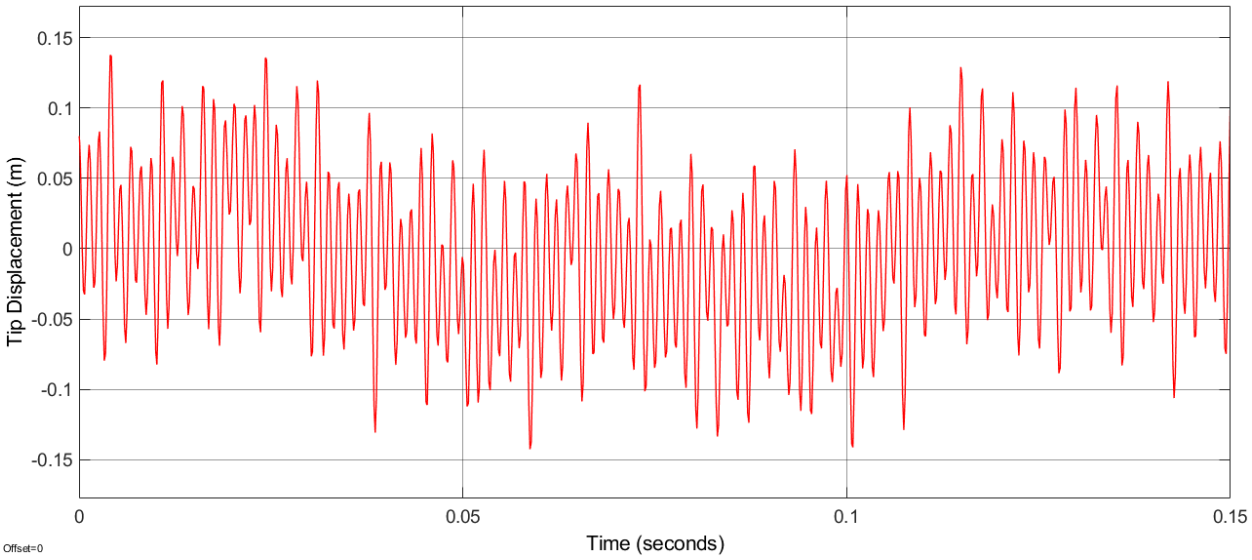

Figure 3.6: Free Vibration δ Response Due to 0.08 m Initial Tip Deflection

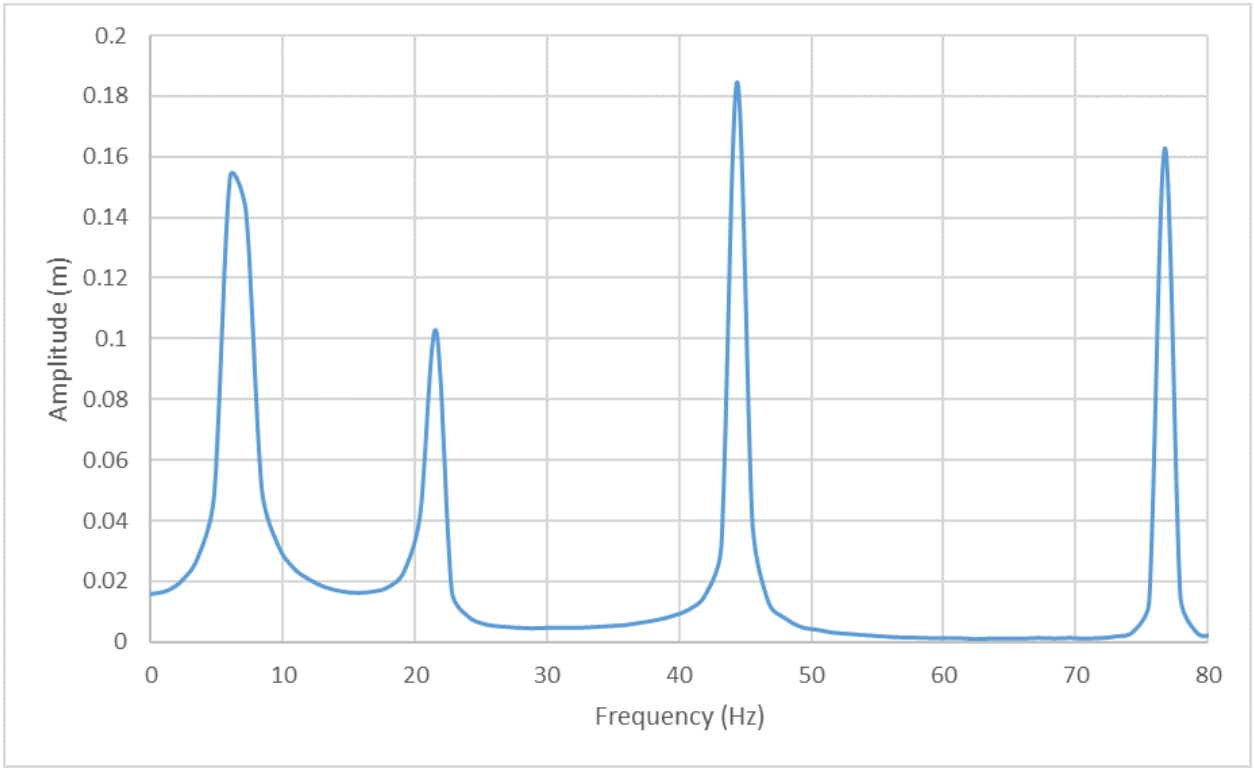

Figure 3.7: Model Fast Fourier Transform Due to Free Vibration Response

3.2.3 Model Limitations

The mathematical model that has been derived uses the Euler-Bernoulli Beam Theory. The three key assumptions of the Euler-Bernoulli Beam theory are that the cross-section is infinitely rigid in its own plane, the cross-section of a beam remains plane after deformation, and the cross-section remains normal to the deformed axis of the beam [24]. These assumption are only valid under small deflections which are where the tip deflection is less than 10% the total length of the link [25].

The other key limitation of the model is that due to the absolute position of δ being measured, the rigid body motion needed to be subtracted to obtain the relative position. This was accomplished by using the small angle approximation which is valid for angles between -0.2 and 0.2 rads [26]. While this limits the range of angles that can be applied to model, the focus of the Fuzzy Logic Controller is to dampen the vibration of the link. It is possible even with these limitation on the model to excite the link to ensure that the vibration is reduced and the the link still moves to the desired angular position.

3.3 Experimental Verification of the Dynamic Model

3.3.1 Experimental Setup

In order to verify that the model developed for MATLAB was comparable to the actual link where testing would be carried out, seen in Figure 3.8, the natural frequencies of the link needed to be verified to see if they matched the model. This was done by connecting a Tektronix TBS 1052B-EDU oscilloscope to piezoelectric sensors placed along the length of the link, exciting the link, and measuring the Fast Fourier Transform (FFT) response of the link. This FFT response was analyzed using the built in FFT analysis function of the oscilloscope.

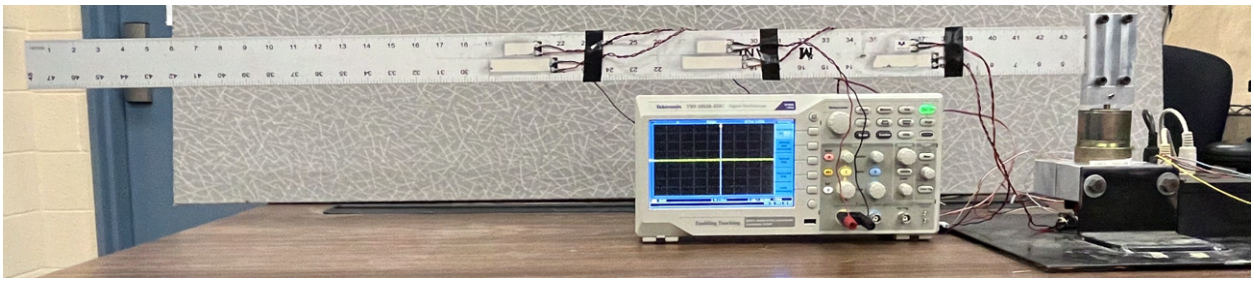

Figure 3.8: Flexible Link FFT Experimental Set-Up

3.3.2 Free Response Evaluation

Upon exciting the link, the Tektronix TBS 1052B-EDU produced the FFT response shown in Figures 3.9 and 3.10. It can be seen from these figures that there are three key natural frequencies that have an impact on the link's vibration not including the rigid body motion. The fourth natural frequency has a sufficiently small amplitude that it does not meaningfully contribute to the link's vibration response. It is also noted that there is a substantial peak at exactly 60 Hz. This is cause by several sources of power, from the light system to other machinery and devices within the area all operating on a 60 Hz frequency. While efforts were made to reduce this, it was never able to be fully eliminated from the signal however it is not a natural frequency of the link and was present with a similar

amplitude even when there was no vibration in the link or when the piezoelectric sensors were not connected at all.

As can be seen in Figure 3.9, the center position on the screen is at 62.5 Hz and each square moves 12.5 Hz (therefore having 0 Hz at the far left of the screen). The data was then captured and read using the cursor function as seen in Figures 3.9 and 3.10. The oscilloscope is only able to read the frequency position in 0.5 Hz increments but the first three natural frequencies occur at 6.5 Hz, 21.5 Hz and 44 Hz which corresponds to the expected results on 6.56 Hz, 21.25 Hz and 44.33 Hz. This does indicate that there will be very little damping occurring in the actual link itself which is expected and will be explained more in depth in Chapter 4 of this thesis where damping is added to the model. This conclusion can be made as the damped natural frequencies (which is what will be observed in real life) is very close to the undamped natural frequency of the link which indicates that there is a very small damping ratio.

These results show that the model developed for the link is an accurate representation of the actual link. While the fourth natural frequency which was verified in the model does not have a substantial impact on the physical system due to its low amplitude, the first three natural frequencies have been matched to the system effectively meaning that the model can be used to design an Active Vibration Controller for this link.

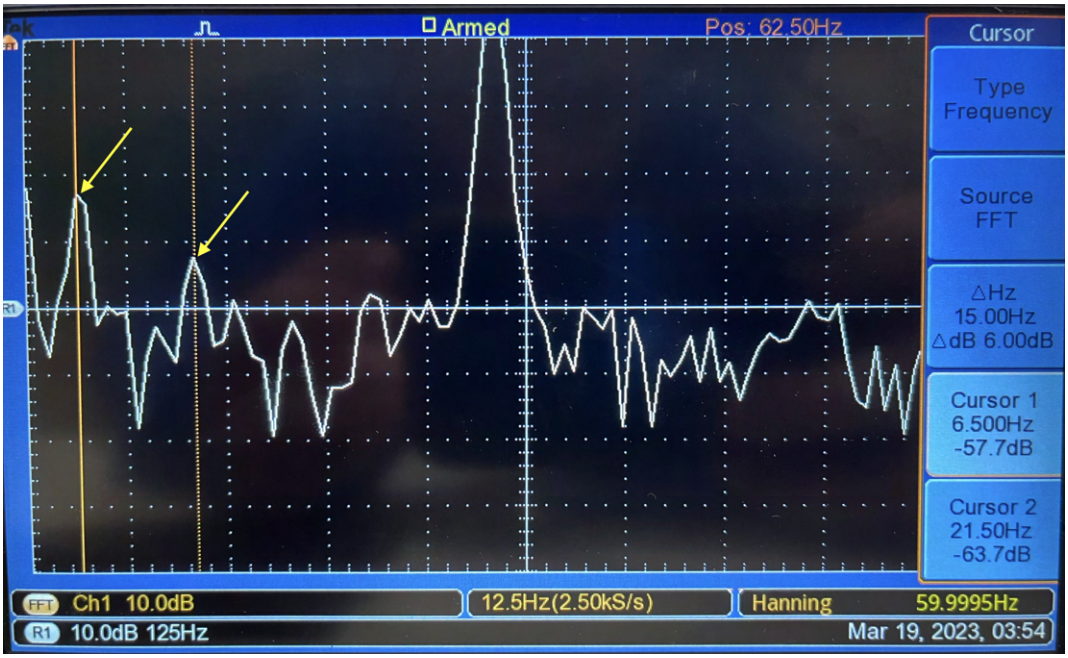

Figure 3.9: Fast Fourier Transform First Two Natural Frequencies

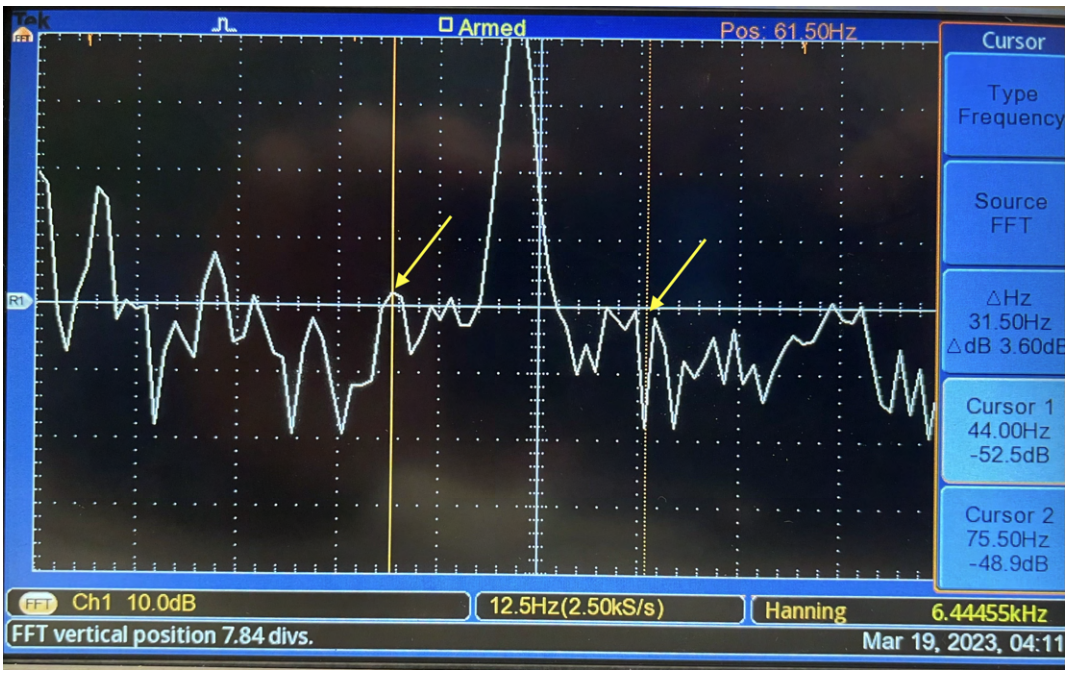

Figure 3.10: Fast Fourier Transform Third and Fourth Natural Frequencies

3.4 Simscape Model

With the Simulink model having been built and verified, another model of the link was designed using Simscape to be used in conjunction with the previously presented Simulink model. This was done both to create a model that is derived from the physical properties of the link as one of the key characteristics of Simscape is that the model is created using physical modeling as opposed to Simulink where the equations of motion must be derived and blocks are used to build the mathematical model. This was also done to address the small angle approximation limitation of the Simulink model because in the Simscape model, the deflection is measured with respect to the rigid body position of the link. Simscape still models the link using the Euler-Bernoulli Beam Theory however so the tip deflection being limited to 10% of the total length of the link is still applicable. Simscape also produces a 3 dimensional model of the physical system which can aid in the understanding and visualization of the response of the system. The Simscape model for the link can be seen in Figure 3.11 with the 3D model shown in Figure 3.12 containing the link and the motor hub. The block diagram consists of:

- model configuration blocks that identify the reference frame of the model, gravity, and solver settings (Figure 3.11 - Blocks 1),
- the input signal which represents the desired angular position (Figure 3.11 - Block 2),
- the proportional gain block that scales the angular position error (Figure 3.11 - Block 3),
- a revolute joint block that models a joint with one rotational degree of freedom (Figure 3.11 - Block 4),
- a solid block that has a cylinder with the dimensions of the motor hub and will be able to have mass added to it in the future to represent the motor hub (Figure 3.11 - Block 5),

- a rigid transform block that creates a fixed spatial relationship between the motor hub and the flexible link where they connect (Figure 3.11 - Block 6),
- the flexible link block that has the length, width, and height dimensions of the link along with its material properties and the option to add structural damping in the future (Figure 3.11 - Block 7),
- a transform sensor block that measures the tip deflection with respect to the rigid body position of the link (Figure 3.11 - Block 8),
- a block that stores the tip deflection data to the workspace (Figure 3.11 - Block 9), and
- a block that stores the angular position data to the workspace (Figure 3.11 - Block 10).

Upon exciting the link in an open loop configuration with a unit pulse input for 0.02 seconds to give the link an initial condition, the free vibration response of the link model compared against the Simulink model with the same input can be seen in Figures 3.13 and 3.14.

While the reponses are not identical they are similar. Both the Simulink and the Simscape θ responses increase over time, reaching approximately 0.3 rads after 2 seconds while oscillating in a 0.12-0.15 rad band. The tip deflection has a smoother response and is slightly larger in the Simscape model than the Simulink model, reaching 0.095 m compared to 0.085 m in Simulink. The response does however have a similar shape between the two models. This is because the Simulink model simulates more of the link's natural frequencies than the Simscape model. While these models are not identical, they are comparable and will be useful for collecting additional simulation data before using the controller on a physical system.

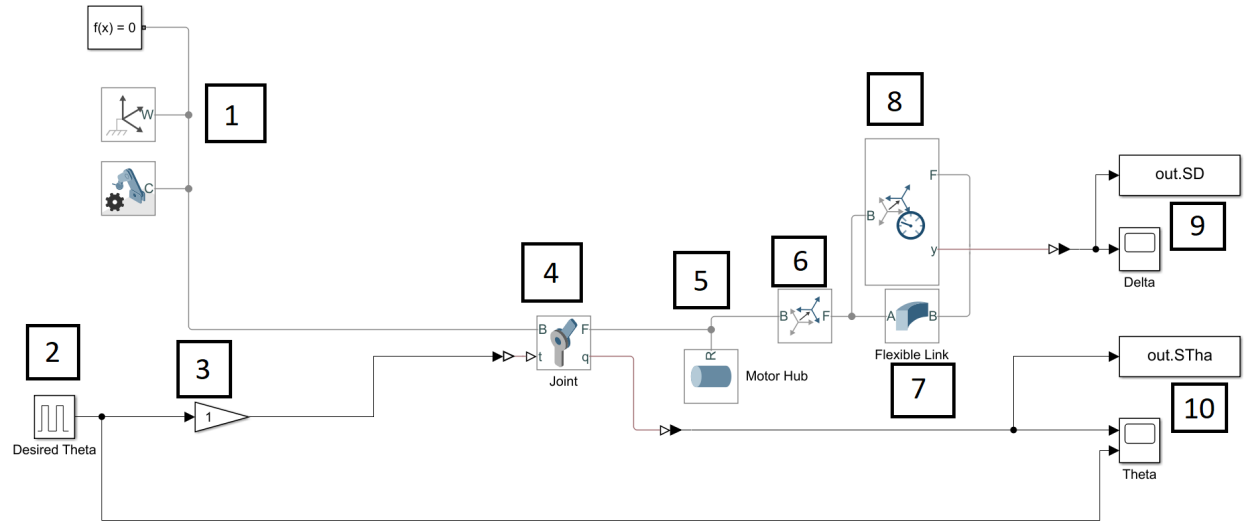

Figure 3.11: Simscape Link Model

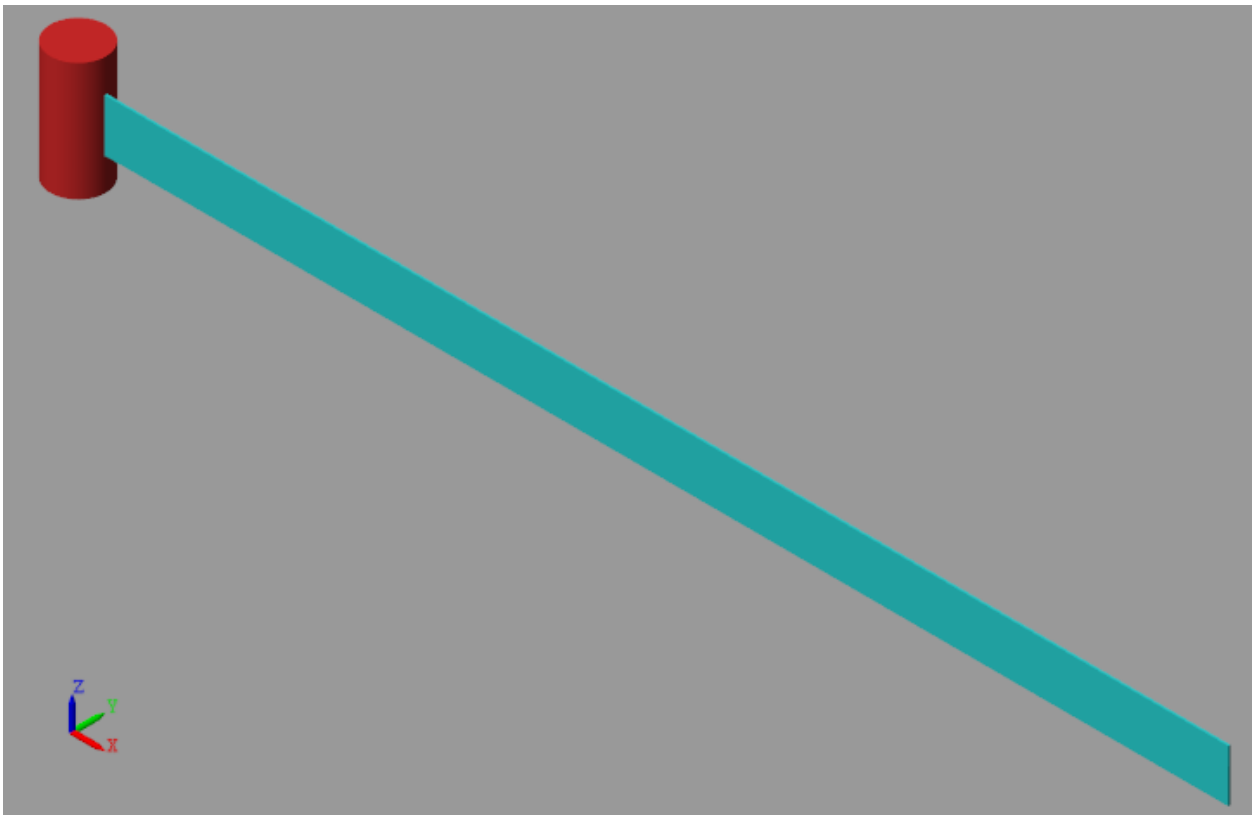

Figure 3.12: Simscape Link Model 3D Rendering

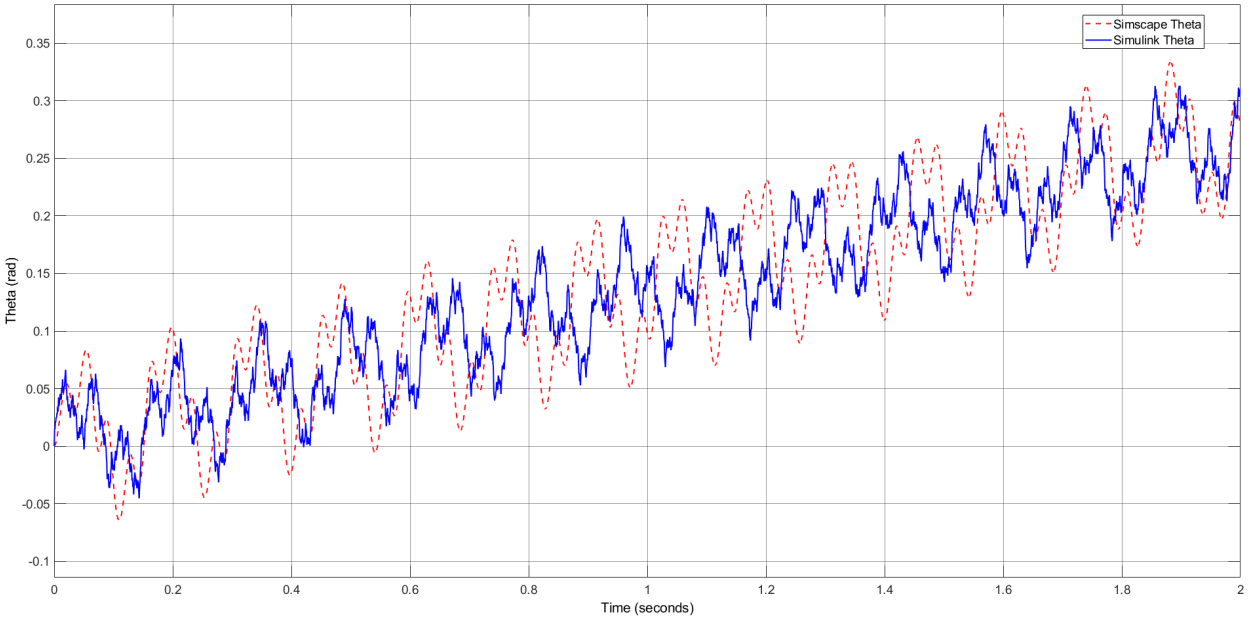

Figure 3.13: Simscape Link Model θ Response Due to a Unit Pulse Input for 0.2 Seconds VS Simulink Response

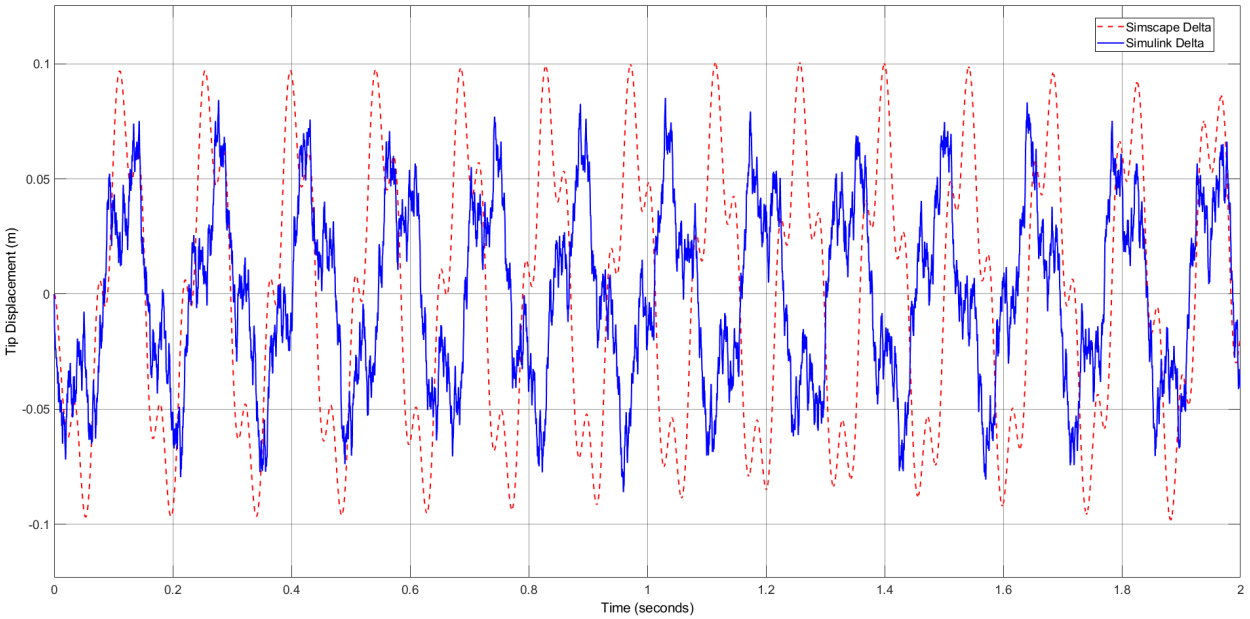

Figure 3.14: Simscape Link Model δ Response Due to a Unit Pulse Input for 0.2 Seconds VS Simulink Response

4 Vibration Control

This chapter focuses on the development and implementation of the Fuzzy Logic Controller to control the link's vibration. The first section within this chapter gives a more in depth overview of Fuzzy Logic and its strengths and weaknesses as a control method. This then moves into a section on the design of a Fuzzy Logic Controller to reduce the vibration in a flexible link. Next, the model was updated by adding a damping matrix and motor inertia to the system. The fourth section of this chapter shows testing with slight variations on the Fuzzy Logic Controller's Rule Base to ensure that the Rule Base has been optimized.

4.1 Fuzzy Logic

Fuzzy Logic was first proposed in 1965 by Lotfi Zadeh [10] as a way to consider classes of objects that contain members that are not neatly described in classical terms as either being true or false. They recognized that in some cases, status of objects within particular classes was ambiguous but not due to lack of information, but rather due to the imprecision or vagueness in the language used [11].

In order to represent the uncertainty of these components of the set, a degree of membership, which ranges from 0 to 1 is assigned to each member of the set. A value of 0 means that the relationship between the component and the set is "totally false" or that the component is not a member of that set and a value of 1 means that the relationship between the component and the set is "totally true". While those are the only option in classical binary logic, Fuzzy Logic can include relationships such as "unlikely true" or "mostly false" by using values between zero and one [27].

These relationships can then be used to plot the relationship between a component and a group of sets, with a component being able to have a non-zero value for more than one

set [11, 27]. For example, if the fuzzy set is representing angular position error, the variables in question could include "Large Positive", "Medium Positive", "Small Positive", "Zero", "Small Negative, "Medium Negative", or "Large Negative", with each angular position error having a value between 0 and 1 for each variable based on how well they fit each description such as with the membership functions seen in Figure 4.1.

In 1975, Mamdani [12] became the first person to successfully apply Fuzzy Logic to control systems, creating a Fuzzy Logic Controller as seen in Figure 2.2 (repeated below). In Fuzzy Logic Control, decision-making is based on the use of if_then_else rules that describe the relationships between inputs and outputs in a system. For example, a rule might state that "if the angular position error is large positive and the tip deflection is large negative, apply a large positive torque to the system". These rules are typically derived through expert knowledge and research and are designed to represent the relationships between the inputs and outputs in the system [11].

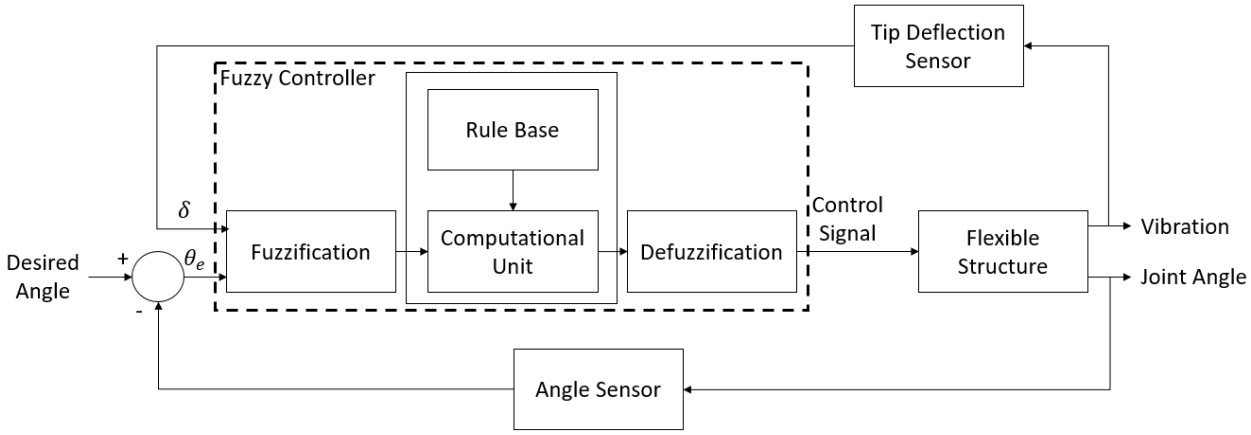

Figure 2.2: Fuzzy Logic Controller (Repeated)

The method of relating the inputs and outputs in the system based on fuzzy rules is known as fuzzy inference and has four key components. These components are the Fuzzifier, the Computational Unit of the controller, the Rule Base, and the Defuzzifier.

This process starts with the fuzzification of the crisp inputs through membership functions [27]. There are several different ways to design these membership functions including triangular, trapezoidal, Gaussian functions and more. The choice of membership function will depend on the application and the desired effect of the controller. When used in position tracking, triangular and trapezoidal membership functions both outperform Gaussian membership functions and while triangular and trapezoidal have similar rise time and overshoot, triangular membership functions have a lower steady-state error [28]. Therefore, in this research, triangular membership functions were primarily used. In order to fuzzify the input, the degree of membership for a crisp input must be known.

An example of the min-max fuzzification process can be shown by using the membership functions from Figures 4.1 to 4.3. If a flexible link has an angular position error of positive 0.8 rads and a tip deflection of positive 0.2 m then the angular position error has the membership degrees that can be seen in Figure 4.1 while the tip deflection has the membership degrees that can be seen in Figure 4.2. The angular position error's degrees of membership are:

- Large Negative: Membership Degree = 0
- Medium Negative: Membership Degree = 0
- Small Negative: Membership Degree = 0
- Zero: Membership Degree = 0
- Small Positive: Membership Degree = 0.4
- Medium Positive: Membership Degree = 0.6
- Large Positive: Membership Degree = 0

and the tip deflection degrees of membership are:

- Large Negative: Membership Degree = 0

- Medium Negative: Membership Degree = 0
- Small Negative: Membership Degree = 0
- Zero: Membership Degree = 0.6
- Small Positive: Membership Degree = 0.4
- Medium Positive: Membership Degree = 0
- Large Positive: Membership Degree = 0

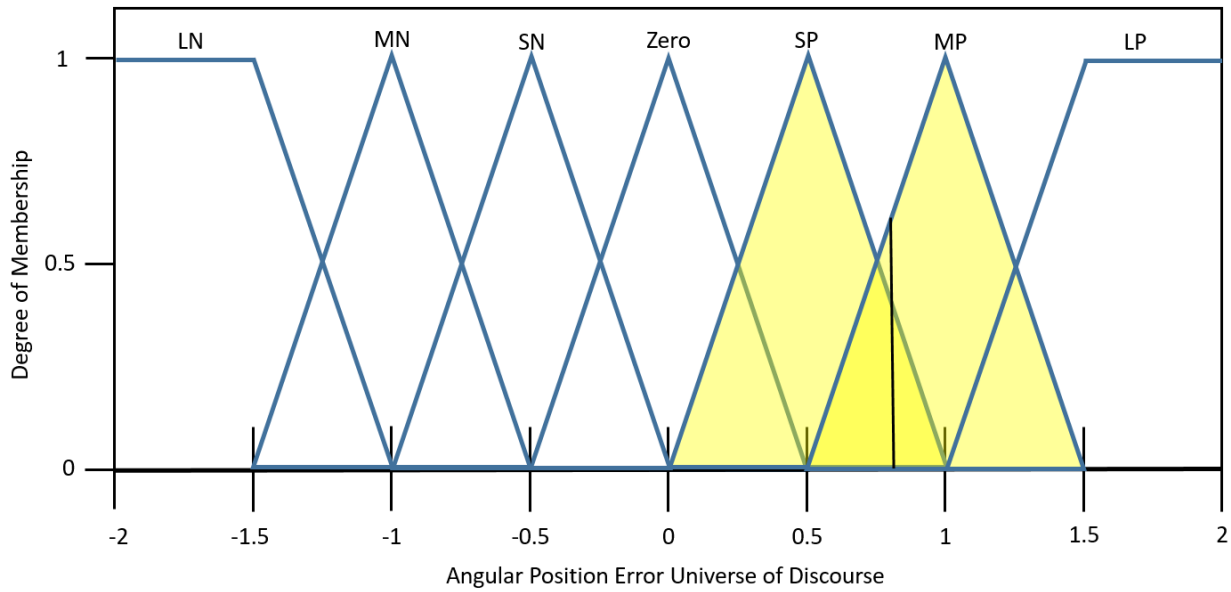

Figure 4.1: Fuzzy Logic Controller Membership Degree for Angular Position Error of 0.8 rad

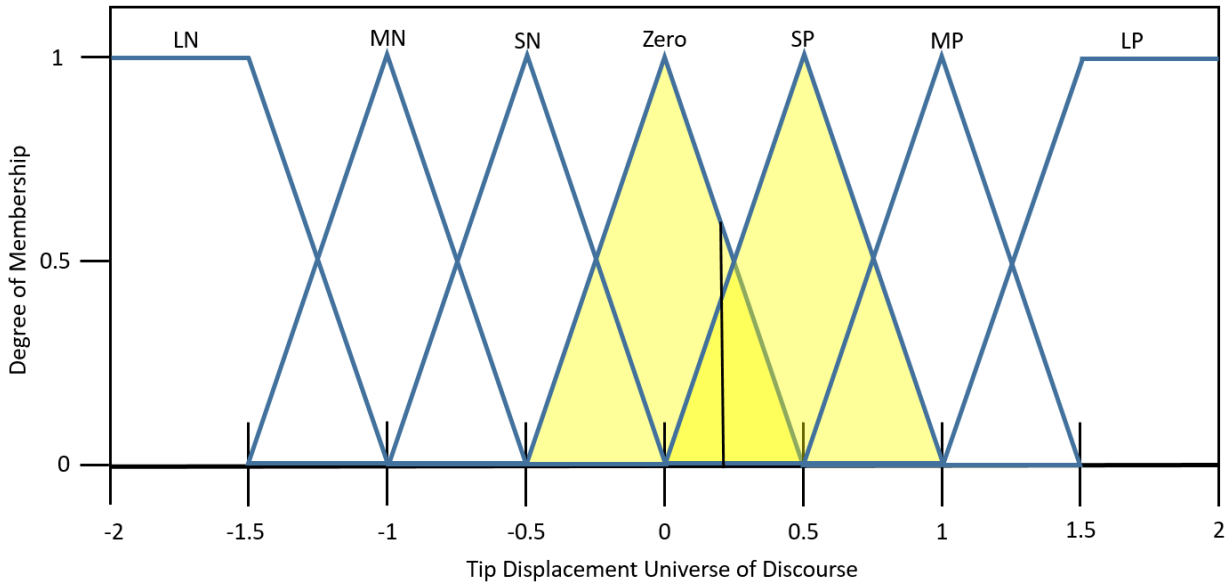

Figure 4.2: Fuzzy Logic Controller Membership Degree for Tip Deflection of 0.2 m

Once the input has been fuzzified, the Rule Base must be applied to produce a fuzzy output. For this example, the four rules that are affected are:

- Rule 1: if the angular position error is Small Positive AND the tip deflection is Zero THEN apply Small Positive torque,
  - Antecedent 1: angular position error is Small Positive - membership degree 0.4,
  - Antecedent 2: tip deflection is Zero - membership degree 0.6,
  - Activation degree = min(0.4, 0.6) = 0.4
- Rule 2: if the angular position error is Small Positive AND the tip deflection is Small Positive THEN apply Zero torque,
  - Antecedent 1: angular position error is Small Positive - membership degree 0.4,
  - Antecedent 2: tip deflection is Small Positive - membership degree 0.4,
  - Activation degree = min(0.4, 0.4) = 0.4
- Rule 3: if the angular position error is Medium Positive AND the tip deflection is Zero THEN apply Medium Positive torque,

  - Antecedent 1: angular position error is Medium Positive - membership degree 0.6,
  - Antecedent 2: tip deflection is Zero - membership degree 0.6,
  - Activation degree = min(0.6, 0.6) = 0.6

- Rule 4: if the angular position error is Medium Positive AND the tip deflection is Small Positive THEN apply Medium Positive torque.
  - Antecedent 1: angular position error is Medium Positive - membership degree 0.6,
  - Antecedent 2: tip deflection is Small Positive - membership degree 0.4,
  - Activation degree = min(0.6, 0.4) = 0.4

Based on the activation degrees, the fuzzy set of the torque can be determined based on the activation degree of the torque membership function. When there are two activation degrees for the same torque membership function, the larger value is used. Therefore the torque fuzzy set seen in Figure 4.3 has the values

- Zero Torque: Membership degree = 0.4,
- Small Positive Torque: Membership degree = 0.4,
- Medium Positive Torque: Membership degree = max(0.6, 0.4)=0.6

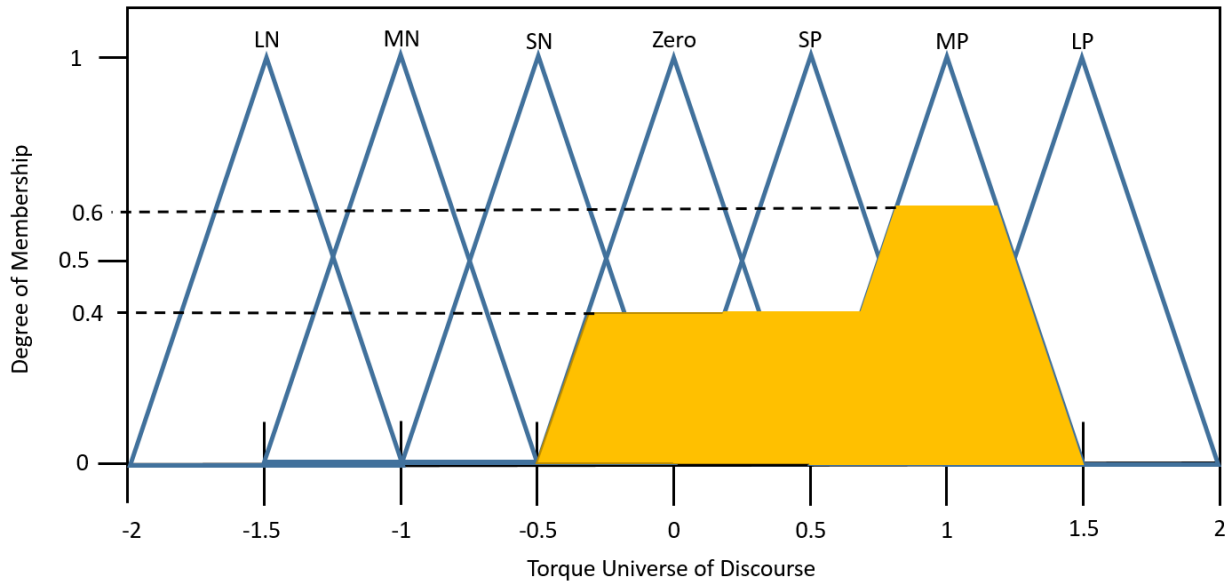

Figure 4.3: Fuzzy Logic Controller Torque Output Fuzzy Set with Angular Position Error of 0.8 and Tip Deflection of 0.2

This fuzzy set is then transformed into a crisp output value using a defuzzification process. There are several methods for defuzzification, including the centroid method (center of area or center of gravity), the mean of maximum (MOM) method, and the weighted average method. The choice of defuzzification method will depend on the application. For this controller, the center of gravity method was used because it requires minimal computational power compared with some other methods, it provides continuous, smooth outputs which minimize abrupt changes to the system, and its key drawback of producing inaccurate outputs when the activated rules have a large gap between them with no activation does not affect this Fuzzy Logic Controller as there are no "forbidden zones" in the controller [11].

The equation to defuzzify the torque fuzzy set using the Center of Gravity method is [11, 27].

$$u_{COG} = \frac{\int u\mu(u)du}{\int \mu(u)du} \tag{4.1}$$

where u_{COG} is the crisp output value of the torque, u is the position along the universe of discourse, and $\mu(u)$ is the membership function. By assuming a uniform density throughout the torque's fuzzy set from Figure 4.3, Equation 4.1 can be solved to determine that the center of gravity of the shape has a value along the Universe of Discourse of u_{COG}=0.561. Therefore, when the angular position error is 0.8 radians and the tip deflection is 0.2 m, the crisp torque output that is applied to the motor is 0.561 Nm.

Fuzzy Logic has a number of advantages when used as a controller, including its ability to represent uncertainty and imprecision and its ability to incorporate expert knowledge. One of the most important advantages of it is that it is a non-model based controller [11]. Therefore, once the controller has been developed and tested, it is able to be incorporated within similar systems without needing to create a model of the system. However, some drawbacks of Fuzzy Logic Controllers are that they can be complex to implement, especially in large-scale control systems, and may call for a lot of tuning to achieve optimal results by adjusting the gain values slightly.

4.2 Controller Design

The Fuzzy Logic Controller seen in Figure 4.4 was designed to control the vibration of the link. The two inputs to the Fuzzy Logic Controller are the angular position error, e_θ=θ_d-θ, where θ_d is the desired angular position and the tip displacement (δ). In order to determine the output, it must be noted that the only direct force or torque that can be placed on the link comes from the motor at the base. This is why a torque applied to the base of the link is the output of the Fuzzy Logic Controller.

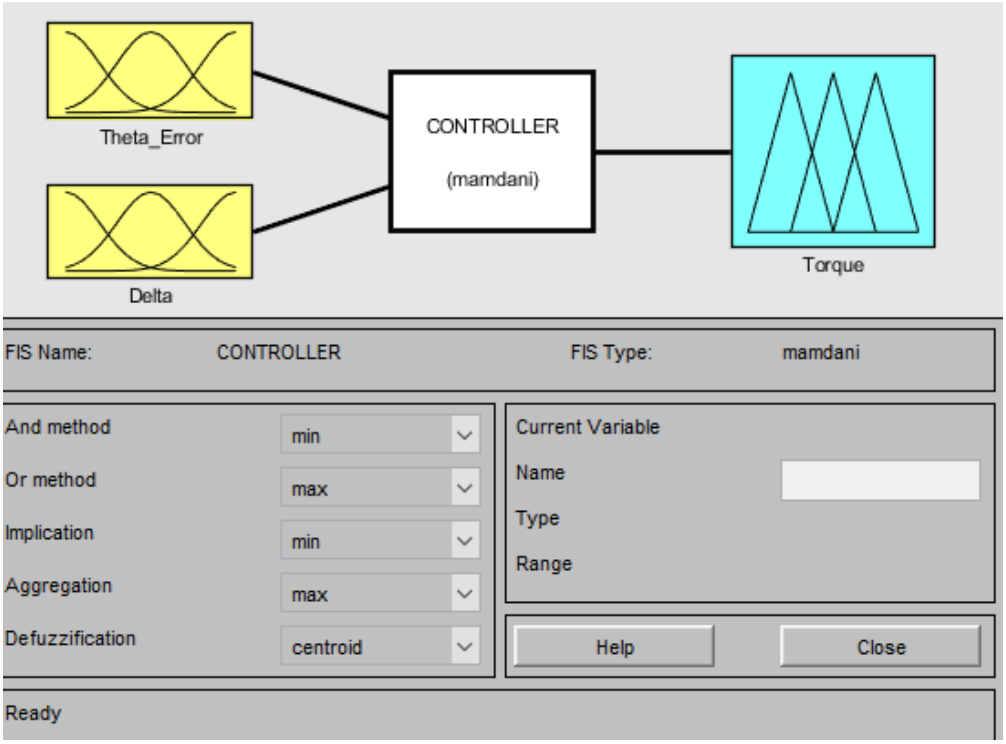

Figure 4.4: Fuzzy Logic Controller Implementation Using Fuzzy Inference Toolbox

The Fuzzy Logic Toolbox in Matlab was used to design a Mamdani style Fuzzy Logic Controller. A Mamdani style controller was chosen over a Sugeno style controller as the key differences between the two are how the crisp output is calculated. With a Mamdani controller, output membership functions are created and a defuzzification process is used to turn it crisp. Sugeno however, uses a weighted average method [29]. The key benefits of the Mamdani is that while it takes more computational power, it is more intuitive to understand and it is easier to incorporate expert knowledge within the controller [29]. The rotating link is a simple enough system that even with the additional computational power required with the Mamdani controller, it is still able to keep up with the real-time operation of the link. There has also been substantial research on controlling the vibration of a rotating, flexible link using conventional and non-conventional controllers [2, 5, 7–9, 13–16, 30] which can be combined to determine an appropriate range of torque values based on the link's properties.

With the layout of the controller designed, membership functions are needed for each input and the output. The membership functions for e_θ, δ, and the output torque can be seen in Figures 4.5, 4.6, and 4.7 respectively.

It was decided to create standardized membership functions with a universe of discourse ranging from -2 to 2 with units of rads for the angular position error, m for the tip displacement, and Nm for the torque. This is because scaling factors will need to be applied to the inputs and output based on what values are reasonable within the link being controlled. For example, a 2 m long link will be able to have a considerably larger tip displacement than a 0.3 m long link. Therefore, instead of designing a controller with membership functions that are ideal for the link analyzed in this research which would require updating all 21 membership functions each time the controller is updated, only the three scaling factors (one for each input and output) must be updated. This also helps to make the controller more easily incorporated with other links as it is more intuitive to scale numbers ranging from -2 to 2 rather than numbers with decimal points.

Each universe of discourse has seven membership functions within them which share names across the universes of discourse. These membership functions are called Large Negative (LN), Medium Negative (MN), Small Negative (SN), Zero, Small Positive (SP), Medium Positive (MP), and Large Positive (LP). An important aspect of these membership functions are the choice in using predominantly triangular membership functions. Triangular membership functions were chosen because when compared against trapezoidal and Gaussian membership functions on position control, the triangular and trapezoidal both outperformed Gaussian and while triangular and trapezoidal have similar rise time and overshoot, triangular membership functions have a lower steady-state error [28].

There are however four membership functions that are not triangular which are the large

negative and large positive input membership functions. Instead they are linear s-shaped saturation membership functions and linear z-shaped saturation membership functions. This is because even if the angular position error or tip displacement goes past the expected values, no matter how large or small the values become, the inputs will still fall within a membership function with a non-zero value. If these were triangular functions, then any inputs that are outside of the largest membership functions would not fall within the Rule Base and therefore, the controller would produce an output torque of 0 Nm. The large positive and negative torque membership functions however remain triangular and that is because even if the inputs are very large, the torque cannot be increased indefinitely or the link would over-correct too aggressively.

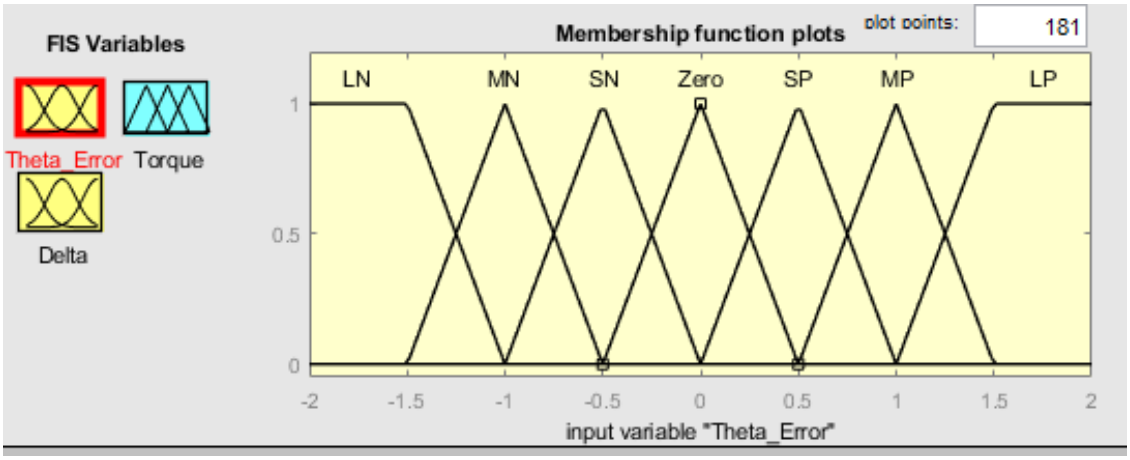

Figure 4.5: e_θ Membership Functions (rads)

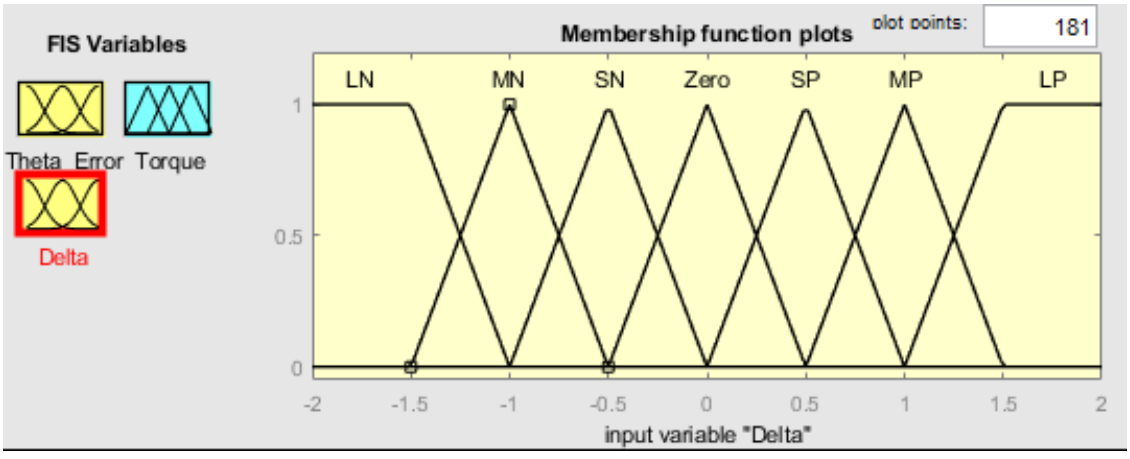

Figure 4.6: δ Membership Functions (m)

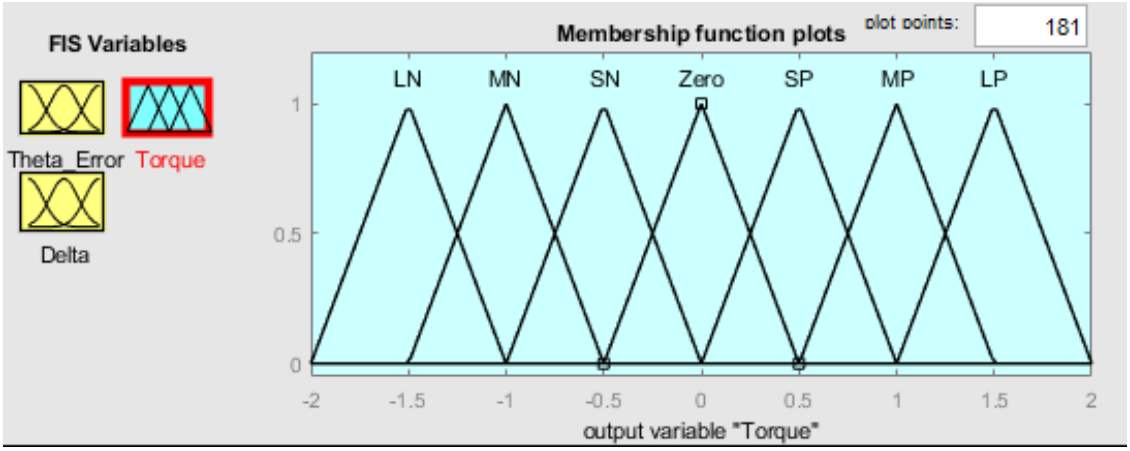

Figure 4.7: Torque Membership Functions (Nm)

With the type of Fuzzy Logic Controller determined and all membership functions created, expert knowledge was analyzed from other research [9, 13–16] related to the vibration control of rotating flexible links to create the if_then_else rule system that can be seen in Table 6. These rules follow the pattern:

If the ANGLE ERROR is _____ and TIP DEFLECTION is _____ then TORQUE is _____; else

Where the angular error is read off the left most column, the tip deflection is read off the top row and the output torque is the membership function where the angle error and tip deflection intersect.

These rules were determined by looking at the four most extreme combinations of the angular joint position error, e_θ and the tip deflection, δ, being large positive and large negative as seen in Figure 4.8. The positive e_θ occurs when the link is clockwise compared to the desired angle. By extension, the negative e_θ occurs when the link is counter-clockwise compared to the desired angle. This is because the angular error is the actual angle subtracted from the desired angle so even though convention is for counter-clockwise to be positive, as the actual angle is subtracted it becomes negative. The

tip deflection is positive when the tip is right of the rigid body position when looking from base to tip and negative when the tip is left of the rigid body position when looking from base to tip. Finally, the applied torque is positive in the counter-clockwise direction and negative in the clockwise direction as per convention.

Using these definitions, it becomes intuitive what torque should be applied. When the angular position error is large negative and the tip deflection is large positive as seen in Figure 4.8 a. or when the angle error is large positive and the tip deflection is large negative as seen in Figure 4.8 b. the deflection of the link is causing the link to want to move further away from the desired angular position. Therefore, a large torque is applied to move the hub to the desired position as it not only needs to move a large distance, but it is also required to overcome the potential energy stored in the link.

The other two extremes are when the angular position error is large negative and the tip deflection is large negative as seen in Figure 4.8 c. or when the angle error is large positive and the tip deflection is also large positive as seen in Figure 4.8 d., a small torque is applied in the opposite direction that the link needs to move to reduce the angular position error as the link's inertia will pull it back towards the desired position but the tip will overshoot and begin oscillating. By applying this torque, it minimizes the potential energy stored in the link due to the deflection and enables the controller to then move the link to its proper position while minimizing oscillations.

The remaining rules are then interpolated by filling in the intermediate values. They can then be used with the membership function values to create fuzzy torque outputs. These are then defuzzified using the center of gravity method to create crisp outputs that have a smooth transition between values that can be seen in Figure 4.9. In the surface mesh, the two flat axis (Theta_Error and Delta) reference the values of the two inputs. Once those

inputs are read, the intersection point of those values can be found on the graph and the torque output is the z axis (vertical axis labelled Torque) value.

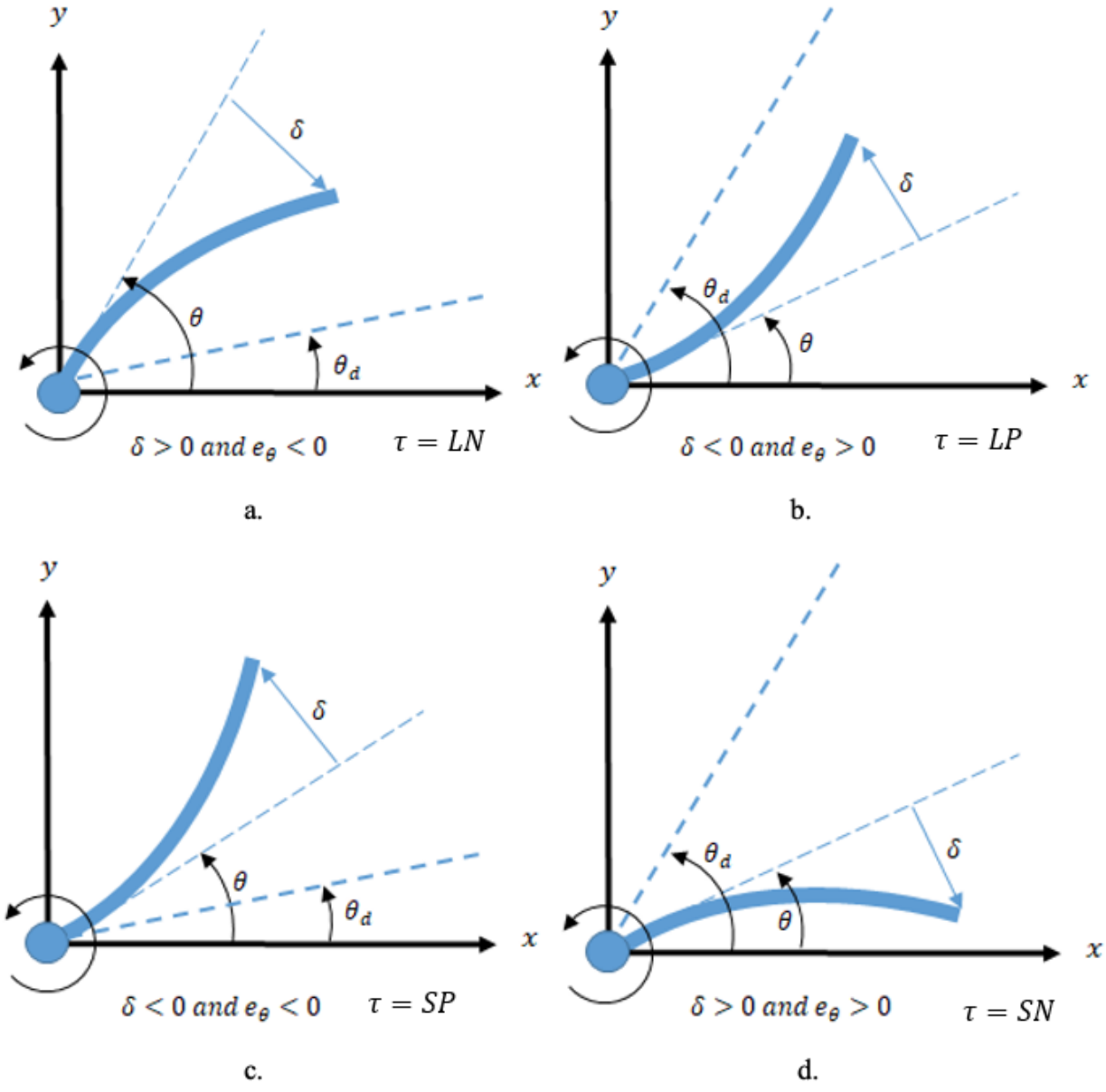

Figure 4.8: Extreme θ Error and δ Error Scenarios

Table 6: Fuzzy Logic Controller Rule Base

		Tip Deflection (δ)						
		LN	**MN**	**SN**	**Zero**	**SP**	**MP**	**LP**
Angle Error (θ Error)	**LP**	LP	LP	LP	LP	MP	SP	SN
	MP	LP	LP	LP	MP	MP	SP	SN
	SP	LP	LP	MP	SP	Zero	SN	MN
	Zero	LP	MP	SP	Zero	SN	MN	LN
	SN	MP	SP	Zero	SN	MN	MN	LN
	MN	SP	SN	MN	MN	LN	LN	LN
	LN	SP	SN	MN	LN	LN	LN	LN

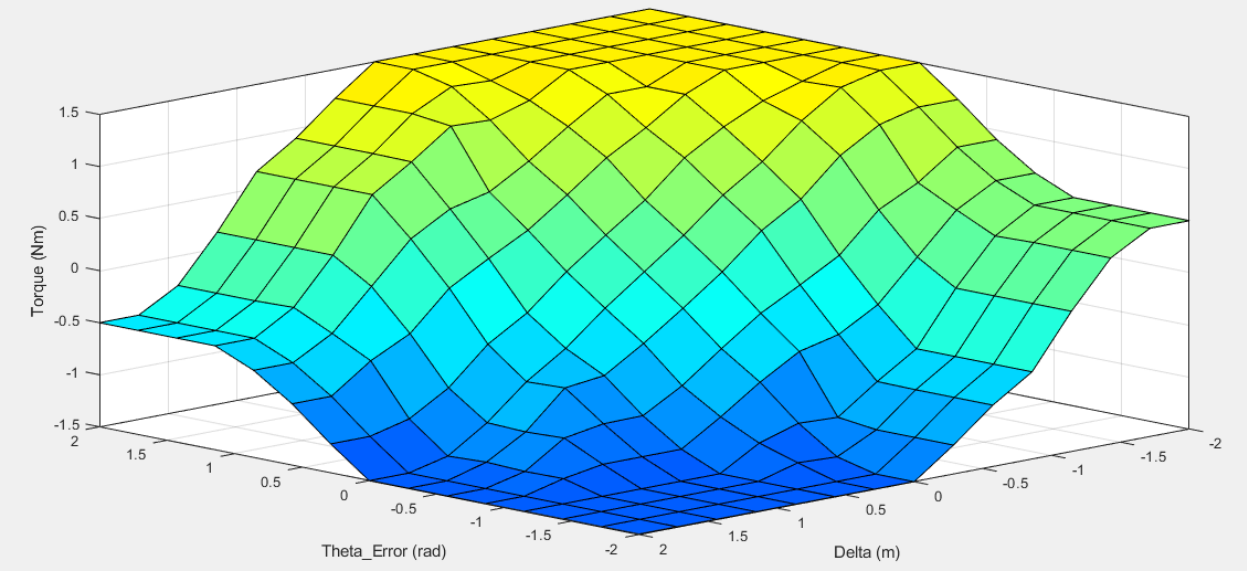

Figure 4.9: Surface Mesh of Controller

4.2.1 Simulated Controller Results

This Fuzzy Logic Controller was then applied to the Simulink model as seen in Figure 4.10 by using a 0.08 m initial tip displacement to determine if it is able to control the link.

The scaling factors for the Fuzzy Logic Controller were chosen as 20 for the angular position scaling factor so that a $\pi/32$ angular position input would fall within the Universe of Discourse. The scaling factor for the tip deflection was chosen to be 18 such that 10% of the total length of the link (0.112 m) would fall within the Universe of Discourse. Finally the torque scaling factor was chosen heuristically such that the torque was small enough to not add to the amplitude of the tip deflection after the initial motion and to ensure that the angular position did not exceed ± 0.2 rad.

The angular position error is the top input into the controller and the tip displacement is the bottom input into the controller in Figure 4.10. The results of these simulations can be seen in Figures 4.11 and 4.12.

While the vibration cannot be fully eliminated due to the lack of damping in the system, θ and δ were able to be reduced with the maximum tip deflection of 0.15 m when uncontrolled able to be reduced to 0.08 m after 30 seconds. While the link is still vibrating due to the lack of damping, the amplitude of the tip deflection is slowly reducing showing that the vibration is still being reduced. Therefore, the controller works properly within the simulation and the model can be updated with damping and the controller applied to the physical system.

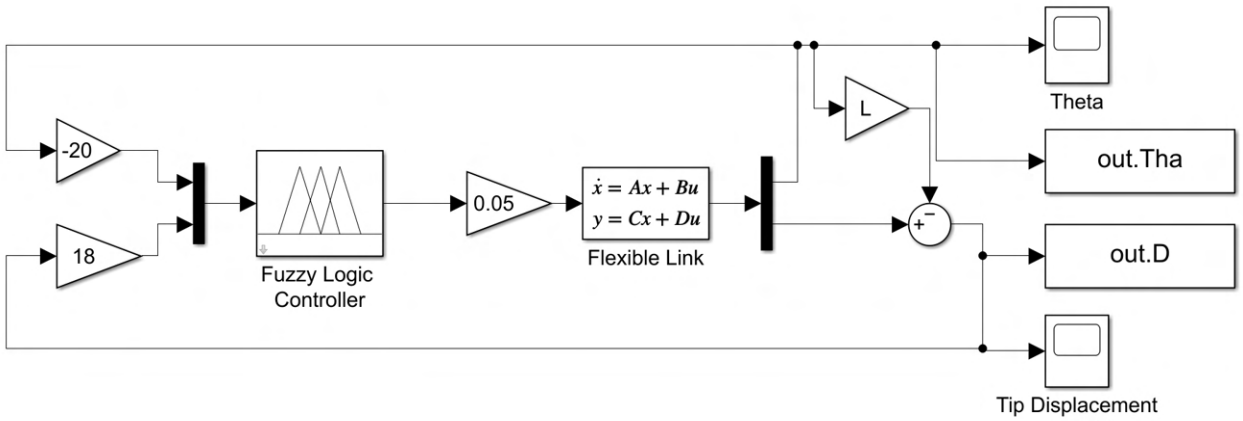

Figure 4.10: Simulink Model with Fuzzy Logic Controller Implemented

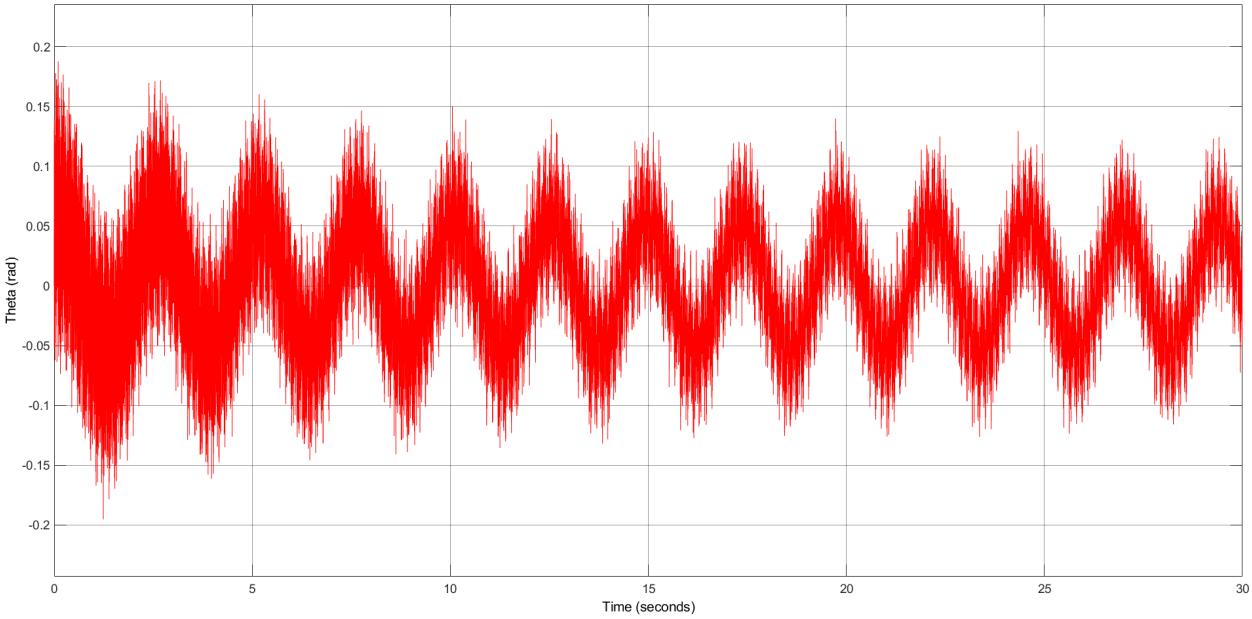

Figure 4.11: Undamped θ Response with Fuzzy Logic Controller

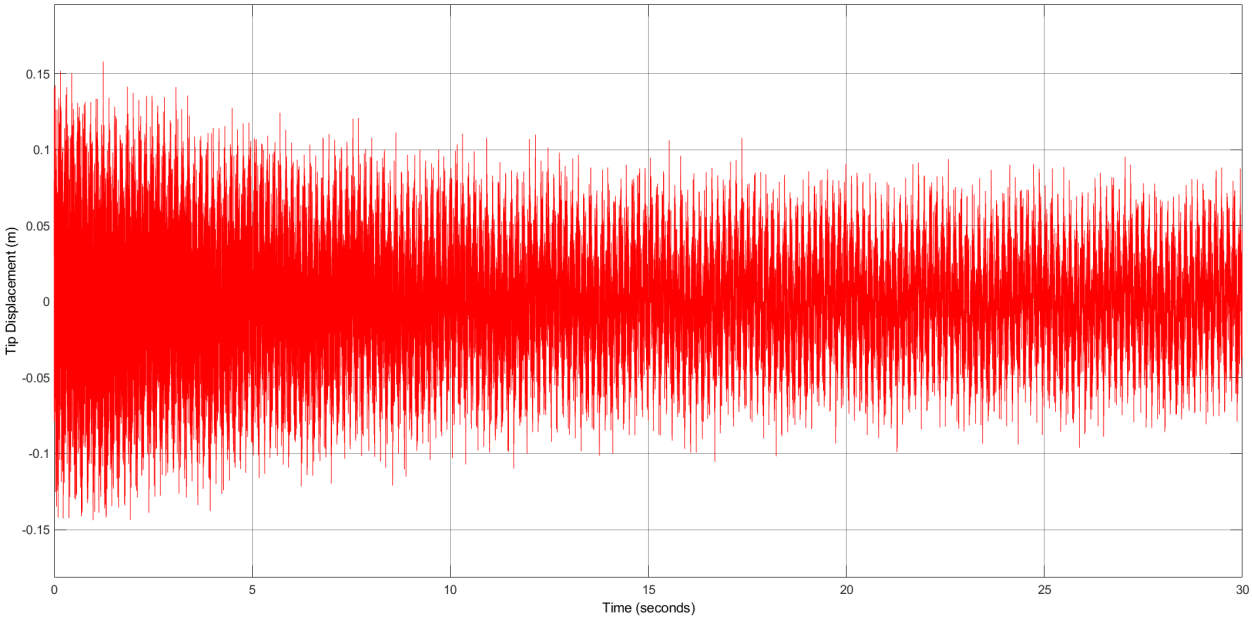

Figure 4.12: Undamped δ Response with Fuzzy Logic Controller

4.3 Updated Model

4.3.1 Adding Damping

Once the natural frequencies of the system have been verified with the experimental setup as seen in Figures 3.9 and 3.10, damping must be added to make the system more realistic. In order to do this, a Rayleigh damping model was used due to its efficacy as a damping matrix that is representative of the energy loss within the link [31].

The Rayleigh Damping Method uses a linear combination of the mass and stiffness matrices proportionally adjusted based on the damping ratio of the system and the natural frequencies affecting the vibration of the link seen in Equations 4.2 to 4.4 [31, 32].

The drawbacks of this method however is that this is only an estimation and is therefore not completely accurate when applied to the physical system. Also, as this only takes into account the energy loss in the link, it does not consider the energy loss caused by

the gears in the motor at the hub. This is still acceptable though as the controller can have the torque gain increased when moved to the physical system.

The damping matrix of the system is calculated as

$$[C_D] = \alpha[M] + \beta[K] \tag{4.2}$$

where $[C_D]$ is the damping matrix of the link, [M] is the mass matrix of the link, [K] is the stiffness matrix of the link, α is [31];

$$\alpha = 2\zeta \frac{(\omega_{max})(\omega_{min})}{\omega_{max} + \omega_{min}} \tag{4.3}$$

and β is;

$$\beta = \frac{2\zeta}{\omega_{max} + \omega_{min}} \tag{4.4}$$

where ζ is the damping ratio of the material, ω_{max} is the largest natural frequency that is being examined in the response of the link, and ω_{min} is the lowest natural frequency that affects the response of the link, not including the rigid body motion.

An estimated ζ for aluminum is required. This will represent how much energy is lost due to the vibration of the link itself and does not take into account outside factors such as air resistance or noise. According to Havaldara and Chate [33], ζ can be calculated using the shear modulus and density of the material which is G=25 GPa and ρ=2700 $\frac{kg}{m^3}$ for aluminum [34] and then applied to Equation 4.5.

$$\zeta = 1.2\left(\frac{G}{\rho}\right)^{-\frac{1}{3}} = 0.0057 \tag{4.5}$$

The coefficients α and β are calculated using ω_{min} and ω_{max} as described in Equations 4.3

and 4.4. The first four natural frequencies of the system were used as they are the frequencies that are being analyzed in the system. Using ω_{min}=41.19 rad/s and ω_{max}=476.30 rad/s the values of α and β based on Equations 4.3 and 4.4 respectively are

$$\alpha = 0.4322 \tag{4.6}$$

$$\beta = 0.00002203 \tag{4.7}$$

By substituting these values into Equation 4.2, the damping matrix is

$$[C_D] = 0.4322[M] + 0.00002203[K] \tag{4.8}$$

Finally, this matrix is added to the State-Space representation of the system within the A matrix of the system. This gives the final model of the system of:

$$\dot{X} = AX + Bu$$

$$A = \begin{bmatrix} [0]_{13X13} & [I]_{13X13} \\ -[M]^{-1}[K] & -[M]^{-1}[C_D] \end{bmatrix}$$

$$B = \begin{bmatrix} [0]_{13X1} \\ [M]^{-1}[B^*] \end{bmatrix}$$

This damping matrix was then added to the Simulink model. An initial tip displacement of 0.1 m was given to the link to ensure that the damping did reduce the vibration over time due to the energy loss in the system from the structural damping which gave the results seen in Figures 4.13 and 4.14. The tip displacement dies out slowly over time centered on the zero position which is what is expected as the damping ratio of the link caused by the material's structural damping is small. Alternatively, θ does not finish

on the zero position due to the link being able to rotate and so the inertia of the link causes its rigid body motion to move the link off of its starting point and it does not have enough energy to return to its starting position. There will be additional energy loss in θ caused by friction in the gears within the motor that is not captured here so when the controller is applied to the physical system, the amount of torque required to move the link will have to be increased.

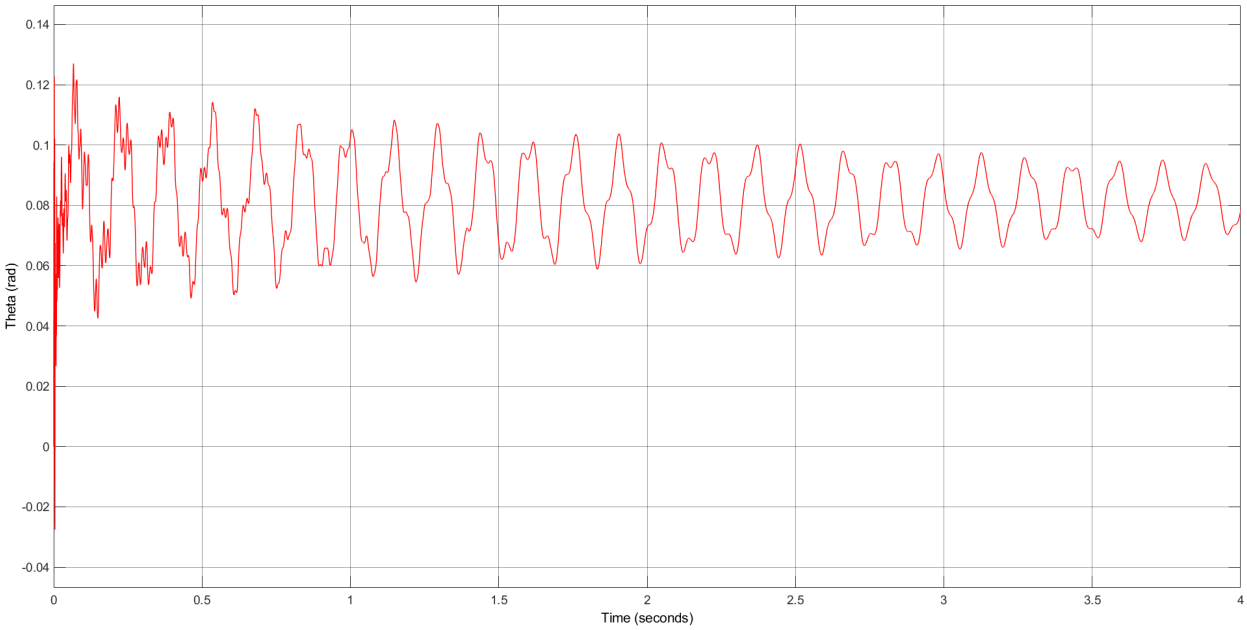

Figure 4.13: Damped θ Free Vibration Response

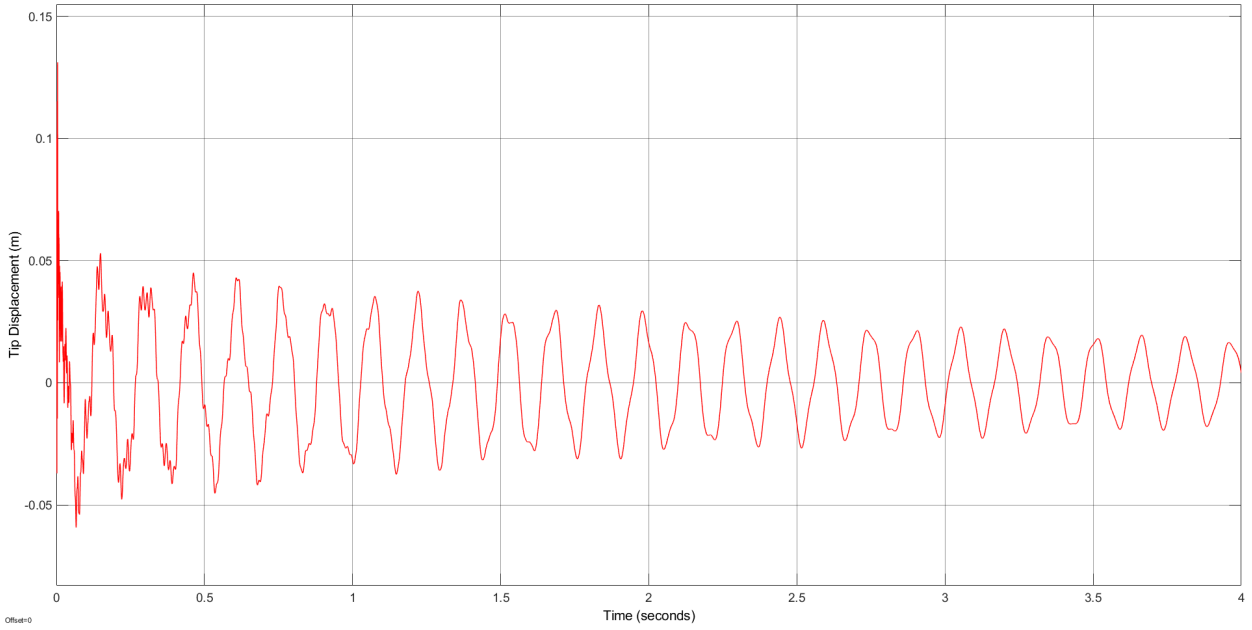

Figure 4.14: Damped δ Free Vibration Response

The damping ratio within the link model can be estimated by using the logarithmic decrement calculation using the data from Figure 4.14 [20]. The logarithmic decrement, Δ, is equal to:

$$\Delta = ln\left(\frac{x_n}{x_{n+1}}\right) = \frac{2\pi\zeta}{\sqrt{1-\zeta^2}} \tag{4.9}$$

where x_n is the n^{th} peak in amplitude of the response and ζ is the damping ratio. Due to a small structural damping being used where $\zeta << 1$, Equation 4.9 can be simplified as

$$\Delta = ln\left(\frac{x_n}{x_{n+1}}\right) = 2\pi\zeta \tag{4.10}$$

$$ln\left(\frac{0.045}{0.043}\right) = 2\pi\zeta \tag{4.11}$$

$$\zeta = 0.0072 \tag{4.12}$$

This damping coefficient of 0.0072 is different from the previously estimated value of

0.0057 due to the smaller value only representing the structural damping of the material whereas the larger value of 0.0072 represents the overall damping in the system.

The damping was then added to the model with the Fuzzy Logic Controller with an initial tip displacement of 0.1 m. The gain values used can be seen in Figure 4.15 and the results can be seen in Figures 4.16 and 4.17 where θ is able to be brought back to oscillating about the zero position and δ is reduced to under 1 mm of deflection within 7.7 seconds.

The scaling factors for the Fuzzy Logic Controller were chosen as 20 for the angular position scaling factor so that a $\pi/32$ angular position input would fall within the Universe of Discourse. The scaling factor for the tip deflection was chosen to be 18 such that 10% of the total length of the link (0.112 m) would fall within the Universe of Discourse. Finally the torque scaling factor was chosen heuristically such that the torque was small enough to not add to the amplitude of the tip deflection after the initial motion.

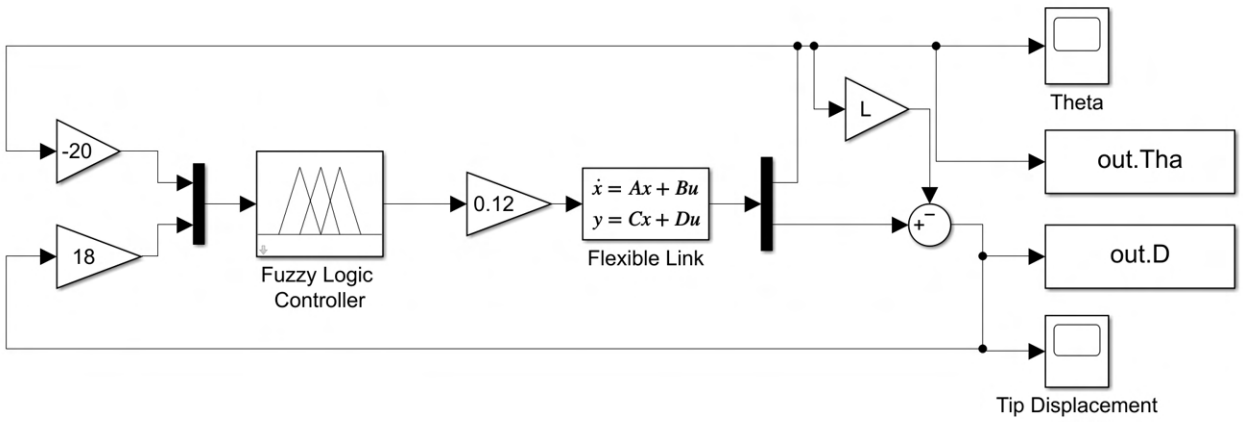

Figure 4.15: Simulink Model with Fuzzy Logic Controller

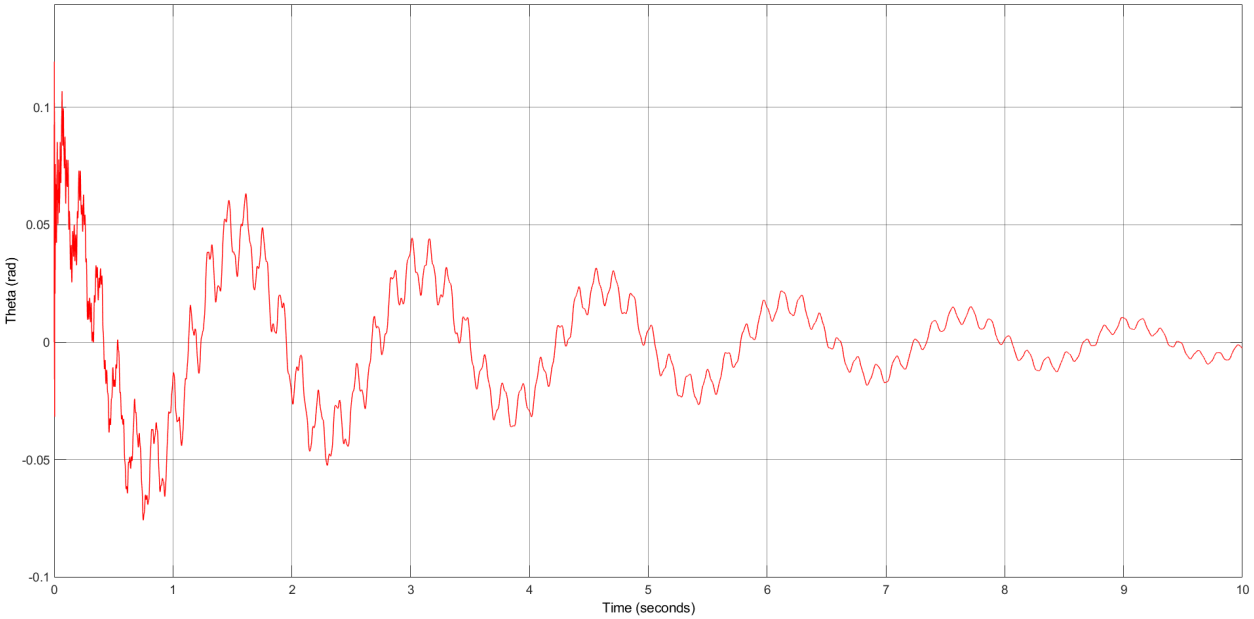

Figure 4.16: Damped θ Response with Fuzzy Logic Controller

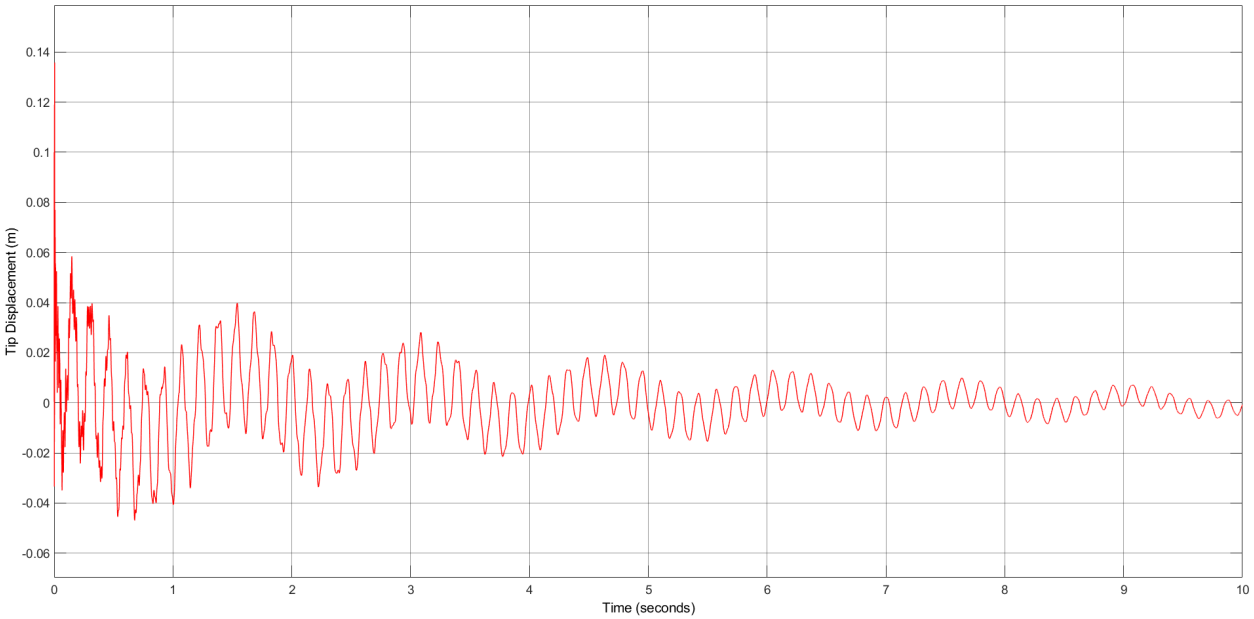

Figure 4.17: Damped δ Response with Fuzzy Logic Controller

4.3.2 Motor Inertia

Another modification to make the model more accurate is the addition of the mass moment of inertia acting on the base of the link. This adds resistance to motion caused by the

additional mass as well as energy loss in the motor due to internal friction. Due to the damping matrix using a combination of the mass and stiffness matrices, this matrix does not need to be updated. However the mass matrix must be changed by adding the motor inertia to the first element of the mass matrix (the effect of the mass on the angular displacement of the base, because the mass is located at the base). The motor inertia was listed as 0.000016 kgm^2 by the manufacturer [35] and the attachment block used to attach the link to the motor has the dimensions 0.09525 m by 0.0508 m by 0.0127 m with a density of 7850 kg/m^3 [36]. The motor inertia was then calculated as:

$$J_{base} = J_{motor} + J_{block} \tag{4.13}$$

$$J_{base} = 0.000016 + \frac{m_{block}}{12}(base^2) \tag{4.14}$$

$$J_{base} = 0.000016 + \frac{(7850)(0.09525)(0.0508)(0.0127)}{12}(0.0508^2) = 0.00012kg/m^2 \tag{4.15}$$

The link was given an initial tip deflection of 0.1 m with the inertia of the base added and the results can be seen in Figures 4.18 and 4.19. The additional inertia at the base of the link caused the θ response of the link to only overshoot the attain an angular position of 0.111 rad as opposed to before the motor inertia was added in Figure 4.16 where the angular position of the link reached 0.119 rad. The maximum tip deflection was also lower with the motor inertia added, never going above the initial condition of 0.1 m whereas without the motor inertia, due to the initial motion of the link, the tip deflection briefly increased to 0.136 m.

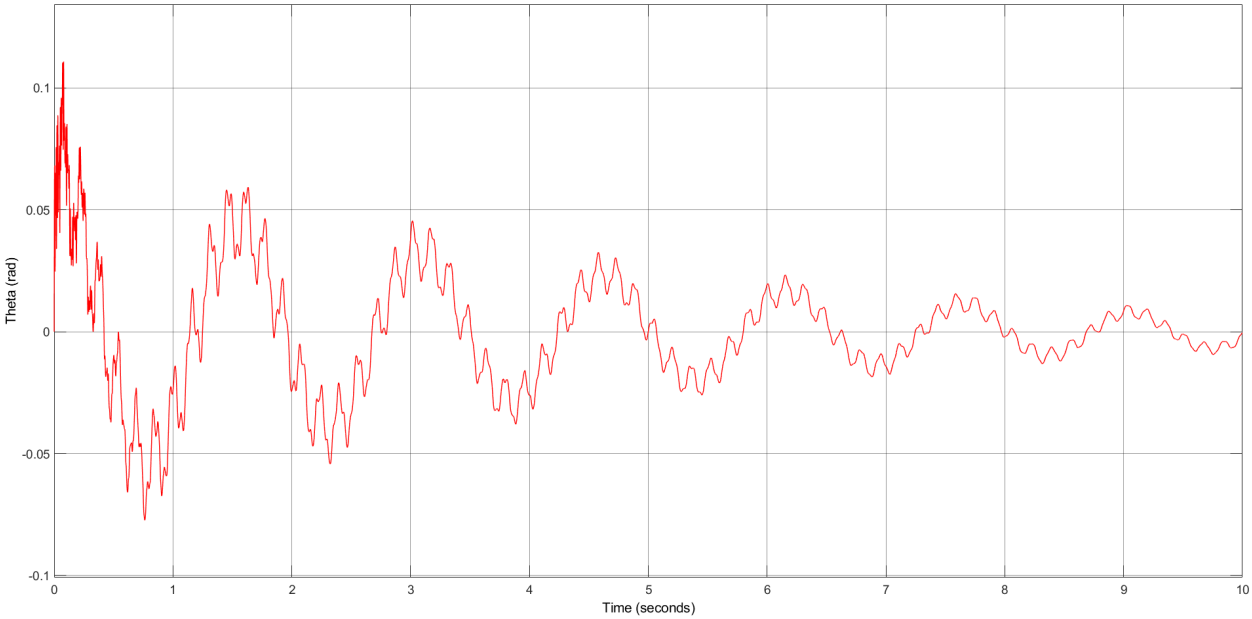

Figure 4.18: Damped θ Response with Fuzzy Logic Controller and Motor Inertia Added

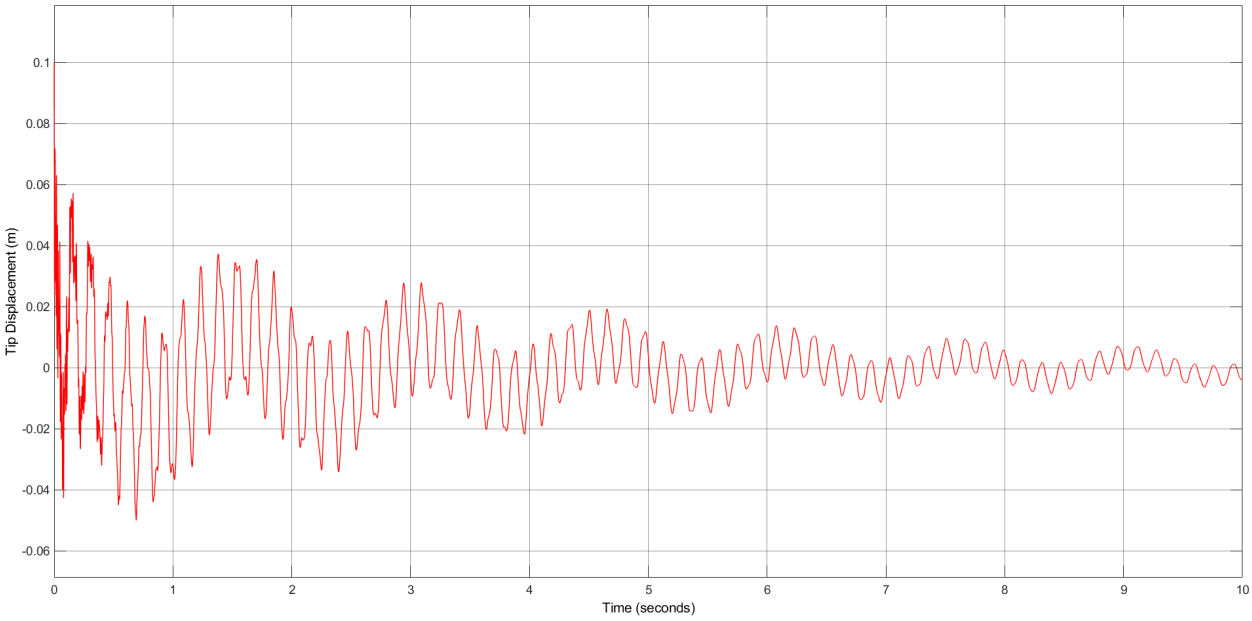

Figure 4.19: Damped δ Response with Fuzzy Logic Controller and Motor Inertia Added

4.3.3 Updated Controller Results in Simulink

Now that the controller has been verified as functional as seen in Figures 4.18 and 4.19, it must be determined how effective it is at controlling the vibration by comparing it with the

uncontrolled response. This is done by running a MATLAB simulation using a proportional controller followed by the Fuzzy Logic Controller with the same input.

The proportional gain used was determined heuristically to ensure that at inputs up to $\pi/2$ the settling-time for the proportional controller remained under 3 seconds. This was done by testing different values of the proportional controller with the block diagram seen in Figure 4.20 with the physical system, resulting in a proportional gain value of 22. This block diagram consists of:

- a signal generator which is the desired angular position (Figure 4.20 - Block 1),
- a proportional gain to scale the angular position error (Figure 4.20 - Block 2),
- an analog output which sends a control signal to apply torque to the motor at the joint (Figure 4.20 - Block 3),
- a block to read the data from the encoder (Figure 4.20 - Block 4),
- a gain block to calibrate the encoder signal so that the signal measures radians (Figure 4.20 - Block 5),
- a block that stores the angular position data to the workspace (Figure 4.20 - Block 6),
- a scope to view the θ response and desired θ response of the link (Figure 4.20 - Block 7),
- a block that stores the desired angular position data to the workspace (Figure 4.20 - Block 8), and
- a Hardware-in-the-Loop Initialize block which initializes the Hardware-in-the-Loop card (Figure 4.20 - Block 9).

A more detailed description of the hardware set-up to understand how the block diagram in Figure 4.20 is integrated with the physical system can be found in Section 5.1.

The conversion of the encoder signal to radians is calculated as [37]

$$\theta = \frac{n_{pulses}\theta_{max}}{4N} \tag{4.16}$$

where n_{pulses} is the number of pulses that the encoder reads, N is the number of windows in the encoder which is only multiplied by four if the encoder is a two or three channel encoder due to the second channel being offset by half of the phase, enabling four counts per window and θ_{max} is the maximum angle that can be read after one full rotation of the encoder disk. While this is often equal to 2π, if there is a gear ratio in the motor which affects the output rotation, θ_{max} will be equal to 2π divided by the gear ratio [37].

The encoder and motor used with this flexible link were a 3 channel encoder with 500 windows and a motor gear ratio of 65.5 [35] which means that the conversion from number of pulses to the angle in radians is

$$\theta = \frac{n_{pulses}2\pi}{4(500)(65.5)} = \frac{n_{pulses}2\pi}{131000} \tag{4.17}$$

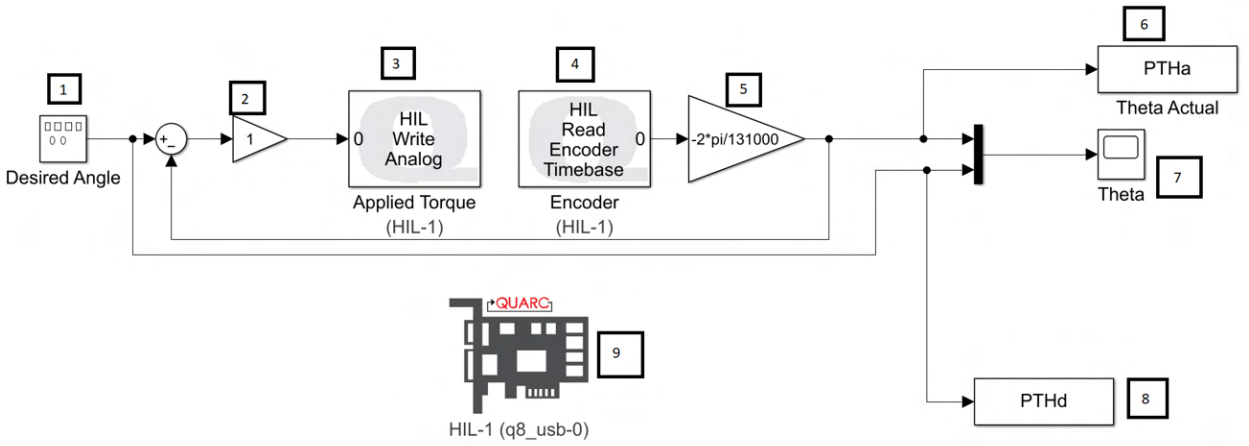

Figure 4.20: Simulink Block Diagram for Proportional Controller Analysis

To compare the uncontrolled response with the response using the Fuzzy Logic Controller, the MATLAB simulation shown in Figure 4.21 which uses a proportional controller was used with a desired θ input of $\pi/32$, followed by the Fuzzy Logic Controller model that was shown in Figure 4.22 with the same input. The most important factor when evaluating the performance of the Fuzzy Logic Controller was its ability to reduce the tip deflection of the link. It was deemed acceptable to be willing to sacrifice some speed in achieving the steady-state response of the link under Fuzzy Logic Control in exchange for reducing the tip deflection.

The scaling factors for the Fuzzy Logic Controller were chosen as 30 for the angular position error by starting with a value of 20 so that the $\pi/32$ angular position input would fall within the Universe of Discourse and then adjusted heuristically. The scaling factor for the tip deflection was chosen to be 20 by starting at 18 such that 10% of the total length of the link (0.112 m) would fall within the Universe of Discourse and then adjusted heuristically. Finally the torque scaling factor was chosen as 1 which was determined by starting with a value of 1.5 so that the maximum proportional torque would be comparable to the maximum Fuzzy Logic Controller torque with a $\pi/32$ desired angular position input (2.25 Nm for the proportional controller VS 2.16 Nm for the Fuzzy Logic Controller) and

then adjusted heuristically.

The results of the θ and δ responses can be seen in Figures 4.23 and 4.24 respectively. The Fuzzy Logic Controller had a maximum tip deflection of 0.074 m and was reduced to less than 5 mm of oscillation within 11.4 seconds. The Fuzzy Logic Controller did have a large maximum overshoot with the maximum angular position achieved being 0.158 rads (a 60.94% overshoot).

This compares with the proportional controller which also overshot the desired angle with a maximum angle achieved of 0.152 rads (a 54.83% overshoot). The proportional controller has substantially more tip deflection with a maximum tip deflection of 0.129 m. It also takes longer for the tip deflection to be reduced with 5 mm of tip deflection not being achieved until 13.9 seconds.

Ultimately, the Fuzzy Logic Controller shows a 17.99% decrease in time to minimize tip deflection and a 42.64% reduction in maximum tip deflection as seen in Figure 4.24.

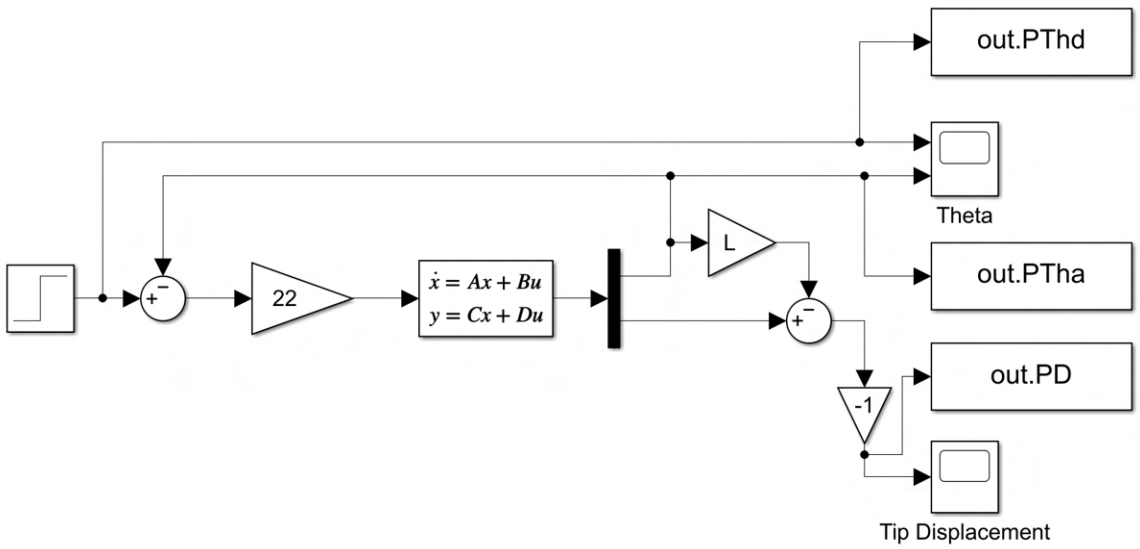

Figure 4.21: Simulink Block Diagram with Proportional Controller

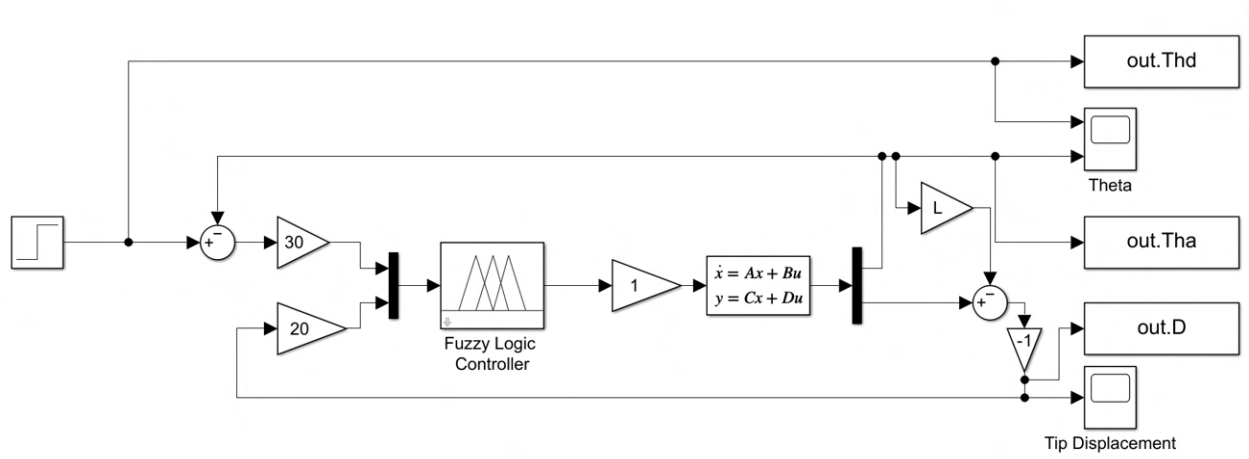

Figure 4.22: Simulink Block Diagram with Fuzzy Logic Controller

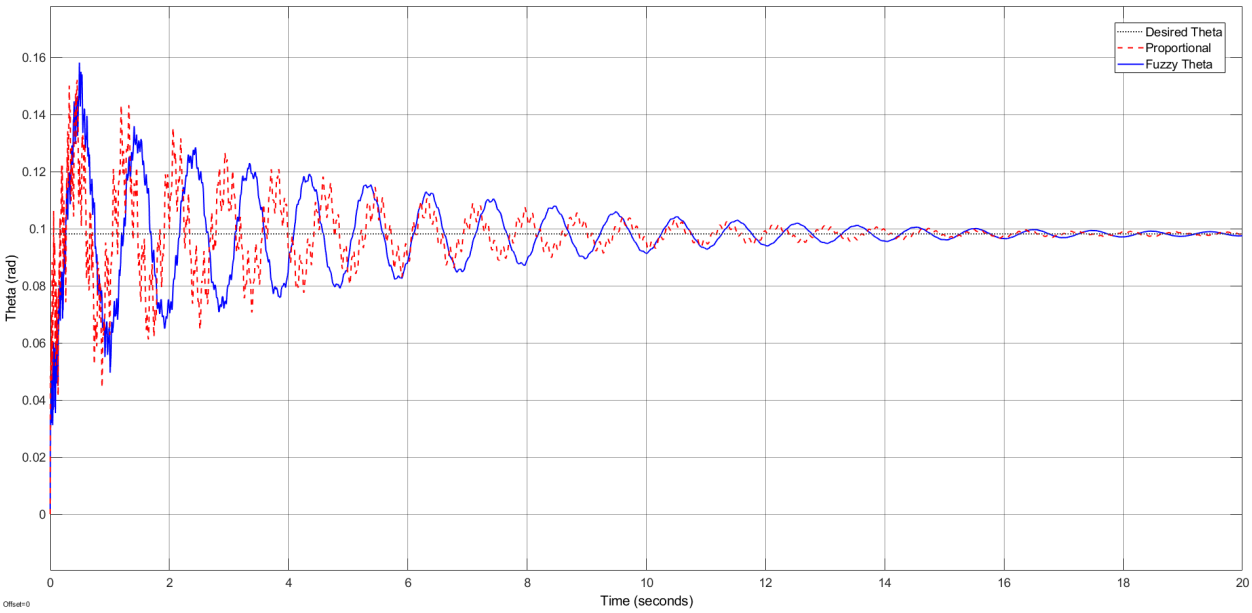

Figure 4.23: Simulation θ Response Due to a Step Input of $\pi/32$

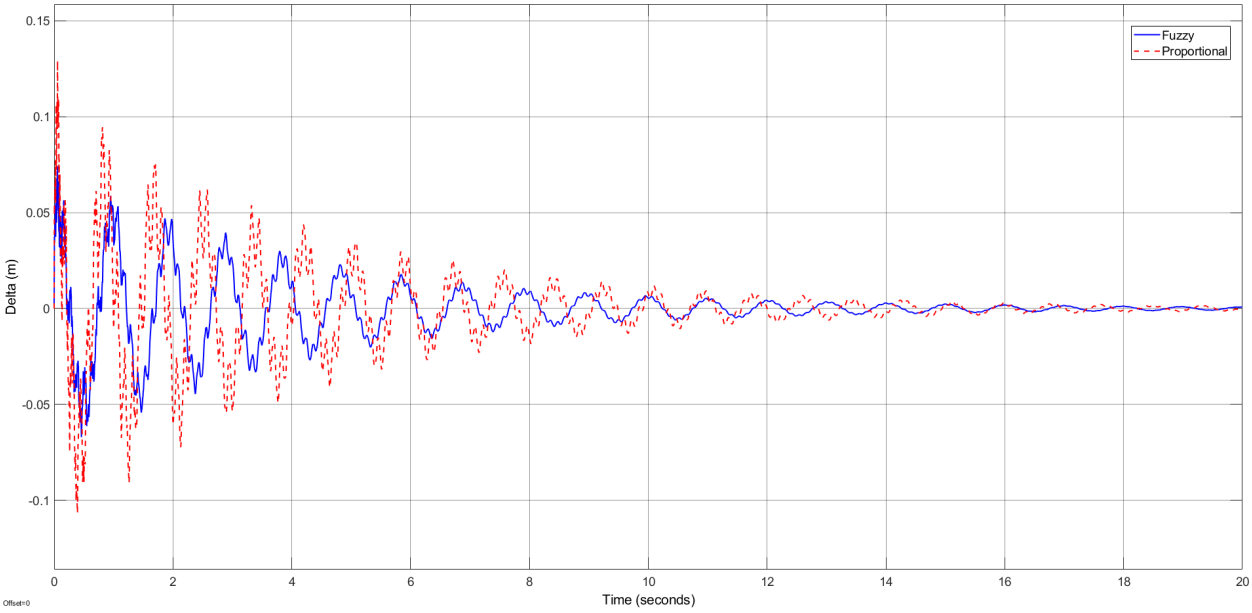

Figure 4.24: Simulation δ Response Due to a Step Input of $\pi/32$

4.3.4 Simscape Model Results

The controller was also compared against the uncontrolled response in the Simscape model that was created. The same proportional gain was used as seen in Figure 4.25. This was compared against the Fuzzy Logic Controller as seen in Figure 4.26. The scaling factors used in Simulink were used as the starting point for the Simscape simulations and were adjusted heuristically to obtain the scaling factors that were utilized. Just as with the Simulink model, a desired input of $\pi/32$ rads was used which produced the results shown in Figures 4.27 and 4.28. The Fuzzy Logic Controller was able to reduce the amount of tip deflection in the link with the maximum tip deflection of the proportional controller being 0.137 m whereas the Fuzzy Logic controller had a 53.28% decrease in maximum tip deflection, only reaching 0.064 m. The Fuzzy Logic Controller also maintained a lower tip deflection throughout operation, having less than 0.01 m of tip deflection after only 17.8 seconds compared with the proportional controller taking 24.2 seconds. The trade off for this reduced tip deflection however was an increased θ response with a maximum overshoot at 0.0178 rads greater than the proportional controller.

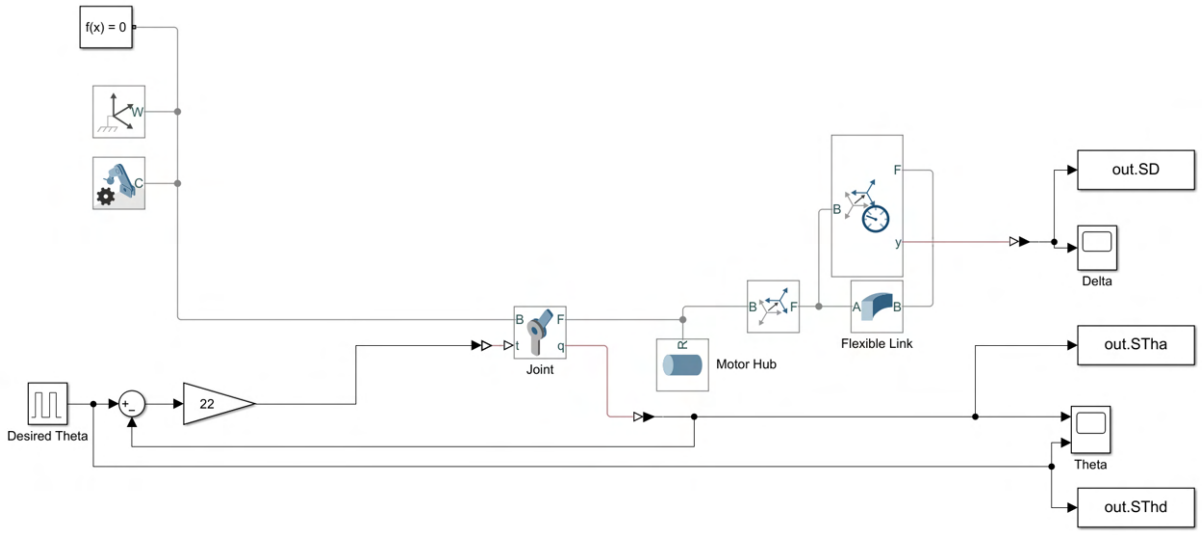

Figure 4.25: Simscape Block Diagram with Proportional Controller

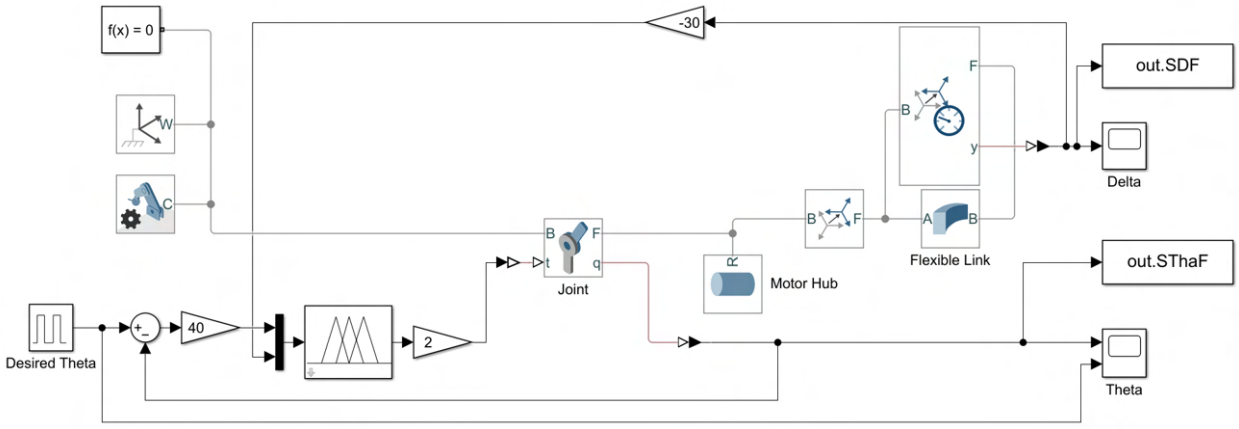

Figure 4.26: Simscape Setup with Fuzzy Logic Controller

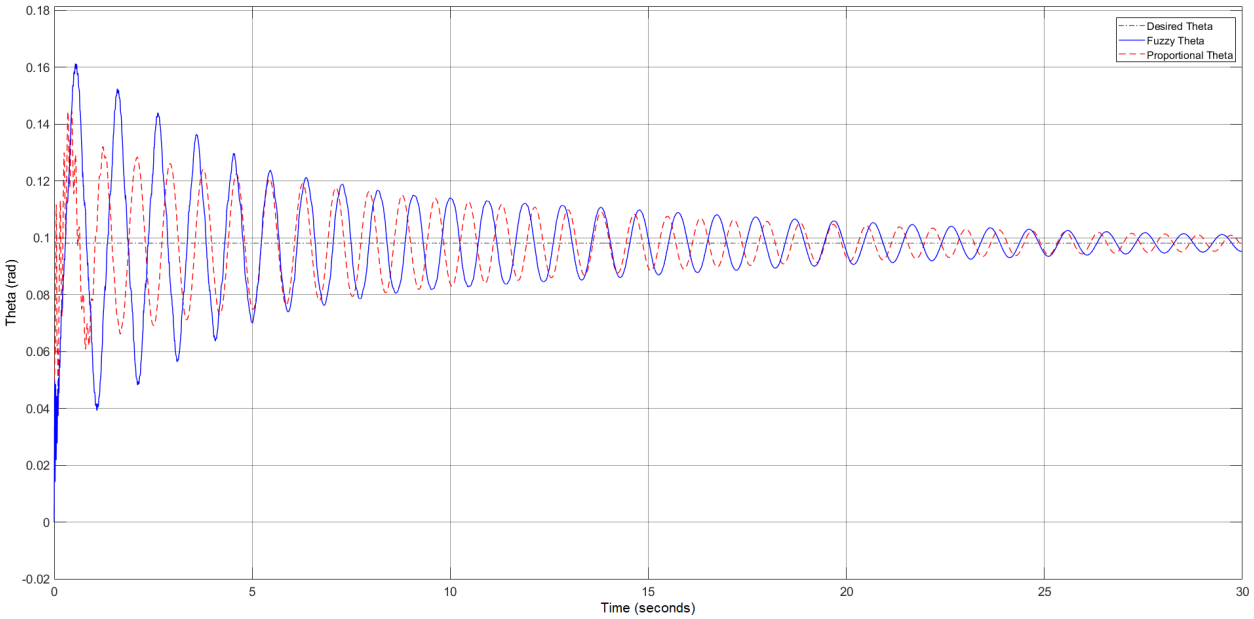

Figure 4.27: Simscape θ Response Due to a Step Input of $\pi/32$

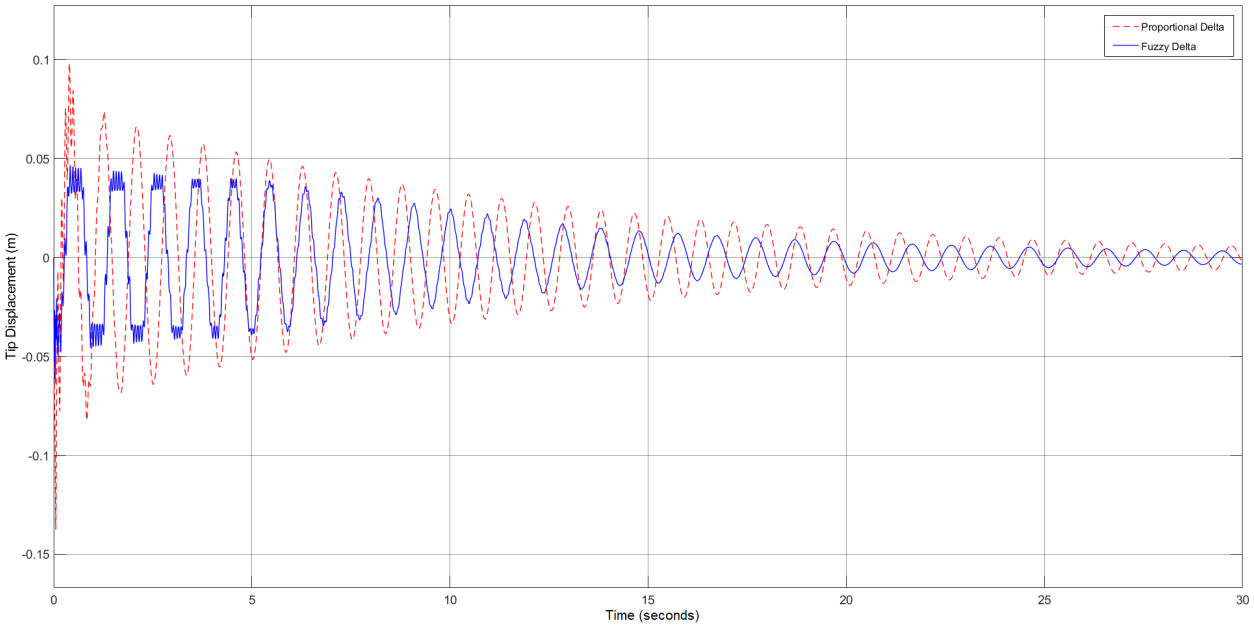

Figure 4.28: Simscape δ Response Due to a Step Input of $\pi/32$

In order to get a wider range of inputs to ensure that the Fuzzy Logic Controller remains effective at reducing vibration in the link, the Fuzzy Logic Controller and proportional controller were used in the Simscape model with a ramp input moving from 0 to $\pi/8$ in 2 seconds and upon reaching $\pi/8$, the signal flattens to remain constant to avoid

the large initial torque that the step input causes so that the proportional controller does not have as large of a tip deflection. The Fuzzy Logic Controller and proportional controller were also subjected to a ramp input of 0 to $\pi/2$ in 2 seconds where upon reaching $\pi/2$, the signal flattens to remain constant. The responses from these inputs can be seen in Figures 4.29 to 4.32 and Table 7 and the gain values used with the Fuzzy Logic Controller were 30 for the angular position gain, 20 for the tip displacement gain, and 2 for the torque gain.

Using the Fuzzy Logic Controller with these ramp inputs, not only is the maximum tip displacement reduced by up to 61.64% but upon reaching the desired angle, while the tip displacement of the link under proportional control increases, the Fuzzy Logic Controller is able to reduce the tip deflection even more and the Fuzzy Logic Controller has less overshoot of the desired position than the proportional controller by up to 3.59%.

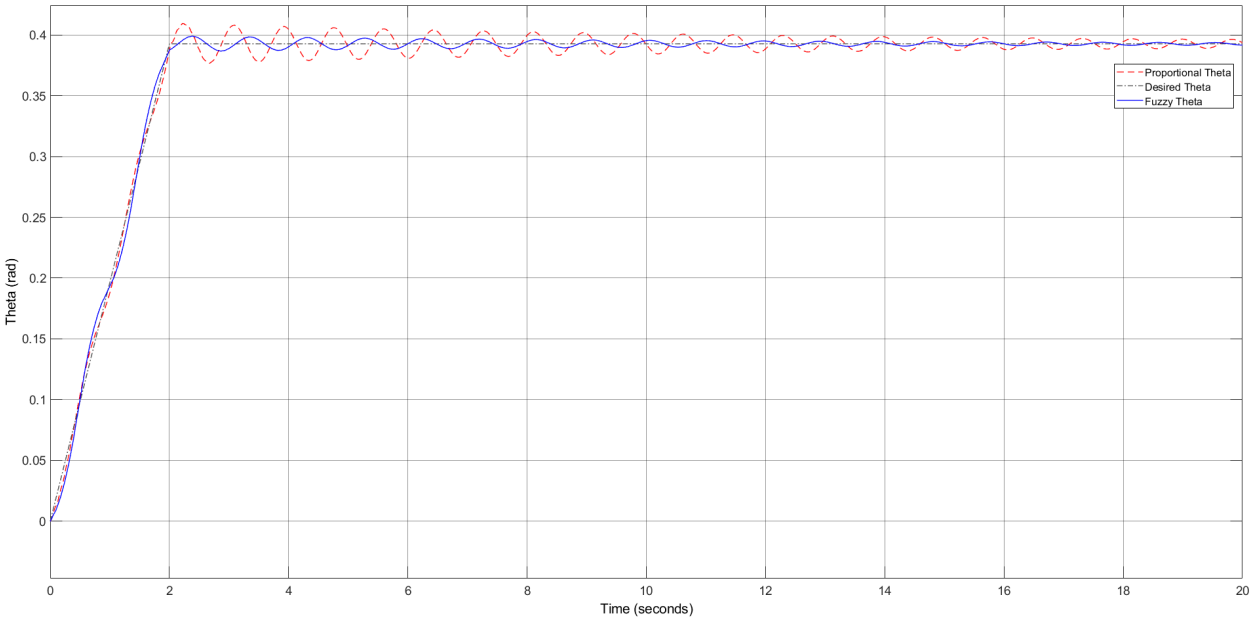

Figure 4.29: Simscape θ Response Due to a Ramp Input from 0 to $\pi/8$ over 2 seconds

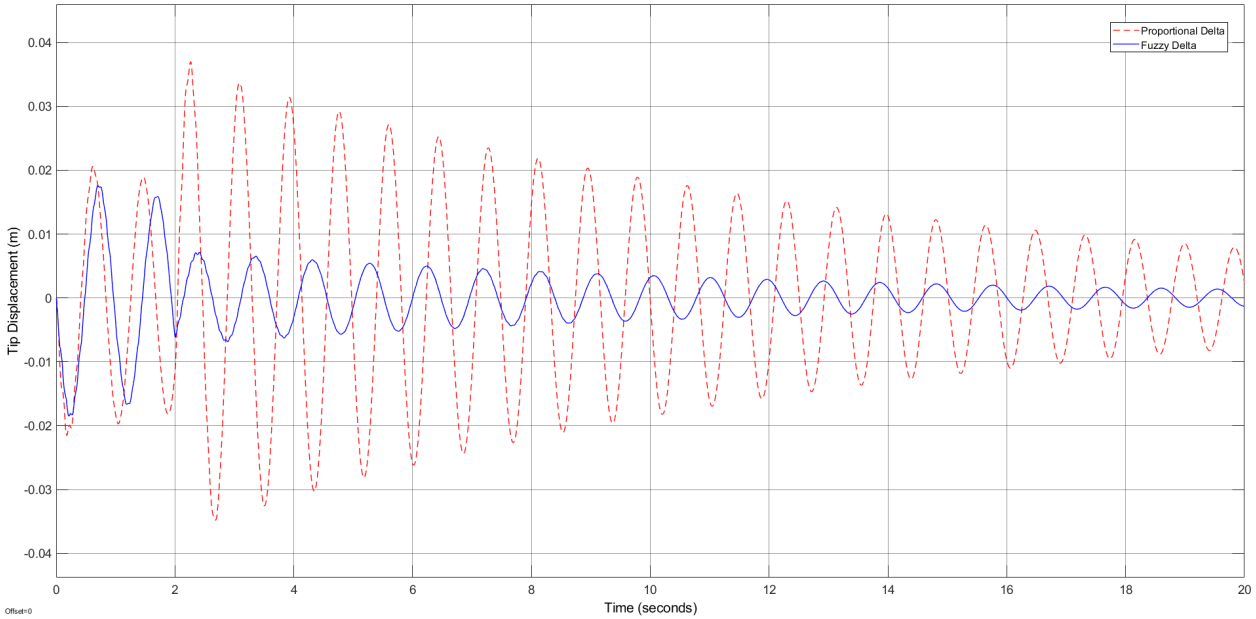

Figure 4.30: Simscape δ Response Due to a Ramp Input from 0 to $\pi/8$ over 2 seconds

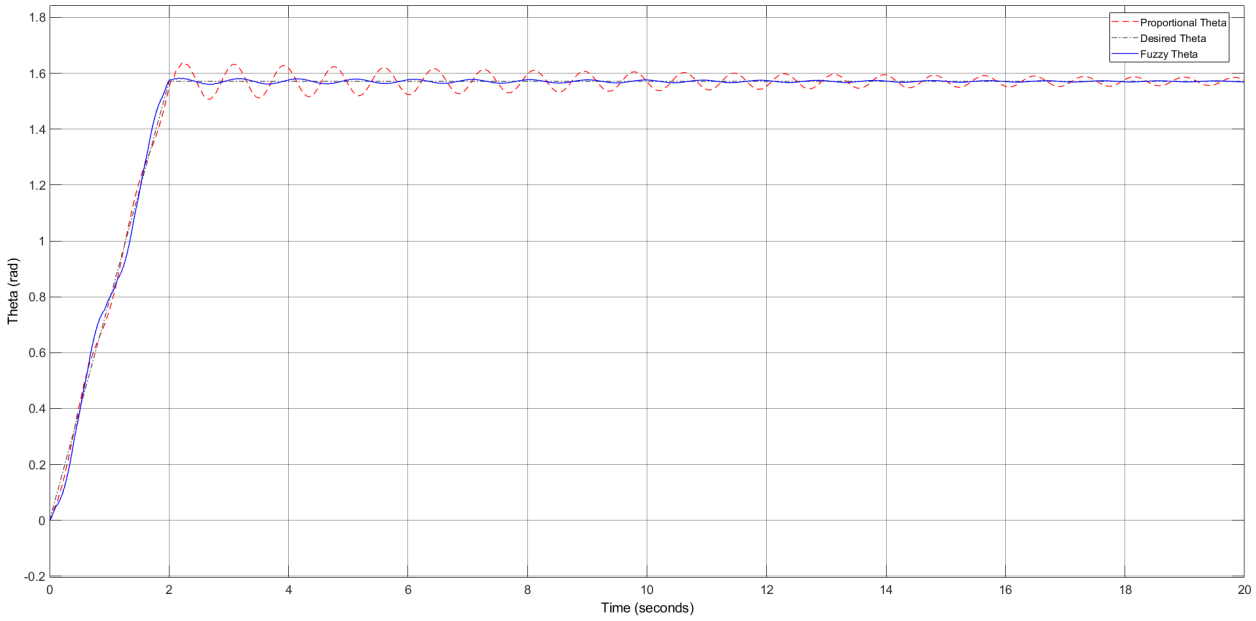

Figure 4.31: Simscape θ Response Due to a Ramp Input from 0 to $\pi/2$ over 2 seconds

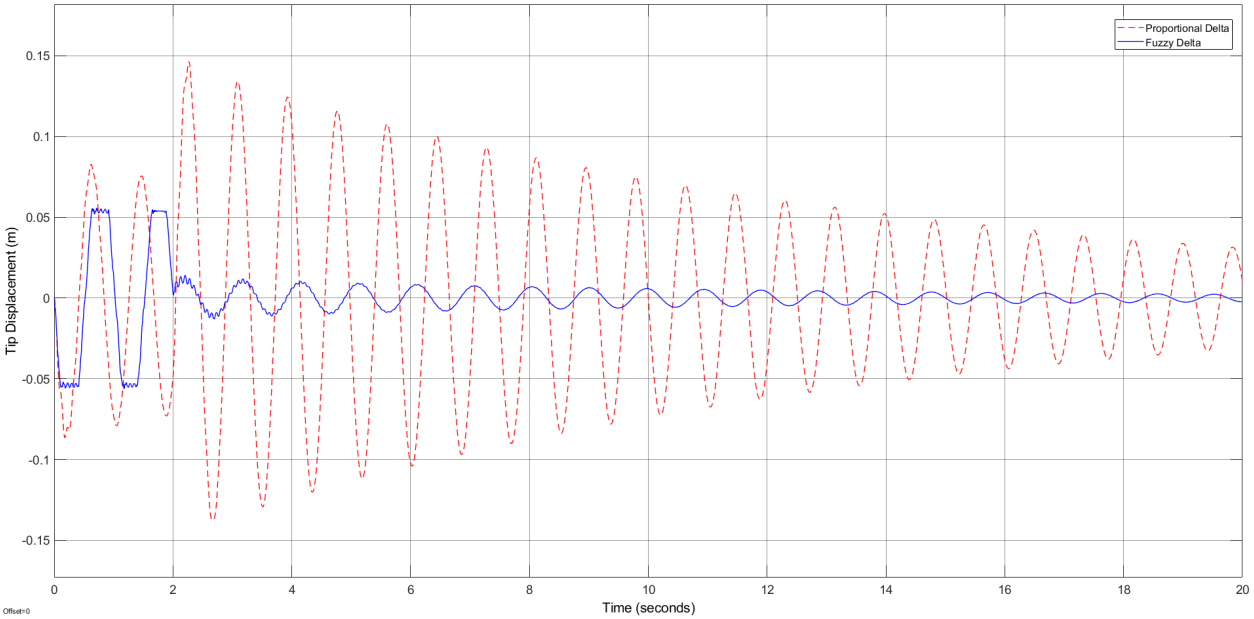

Figure 4.32: Simscape δ Response Due to a Ramp Input from 0 to $\pi/2$ over 2 seconds

Table 7: Simscape Data

	π/**32 Input**	π/**8 Input**	π/**2 Input**
Proportional Controller Percent Overshoot (%)	45.86	4.31	4.30
Fuzzy Logic Controller Percent Overshoot (%)	63.99	1.60	0.71
Proportional Controller Max Tip Displacement (m)	0.137	0.037	0.146
Fuzzy Logic Controller Max Tip Displacement (m)	0.064	0.018	0.056
δ Percent Improvement (%)	53.28	51.35	61.64

4.4 Controller Optimization

In this section, the Rule Base of the Fuzzy Logic Controller was modified to create two additional Fuzzy Logic Controllers with slightly different Rule Bases. These controllers were then given the same inputs and compared against each other to determine which was the best controller to use during practical testing. The most important response for the controllers was minimizing the tip deflection.

4.4.1 Controller Variations

These controllers kept the expert knowledge that was used to design the Fuzzy Logic Controller presented in Section 4.2 however a different Rule Base was used as seen in Tables 8 and 9. These Rule Bases were chosen based on a heuristic analysis of extreme cases as well as an adjustment to the intermediary rules such that there was a more gradual change between torque values. An input of $\pi/32$ was given to the link. During normal operation of the controller, the extreme cases are not activated very often unless an external force is applied to the link such as by giving the link an initial tip deflection.

Table 8: Fuzzy Logic Controller Iteration Two of Rule Base

		Tip Deflection (δ)						
		LN	**MN**	**SN**	**Zero**	**SP**	**MP**	**LP**
Angle Error (θ Error)	**LP**	LP	LP	LP	LP	MP	SP	Zero
	MP	LP	LP	LP	MP	MP	Zero	SN
	SP	LP	LP	MP	SP	Zero	SN	MN
	Zero	LP	MP	SP	Zero	SN	MN	LN
	SN	MP	SP	Zero	SN	MN	MN	LN
	MN	SP	Zero	MN	MN	LN	LN	LN
	LN	Zero	SN	MN	LN	LN	LN	LN

Table 9: Fuzzy Logic Controller Iteration Three of Rule Base

		Tip Deflection (δ)						
		LN	**MN**	**SN**	**Zero**	**SP**	**MP**	**LP**
Angle Error (θ Error)	**LP**	LP	LP	LP	MP	MP	SP	Zero
	MP	LP	LP	MP	MP	SP	Zero	SN
	SP	LP	MP	MP	SP	Zero	SN	MN
	Zero	MP	MP	SP	Zero	SN	MN	MN
	SN	MP	SP	Zero	SN	MN	MN	LN
	MN	SP	Zero	SN	MN	MN	LN	LN
	LN	Zero	SN	MN	MN	LN	LN	LN

4.4.2 Controller Comparison

These controllers were then used within Simulink with a $\pi/32$ input as seen in Figure 4.33 using the same scaling factors that were used to determine the Fuzzy Logic Controller's ability to reduce vibration in Section 4.3.3. The results of the three controllers can be seen in Figures 4.34 and 4.35. Figures 4.34 and 4.35 show that all three Controllers performed similarly.

The primary goal of the controller is generally to minimize tip deflection and then to move the link into the proper angular position while minimizing the creation of additional vibration. This means that the extreme rules (large for both e_θ and δ) do not usually occur unless there is an external force applied to the system, causing them to deviate greatly from the desired position. While all three controllers performed similarly, it can be seen that there are differences in the motion. Controller 1 reduces the tip deflection more aggressively as it

prioritizes reducing the vibration as opposed to Controllers 2 and 3 which allow the link's stiffness to reduce the tip deflection, thus enabling Controller 1 to have less tip deflection. While the difference in values are small, Controller 1 has a maximum tip deflection of 0.0742 m, but Controller 2 has a maximum tip deflection of 0.0774 m and Controller 3 has a maximum tip deflection of 0.0751 m. Therefore Controller 1 is the best suited to carry forward into practical testing however Controllers 2 and 3 are still functional controllers.

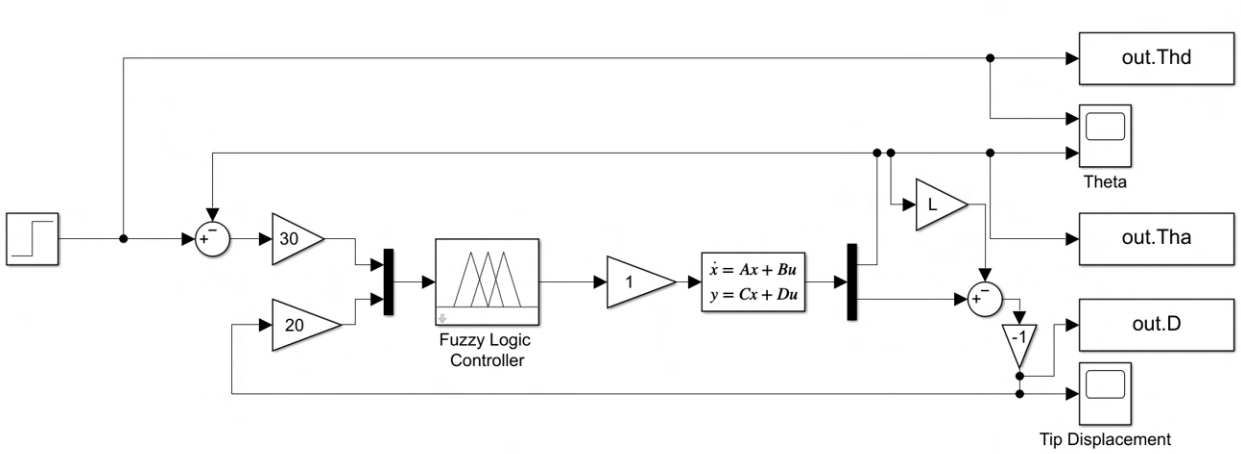

Figure 4.33: Simulink Block Diagram for Comparing Fuzzy Logic Controller Iterations

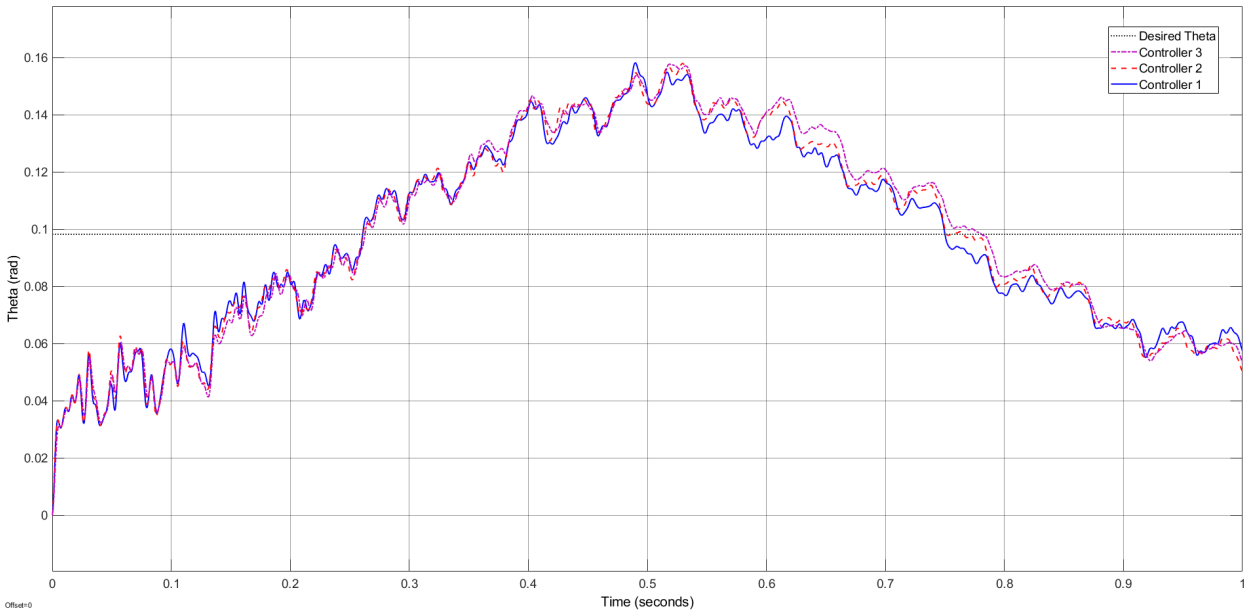

Figure 4.34: θ Response of Fuzzy Logic Controller Comparison

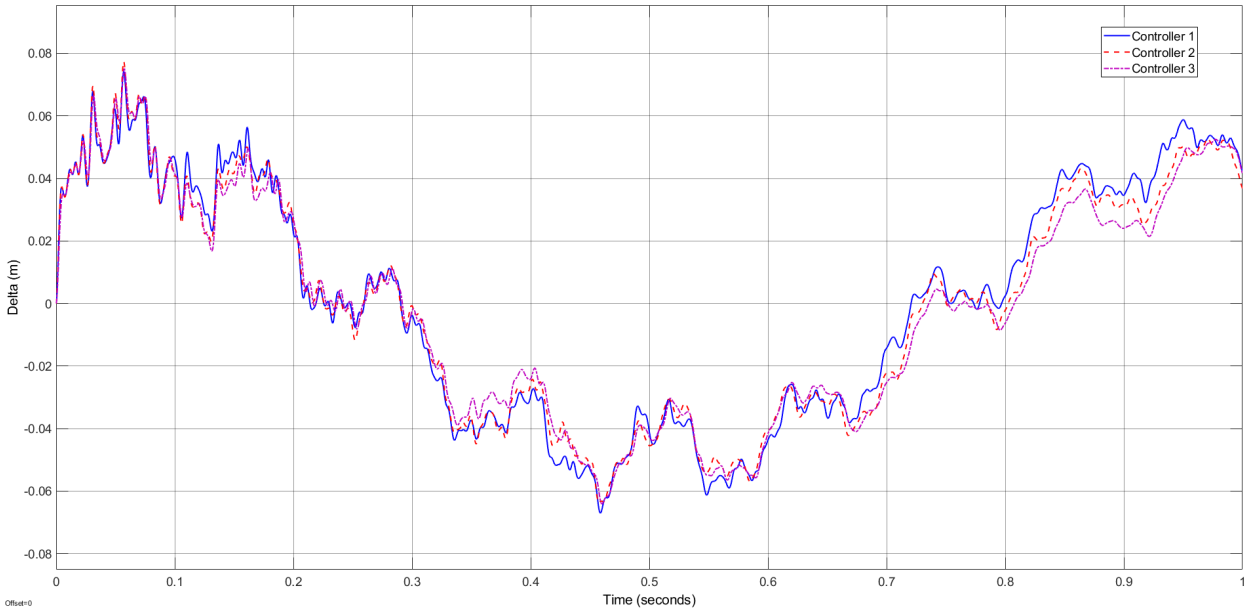

Figure 4.35: δ Response of Fuzzy Logic Controller Comparison

5 Experimental Verification

This chapter focuses on the physical testing of the controller on the link that the model was developed for. Section 5.1 is a description of the experimental set-up of the link. Section 5.2 is the calibration of the strain gauge used to measure the tip deflection of the link. Section 5.3 is the physical results of implementing the Fuzzy Logic Controller with the link. Finally, Section 5.4 implements the controller with a completely separate link to ensure that the controller is still effective even when a model of the system was not developed.

5.1 Experimental Setup

In order to analyse the link seen in Figure 5.1, a strain gauge was used to measure the tip deflection with an Omega strain gauge bridge seen in Figure 5.2. An encoder is used to measure the angular position of the base which has a resolution of 0.00004796 rads due to a quadrature encoder resolution of 2000 counts per rotation with a gear ratio of 65.5:1. A Quanser VoltPAQ-X4 power amplifier seen in Figure 5.3 is used to power the motor at the joint, and for signal conditioning a Quanser Q8 USB Data Acquisition Board (DAB) featuring

- 8 analog inputs,
- 8 analog outputs,
- 8 encoder inputs,
- 8 pulse width modulation channels, and
- 8 digital input/output channels

seen in Figure 5.4 is used to interface with a laptop through a USB port. The laptop is using MATLAB/Simulink and a Real-Time Control software called QUARC. QUARC is a

prototyping and production system designed to be integrated with Simulink to provide real-time control to a system. A diagram of the set up can be seen in Figure 5.5.

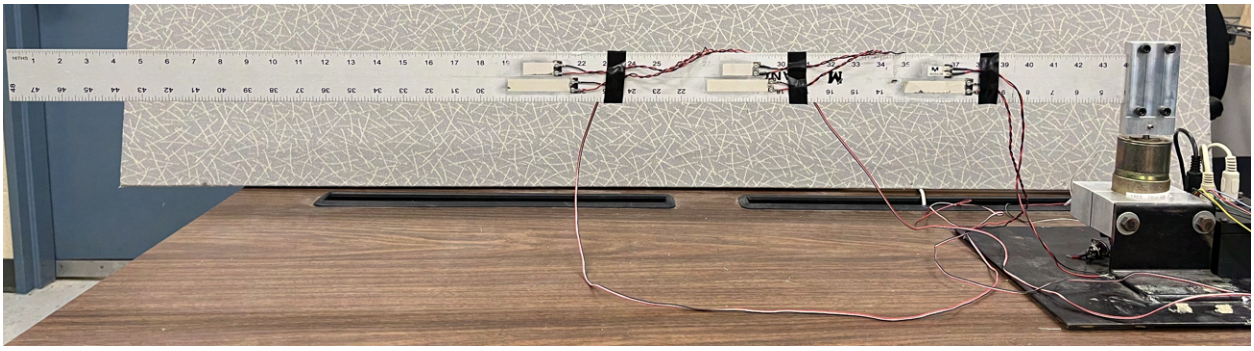

Figure 5.1: Flexible Link Experimental Set-up

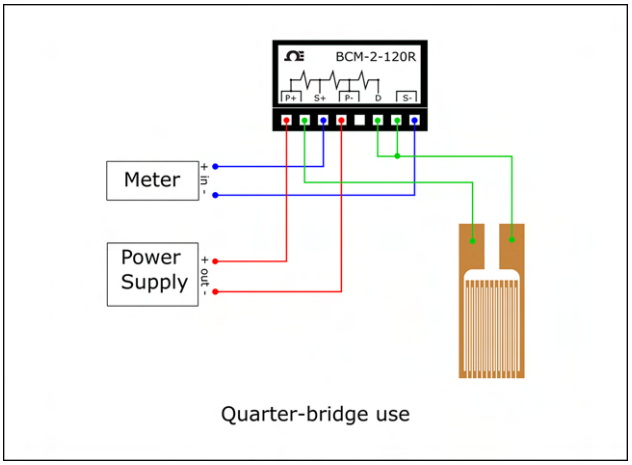

Figure 5.2: Strain Gauge Bridge Diagram [38]

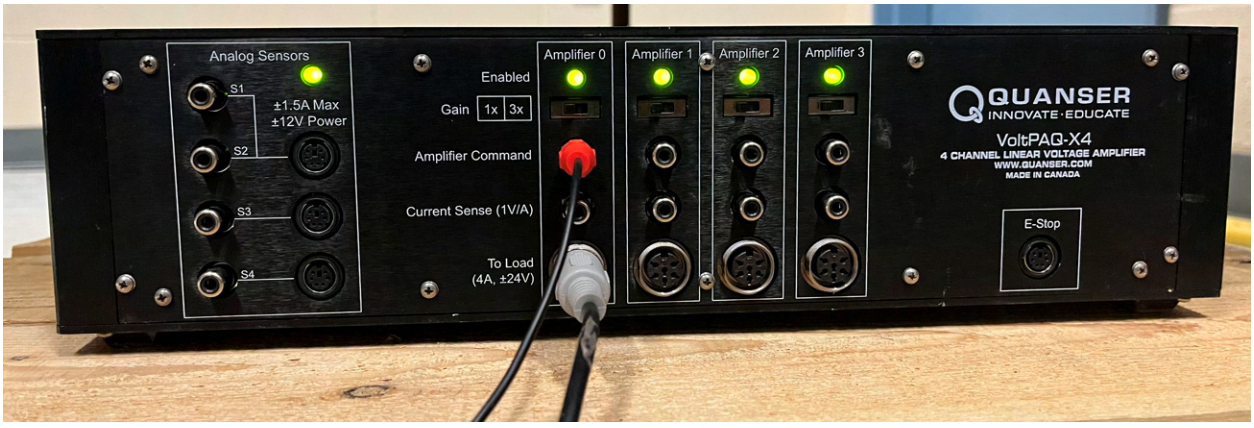

Figure 5.3: Amplifier Used to Power Motor

Figure 5.4: Data Acquisition Board with Flexible Link Set-up

The control signal comes from the laptop, moves into the DAB through the analog output signal into the amplifier under channel 0, and finally it goes into the motor which moves the link. The encoder, which reads the angular position of the motor, connects directly to the encoder input of the DAB in channel 0 which feeds the data into the laptop. Finally the strain gauge is connected to a strain gauge bridge to complete a quarter-bridge which feeds directly into the DAB in channel 1 of the analog inputs which then feeds the data into the laptop.

For the laptop set-up, MATLAB 2019b with Simulink was used with a QUARC. The Fuzzy Logic Controller had to be loaded into the workspace as well by opening the Fuzzy Inference Toolbox, importing the controller from a file, then exporting the controller to the workspace.

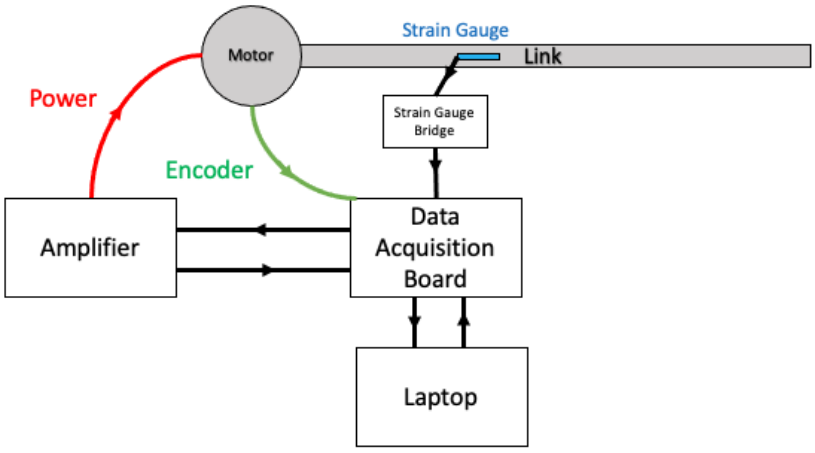

Figure 5.5: Flexible Link Experimental Set-up Diagram

5.2 Strain Gauge Calibration

Before the link could be used, the strain gauge had to be calibrated. The strain gauge was placed 0.406 m from where the motor is attached to the link as this avoids all node positions of the first four natural frequencies of the link which can be seen by plotting the mode shapes shown in Equation 5.1 [20] using the values of $\beta_n L$ from Table 2.

$$W_n(x) = sin\beta_n x + \alpha_n sinh\beta_n x \tag{5.1}$$

where $W_n(x)$ is the normalized mode shape equation for the n^{th} mode of the link, β is a combination of properties of the link given in Equation 3.9, x is the position along the length of the link, and

$$\alpha_n = \frac{sin\beta_n L}{sinh\beta_n L} \tag{5.2}$$

The first four mode shapes with normalized amplitudes (not including the rigid body motion) were plotted as seen in Figure 5.6 which shows that by placing the strain gauge 0.406 m from the motor ($x = 0$), the effects of each of these mode shapes will be able to be seen if they have a large enough amplitude. If the strain gauge had been placed where any

of these mode shapes intersect with the x-axis (called a node), that mode shape would not cause any displacement at that position and the strain gauge would not experience strain from the vibration due to that mode [19]. The strain gauge location is also near the peak deflection of the first mode shape which will have the largest amplitude of the first four mode shapes meaning that it will have the greatest impact on the response of the link and the strain gauge is in a position that is best suited to measure strain caused by the first mode.

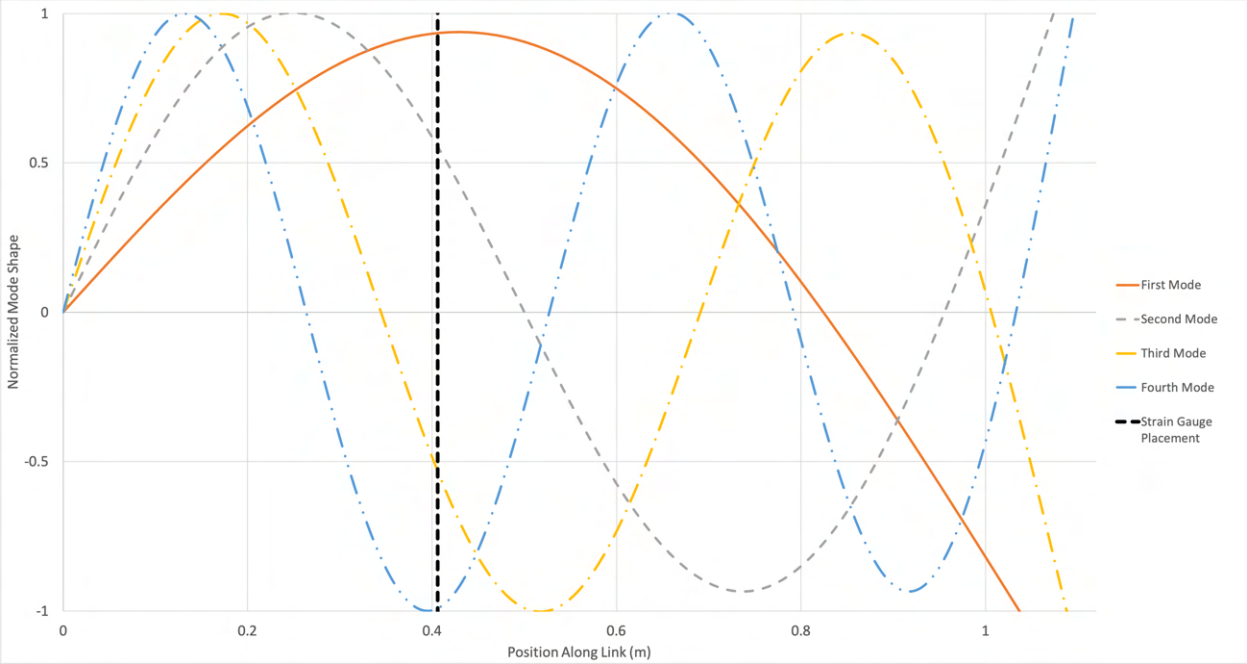

Figure 5.6: First Four Mode Shapes of Flexible Link

The Simulink diagram shown in Figure 5.7 was used in the calibration with an input signal amplitude of zero because it caused the motor to apply a torque to maintain its position as the motor was actively holding the link in its starting position. This is due to the encoder reading that as the zero position which is the desired position. This helped ensure accuracy with the calibration of the strain gauge.

The block diagram uses the same blocks from Figure 4.20 but with the addition of

- an analog input that reads the change in voltage across the strain gauge bridge (Figure

5.7 - Block 1),

- a gain block to scale the change in voltage to be calibrated to represent change in centimeters (Figure 5.7 - Block 2),
- a constant block to center the voltage change on the zero position (Figure 5.7 - Block 3), and
- a block that stores the tip displacement data to the workspace (Figure 5.7 - Block 4).

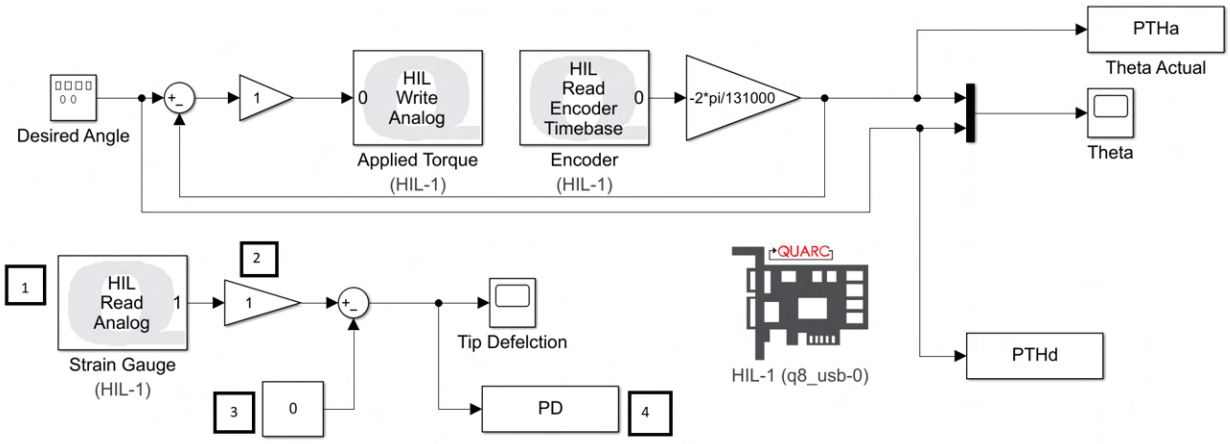

Figure 5.7: Simulink Block Diagram Used for Strain Gauge Calibration

A piece of paper was attached directly underneath the tip of the link with a straight line with marks every centimeter. The line is straight as the deflection of the link is measured by straight line distance away from the rigid body motion and does not follow an arc. The tip of the link was then moved two centimeters at a time to 12 centimeters on either side of the rigid body position as seen in Figure 5.8. The voltage that the strain gauge produced at each position was then recorded. This data was then graphed and used to form a linear trend line mapping the voltage measured against the tip displacement as seen in Figure 5.9. The equation of this trend line then gives the gain and sum values required in the Simulink model to ensure that the strain gauge was accurately calibrated.

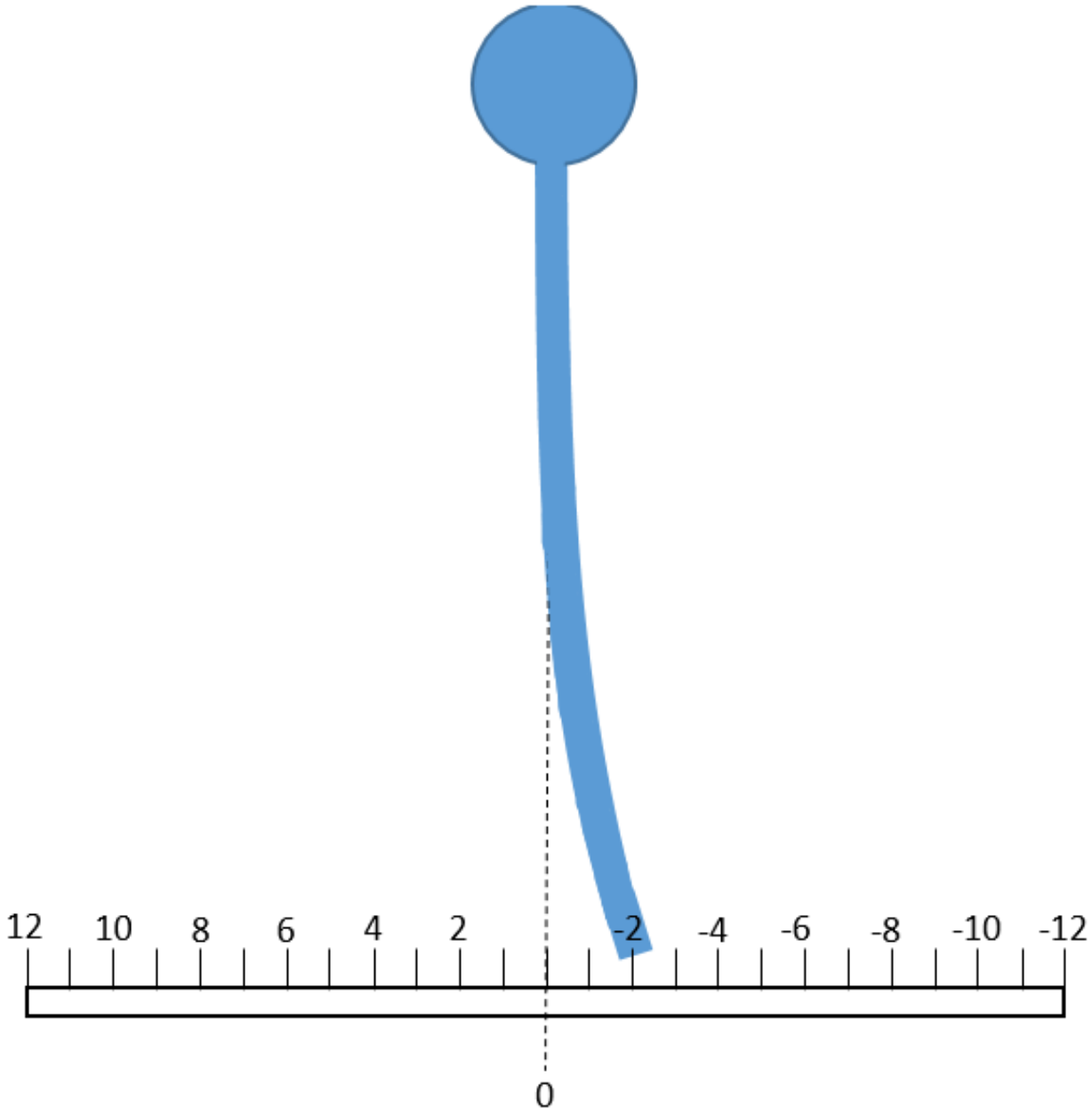

Figure 5.8: Strain Gauge Calibration Method

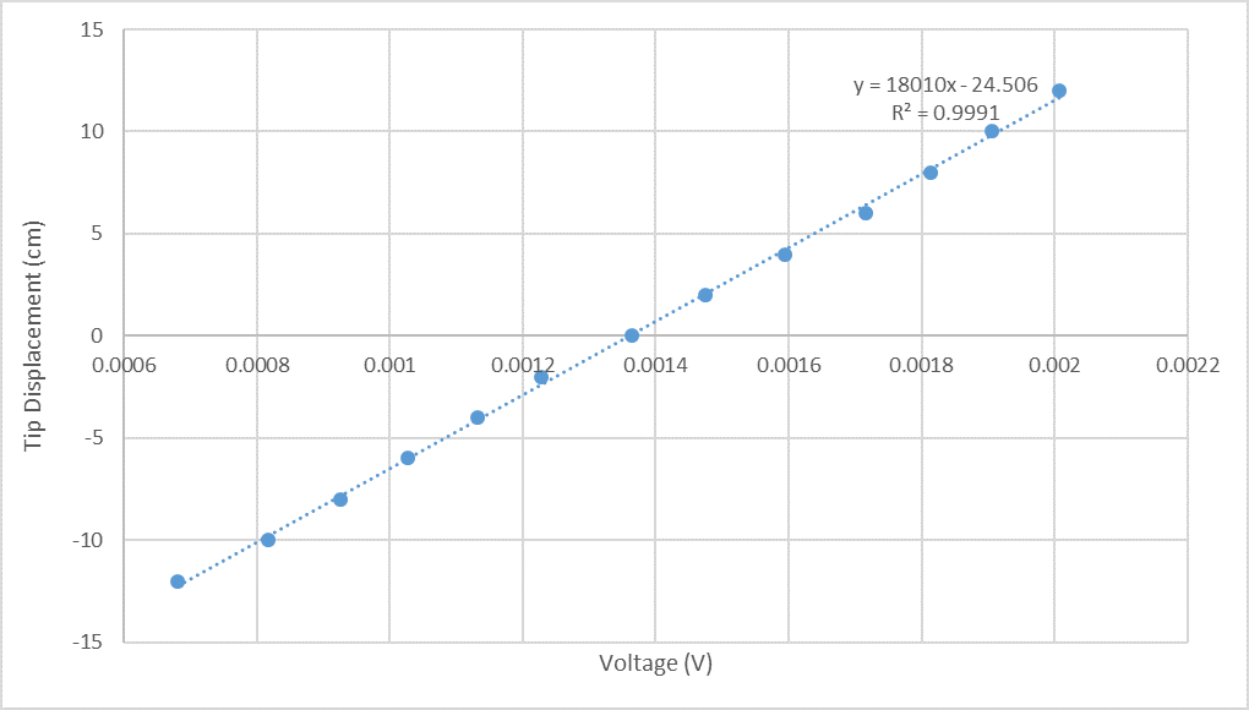

Figure 5.9: Strain Gauge Calibration Graph

5.2.1 Filtering Strain Gauge Signal

Throughout the strain gauge calibration process, noise was present in the response. In order to minimize this noise to obtain a more accurate tip deflection response, the signal can be filtered. Filtering a signal involves altering the amplitude of the frequency components of a signal [39]. While there are many types of filters, some of the most common types of filters are Low-Pass Filters, High-Pass Filters, Band-Pass Filters, and Moving Average Filters. A Low-Pass Filter attenuates frequencies higher than the cutoff frequency, a High-Pass Filter attenuates frequencies lower than the cutoff frequency, a Band-Pass Filter attenuates frequencies both higher and lower than the range of frequencies identified. The Moving Average Filter averages the output for any time based on the measured values within a designated range of that time. This is meant to have a larger effect on the high frequency components of the signal while smoothing the overall response of the system [39].

The High-Pass Filter should not be used in this context as the vibration response of a flexible link has larger amplitudes at lower modes of the link and therefore it is important not to attenuate lower frequencies. This is also why the Band-Pass Filter is not useful as the low frequencies are important and if the lower frequencies should not be attenuated, it leaves either the Low-Pass Filter or the Moving Average Filter as the best choice for conditioning the signal. The Moving Average Filter is particularly useful for reducing random noise in the time domain whereas the Low-Pass Filter is better suited for targeting high frequency noise.

To determine which of these two filters should be used, the FFT response of the noise was obtained with no vibration in the link as seen in Figure 5.10 to determine the frequency response of the noise. As seen from Figure 5.10, there is a consistent, low amplitude of noise across all frequencies. Therefore, a Low-Pass Filter will not sufficiently reduce the noise in the strain gauge without attenuating the frequencies of the link's vibration that are being observed. The Moving Average Filter however, will smooth all of these frequencies.

In order to determine the number of frames that are being used in the Simple Moving Average Filter, a desired angular position input of $\pi/8$ with a frequency of 0.05 Hz was given to the link with a varying number of frames being used in the moving average with each frame being 0.01 seconds. This testing was conducted with no filtering, 4 frames used, 8 frames used, 10 frames used, 15 frames used, and 20 frames used. The results of the strain gauge readings can be seen in Figures 5.11 to 5.15. The percent difference was then calculated for each response as seen in Table 10. An 8 frame Moving Average Filter was chosen to reduce the noise in the strain gauge as it smooths the response while having less than a 10% difference between the peak values of the strain gauge as seen in Figure 5.12. Beyond

8 frames of averaging, the signal begins to reduce the peaks during oscillation causing up to a 40.73% difference at 10 frames, 49.03% difference at 15 frames, and 44.88% difference at 20 frames.

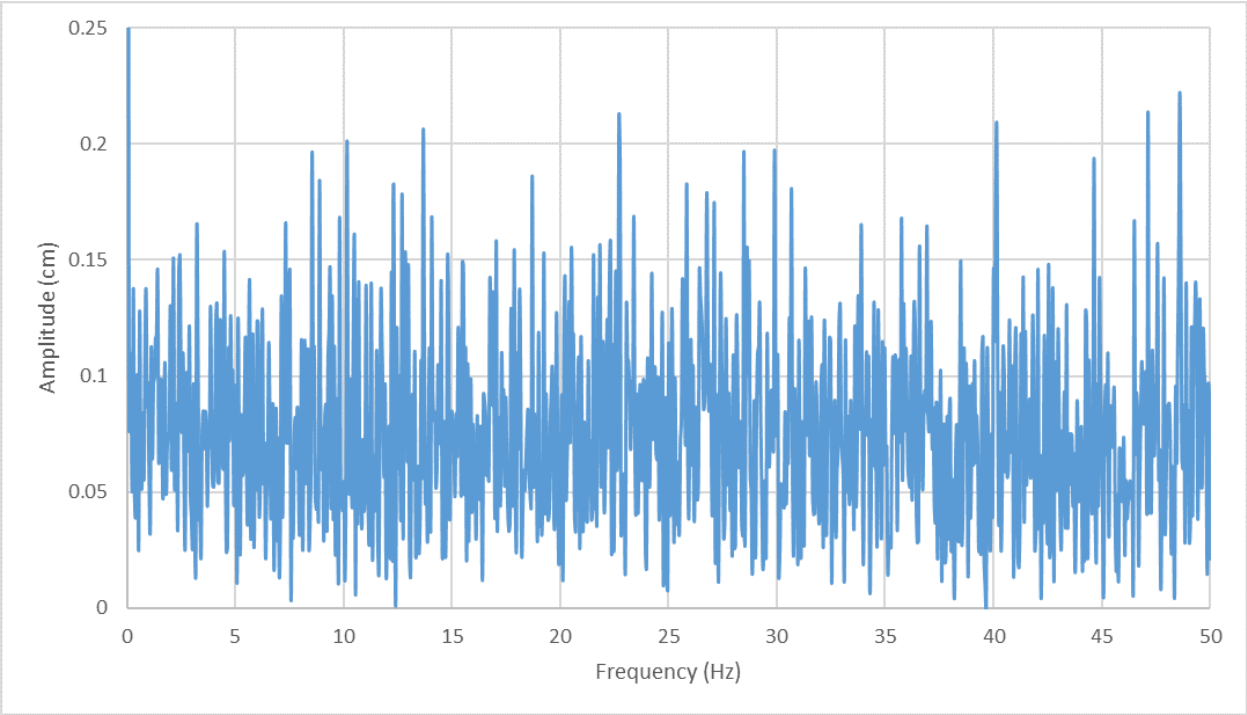

Figure 5.10: Strain Gauge Noise FFT Response

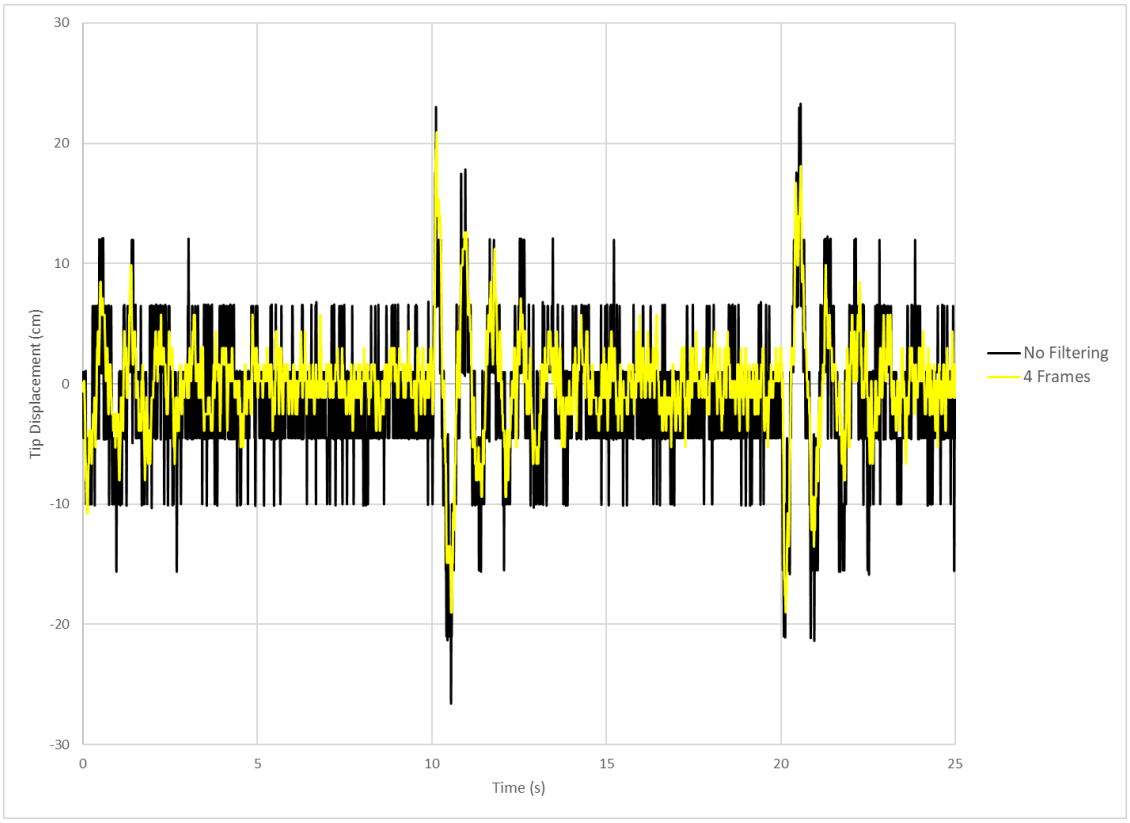

Figure 5.11: Strain Gauge Response With 4 Frames Used for Moving Average

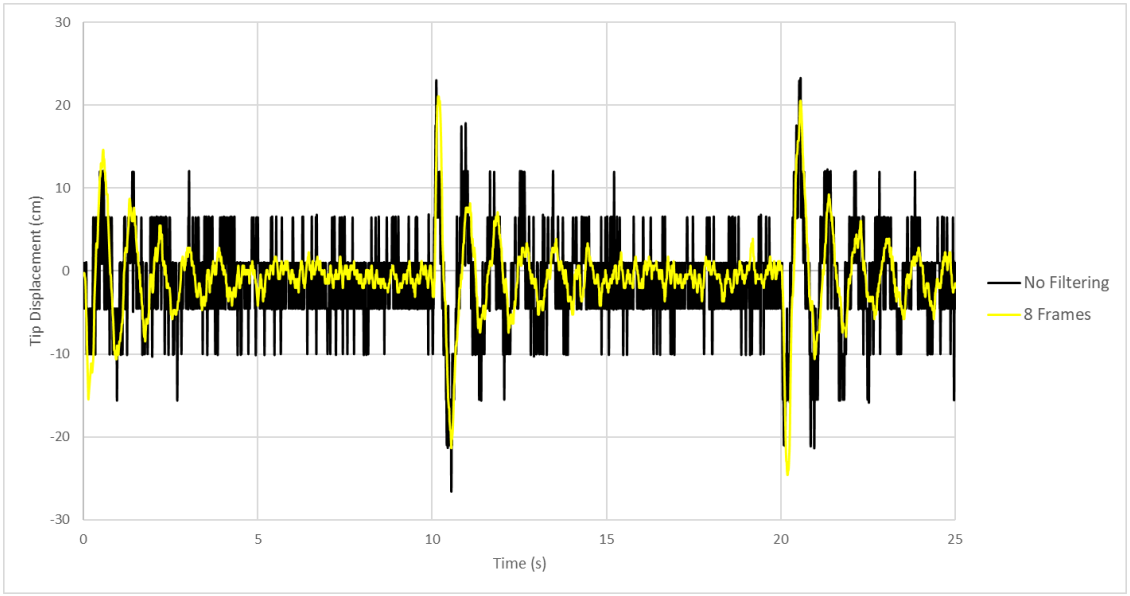

Figure 5.12: Strain Gauge Response With 8 Frames Used for Moving Average

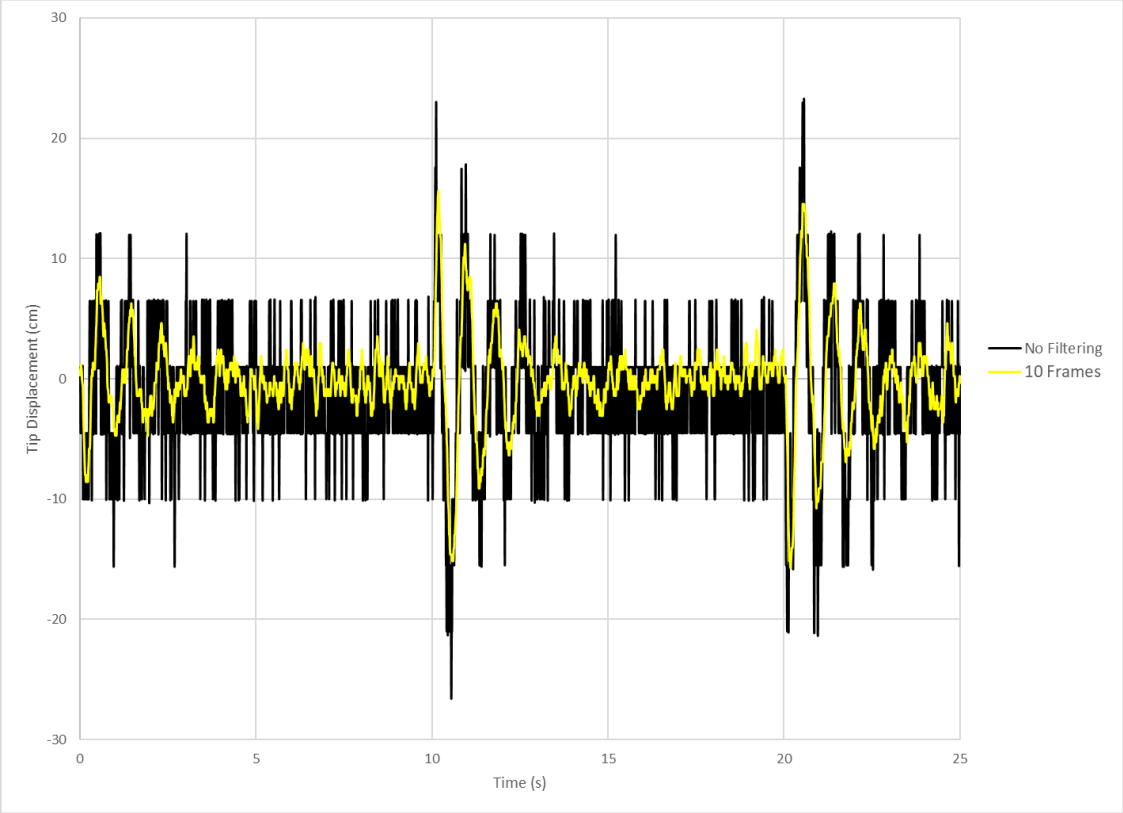

Figure 5.13: Strain Gauge Response With 10 Frames Used for Moving Average

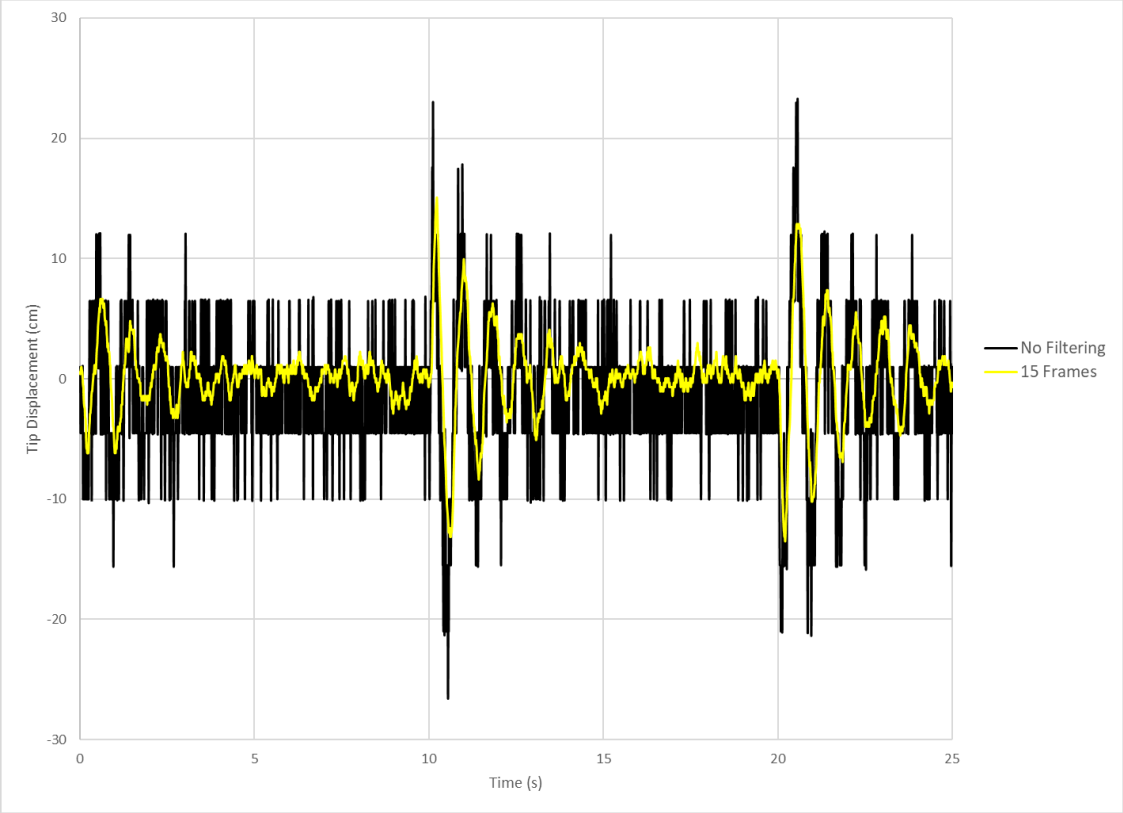

Figure 5.14: Strain Gauge Response With 15 Frames Used for Moving Average

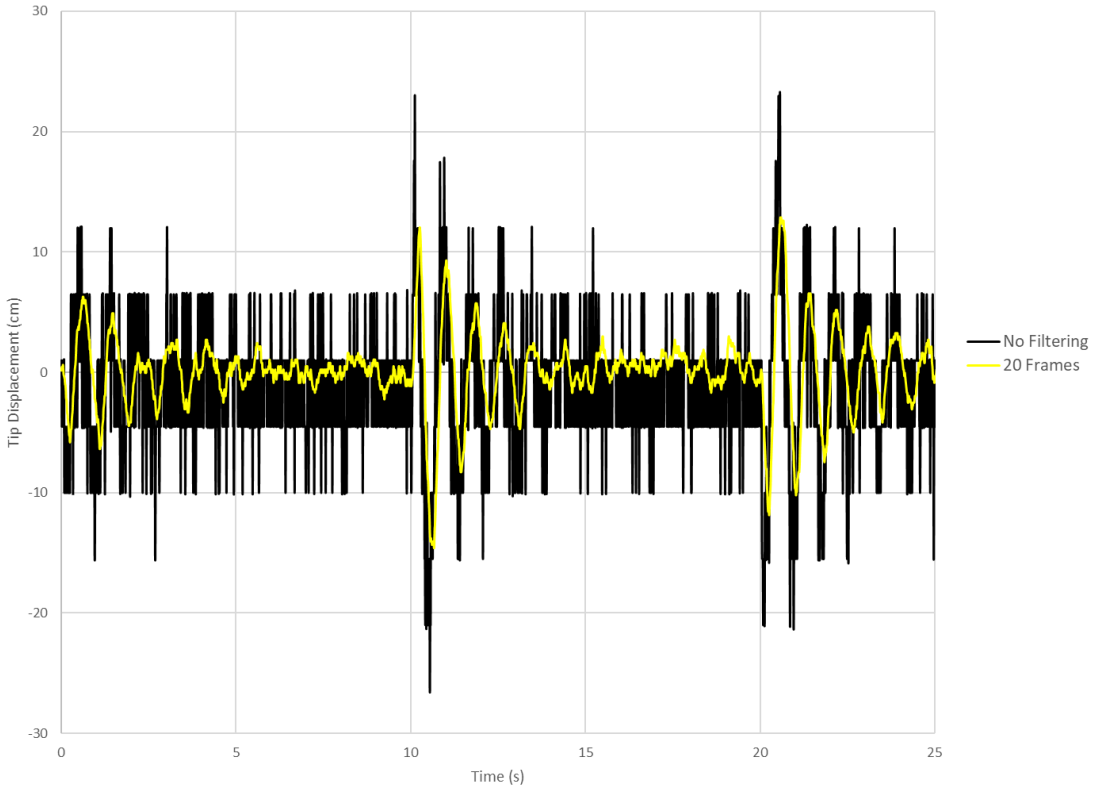

Figure 5.15: Strain Gauge Response With 20 Frames Used for Moving Average

Table 10: Simulink Gain Values

	4 Frames	**8 Frames**	**10 Frames**	**15 Frames**	**20 Frames**
Max Peak Percent Difference(%)	9.24	8.28	31.98	34.37	43.94
Min Peak Percent Difference(%)	28.29	7.27	40.73	49.03	44.88

5.3 Experimental Controller Results

With the calibration completed, the link was then tested with a proportional controller and then again with the Fuzzy Logic Controller designed from this research.

The link was given a square pulse input with amplitudes of $\pi/16$, $\pi/8$, and $\pi/2$ rads and a frequency of 0.05 Hz. The link's vibration response was first measured using a proportional controller as seen in Figure 5.16. A torque response of the controller was also measured for

the link with a $\pi/16$ angular position input as it was noted that the motor produced a lot of noise under proportional control. The proportional gain value used was the value determined in Chapter 4.

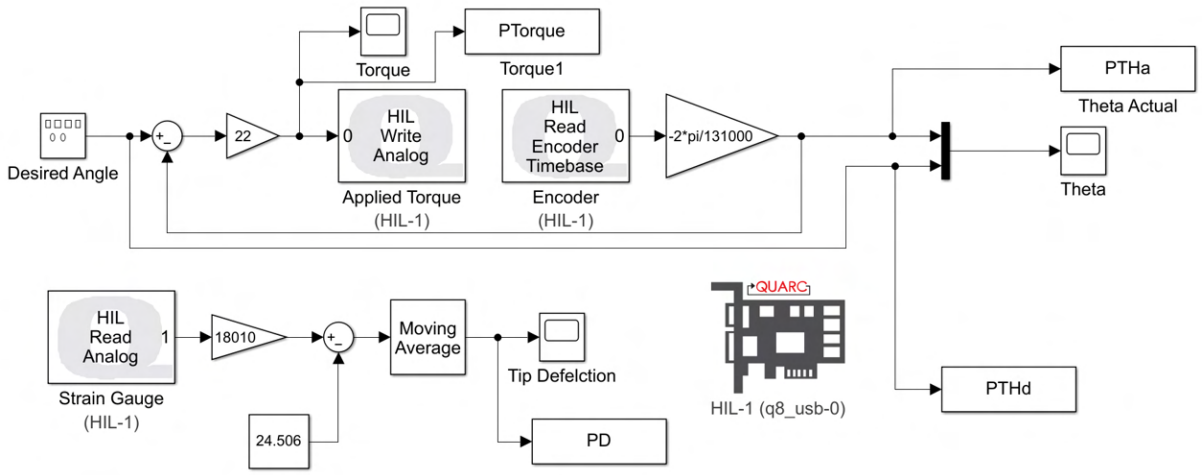

Figure 5.16: Simulink Block Diagram with Proportional Controller Implemented

Once the link's response with a proportional controller was obtained, the link was subjected to the same loading conditions but with the Fuzzy Logic Controller as seen in Figure 5.17. The gains for the Fuzzy Logic Controller were 75 for the angular position error gain, 0.1 for the tip displacement gain, and 1 for the torque gain except with the $\pi/2$ input where the torque gain was increased to 2. These were chosen by starting with the scaling factors used in simulations and adjusting heuristically. The angular position error scaling factor was increased to 75 as this was the value where the steady-state error was completely eliminated with the Fuzzy Logic Controller. The tip deflection scaling factor was reduced by a factor of 100 for its starting point as well because the tip deflection was measured in meters in simulation but in centimeters with the physical test rig. The corresponding θ and δ responses can be seen in Figures 5.18 through 5.24, overlaid with the proportional results to show the reduction in tip deflection compared with an uncontrolled response.

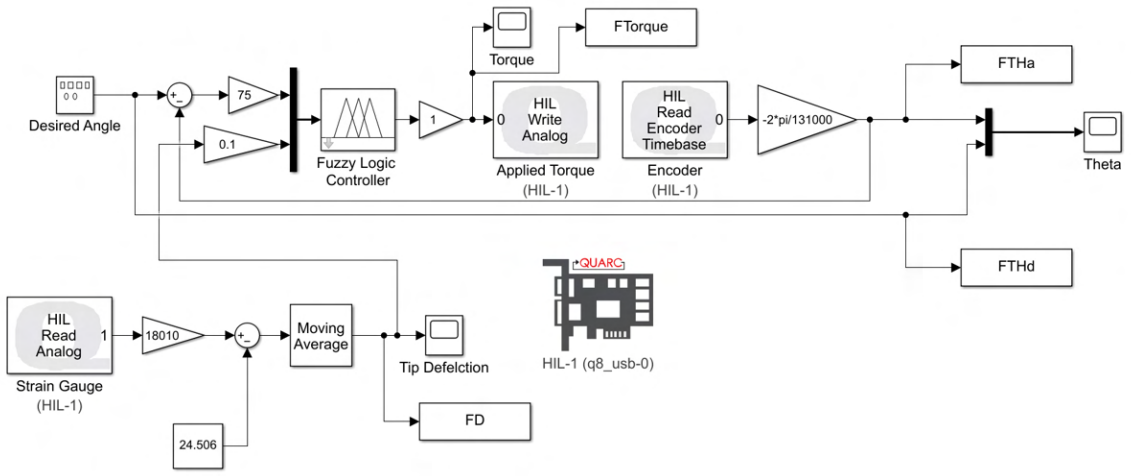

Figure 5.17: Simulink Block Diagram with Fuzzy Logic Controller Implemented

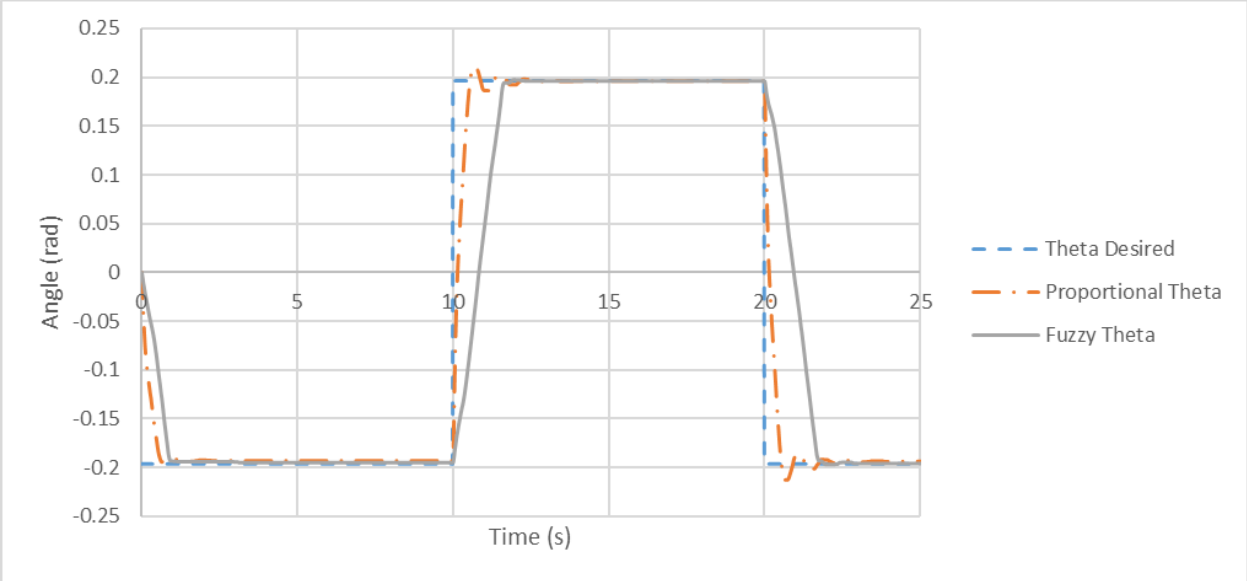

Figure 5.18: Flexible Link θ Response with a $\pi/16$ Input

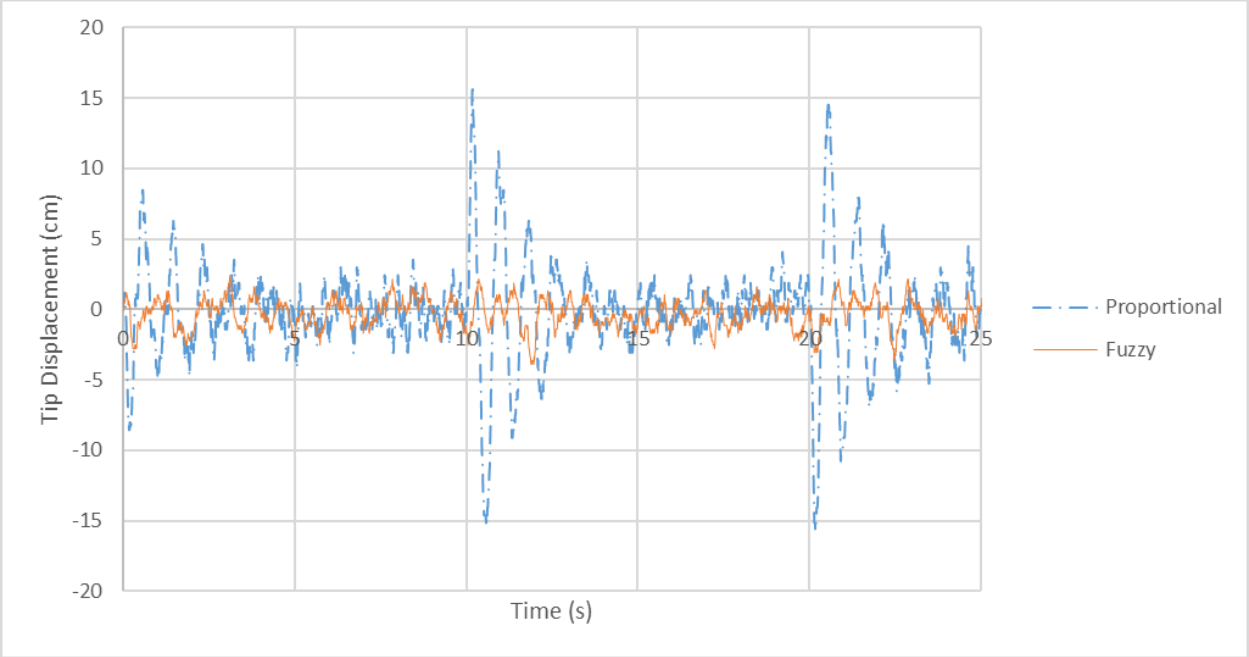

Figure 5.19: Flexible Link δ Response with a $\pi/16$ Input

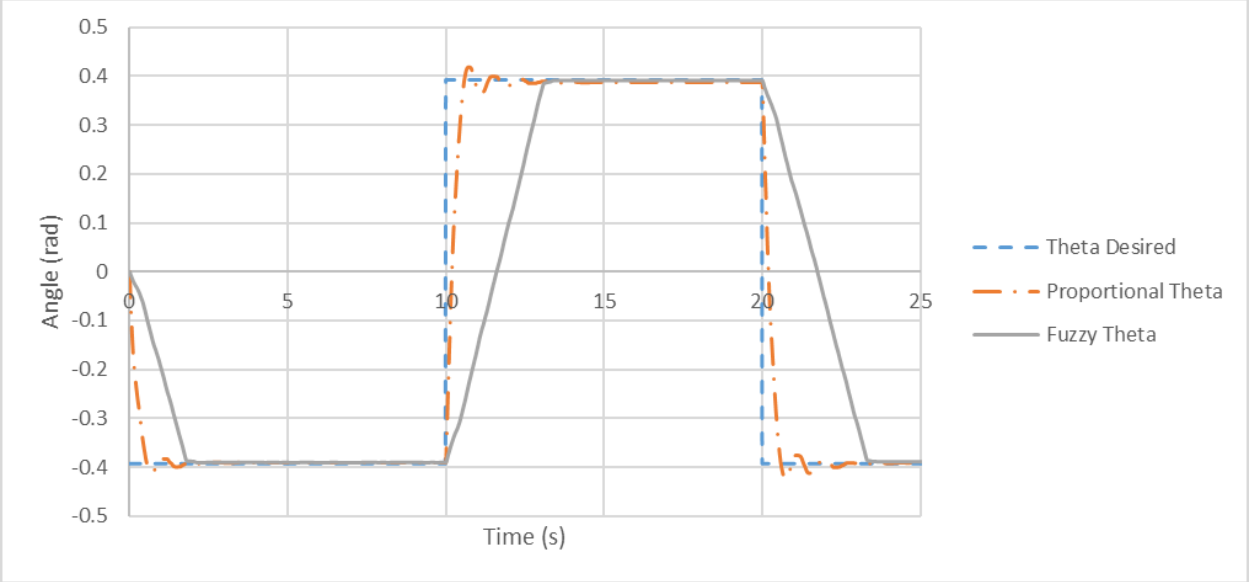

Figure 5.20: Flexible Link θ Response with a $\pi/8$ Input

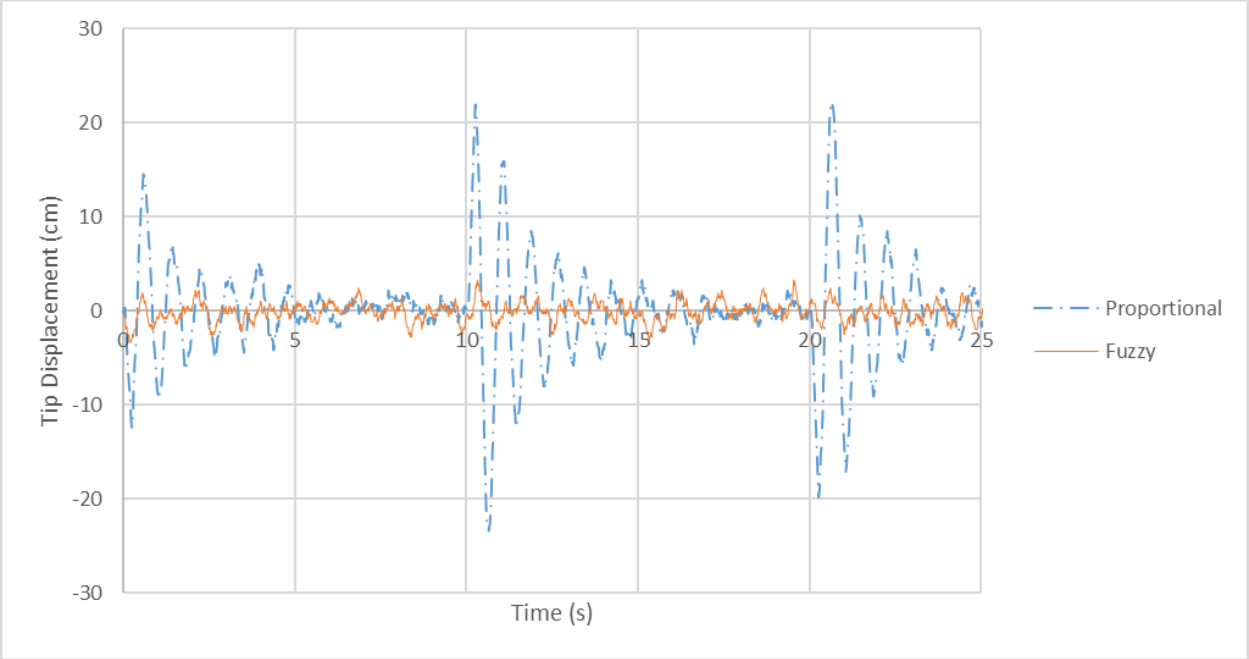

Figure 5.21: Flexible Link δ Response with a $\pi/8$ Input

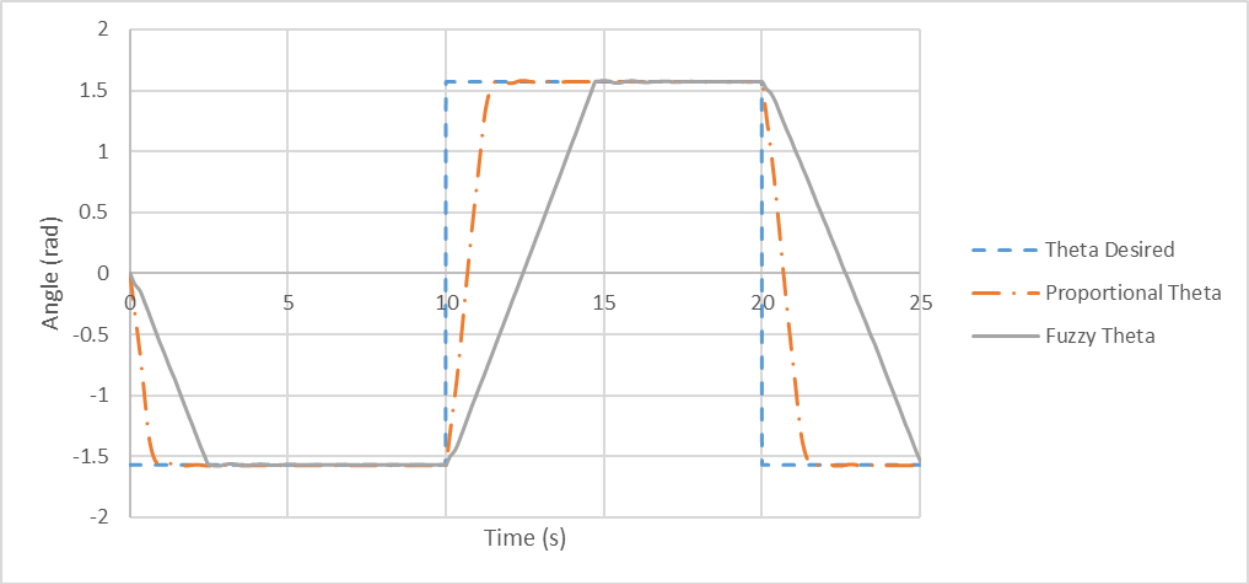

Figure 5.22: Flexible Link θ Response with a $\pi/2$ Input

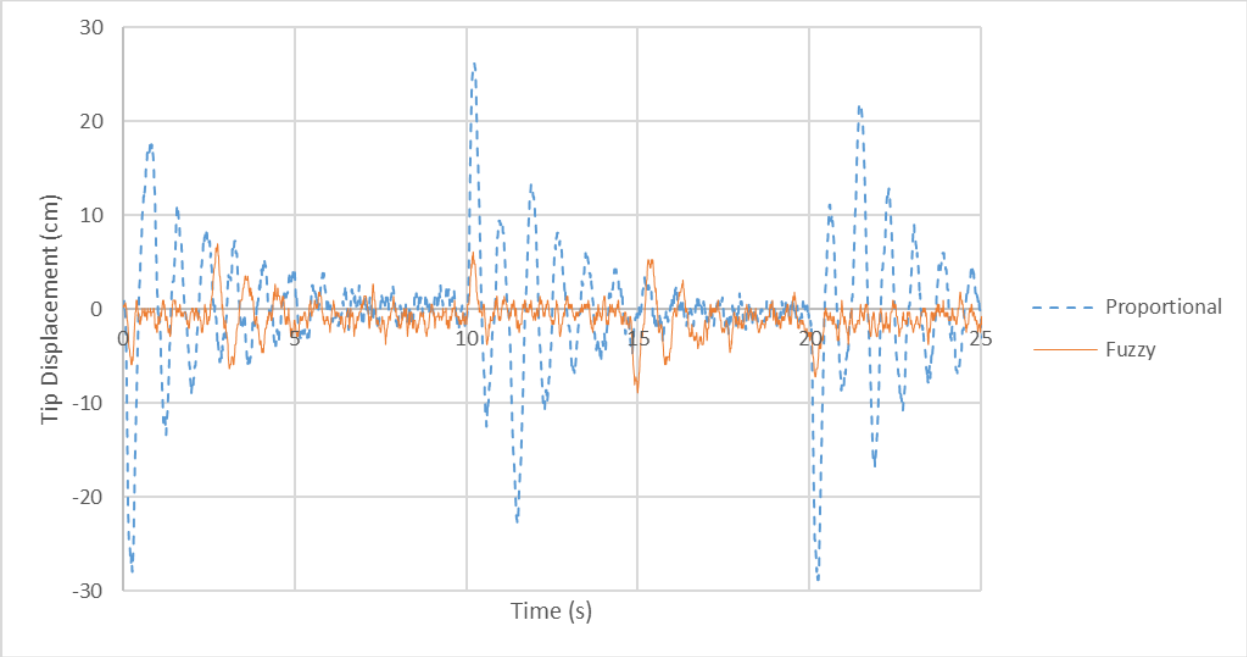

Figure 5.23: Flexible Link δ Response with a $\pi/2$ Input

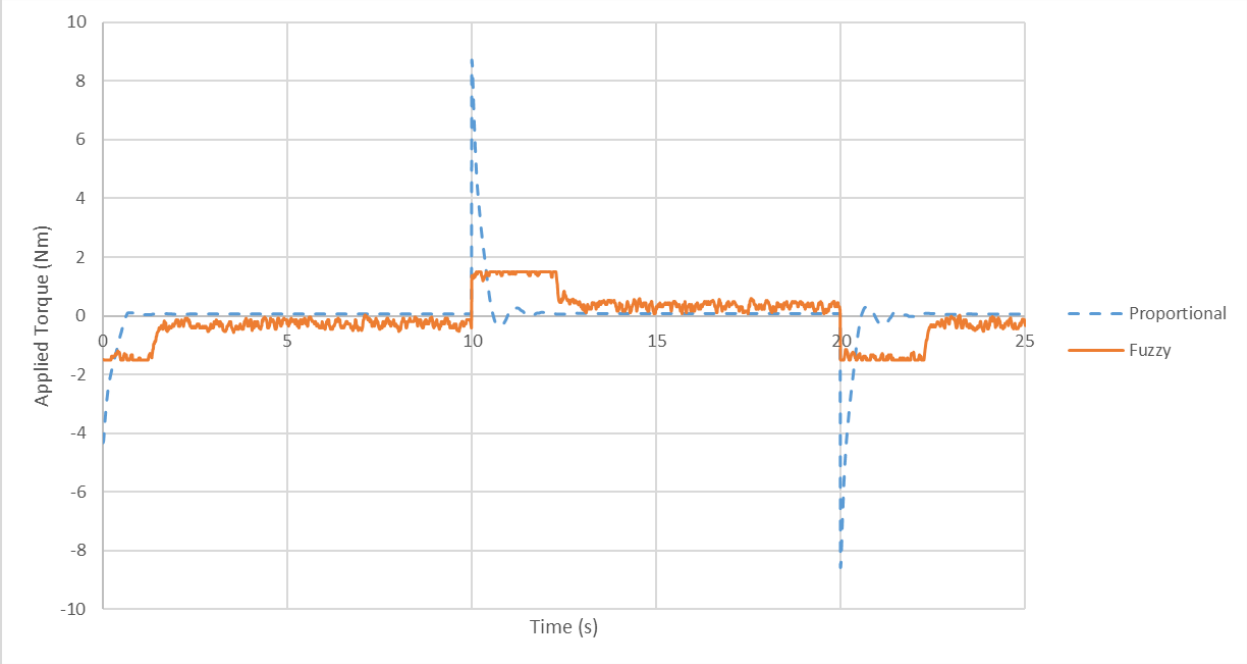

Figure 5.24: Flexible Link Torque Response with a $\pi/16$ Input

The Fuzzy Logic Controller was able to reduce the tip deflection and eliminate the angular position error of the link. The results of the testing can be seen in Table 11. The

proportional controller moved to the steady-state position faster than the Fuzzy Logic Controller. However, with the proportional controller, the link overshoots the position so the trade off for the Fuzzy Logic Controller is that it moves into position more slowly but with more accuracy and less vibration.

The link's angular position under proportional control also had a steady-state error. The Fuzzy Logic Controller had a very small amount of motion caused by the noise in the strain gauge where after being in position it continued to try to adjust the angle to try to reduce the "vibration" that it was reading but was not actually there. The angular position error in the proportional controller was as high as 0.00408 rad.

The Fuzzy Logic Controller also reduced the tip deflection. The link under proportional control had a maximum tip deflection of 28.80 cm with a $\pi/2$ input. The link using the Fuzzy Logic Controller however reduced the maximum tip deflection to 8.90 cm with the same input. The Fuzzy Logic Controller saw up to a 85.73% reduction in end point deflection.

The Fuzzy Logic Controller also used 82.64% less torque than the proportional controller with a $\pi/16$ angular position input (8.64 Nm VS 1.5 Nm). This is also the input where the proportional controller uses the least amount of torque of the tests conducted. There was a loud noise given off by the motor when used with proportional control which was not present with the Fuzzy Logic Controller. This rapid change to high torque values puts additional stress on the motor and will reduce its lifespan.

Table 11: Flexible Link Experimental Results

	π/16 Input	π/8 Input	π/2 Input
Proportional Controller θ Error (rad)	0.0034	0.0020	0.0041
Proportional Controller Percent Overshoot (%)	7.18	6.61	0.58
Fuzzy Logic Controller Percent Overshoot (%)	0.49	0	0.60
Proportional Controller Max Tip Displacement (cm)	15.71	23.40	28.80
Fuzzy Logic Controller Max Tip Displacement (cm)	3.89	3.34	8.90
δ Percent Improvement (%)	75.24	85.73	69.10

5.4 Flexcam Link Vibration Control

One of the key benefits of using a Fuzzy Logic Controller is that it is a non-model based controller. Therefore, this controller should be able to be applied to any other rotating, flexible link with minimal tuning required. In order to verify that the controller was effective with multiple systems and not just the one that it was designed for specifically, a 0.420 m long link with a Quanser Flexcam sensor, seen in Figure 5.25 and 5.26, was used to ensure the Fuzzy Logic Controller remained effective.

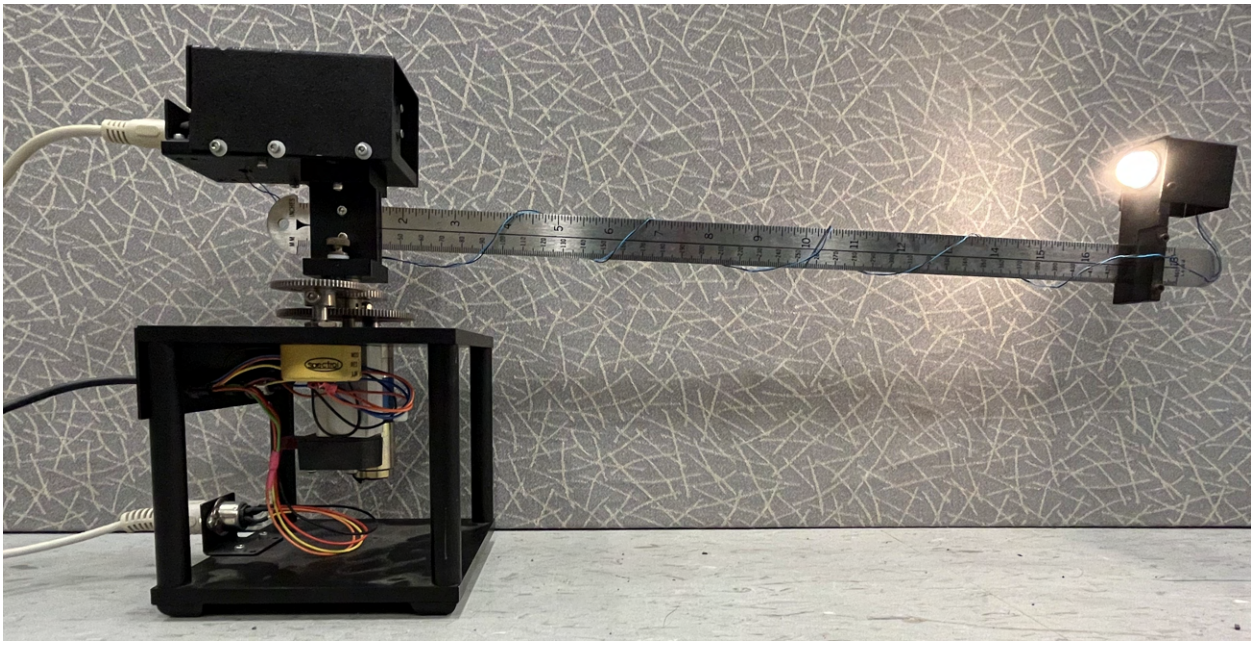

Figure 5.25: Flexible Link with Flexcam Set-up

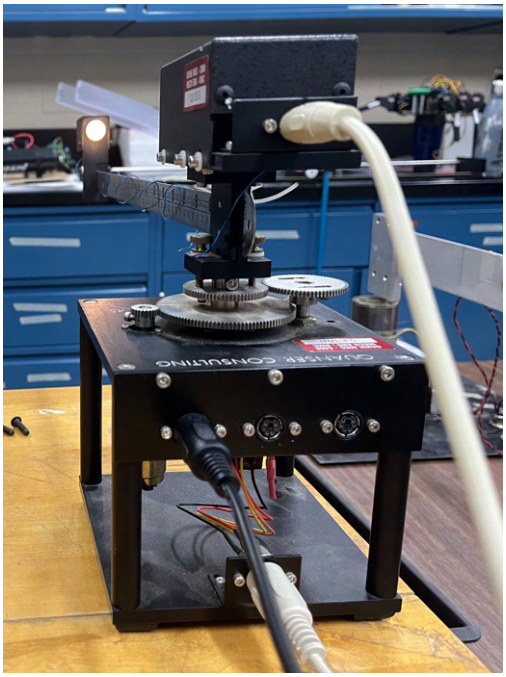

Figure 5.26: Flexible Link with Flexcam Wire Connections

5.4.1 Flexcam Experimental Setup

The second link to be analyzed uses a Quanser Flexcam device to measure the tip deflection. The Flexcam works by attaching a light to the end of the link with a camera at the motor hub. The camera is able to measure the deviation of the light from the rigid body position. The set-up is similar to the first link which was describes in Section 5.1 however, the Flexcam link uses an encoder which has a quadrature resolution of 4096 counts per rotation to measure the angular position of the base, and the Flexcam sensor connects to the Quanser VoltPAQ-X4 power amplifier using the set-up from Figure 5.28 which is then connected to the Quanser Q8 USB DAB in the same position seen in Figure 5.4. A diagram of the set up can be seen in Figure 5.27.

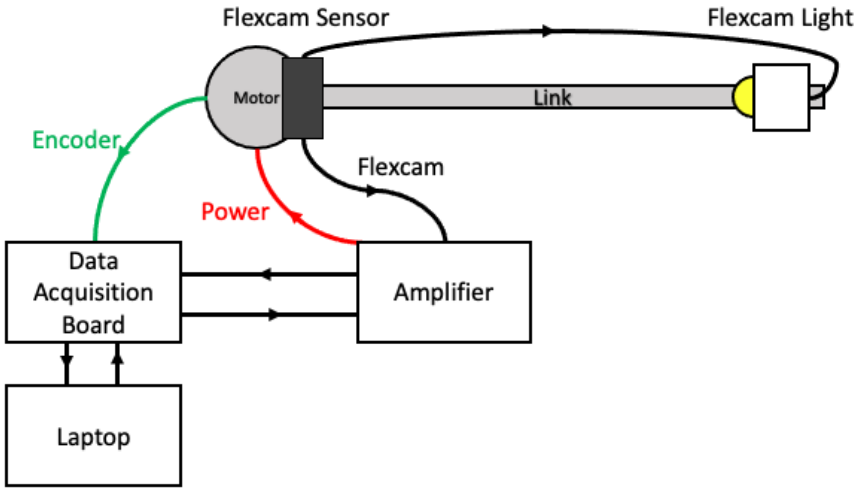

Figure 5.27: Flexcam Flexible Link Set-up Diagram

Figure 5.28: Amplifier Used to Power Motor and Flexcam Sensor

5.4.2 Flexcam Calibration

Before the Flexcam could be used, it had to be calibrated based on its location along the link. The leading edge of the Flexcam light was placed 0.395 m from where the Flexcam camera is attached at the joint position. The method for calibration is similar to that described in Section 5.2 for calibrating the strain gauge except the tip of the link was then moved one centimeter at a time to eight centimeters on either side of the rigid body position. This data was graphed and used to form a linear trend line mapping the voltage measured against the tip displacement as seen in Figure 5.29. The equation of this trend

line then gives the gain and sum values required in the Simulink model to ensure that the Flexcam was accurately calibrated.

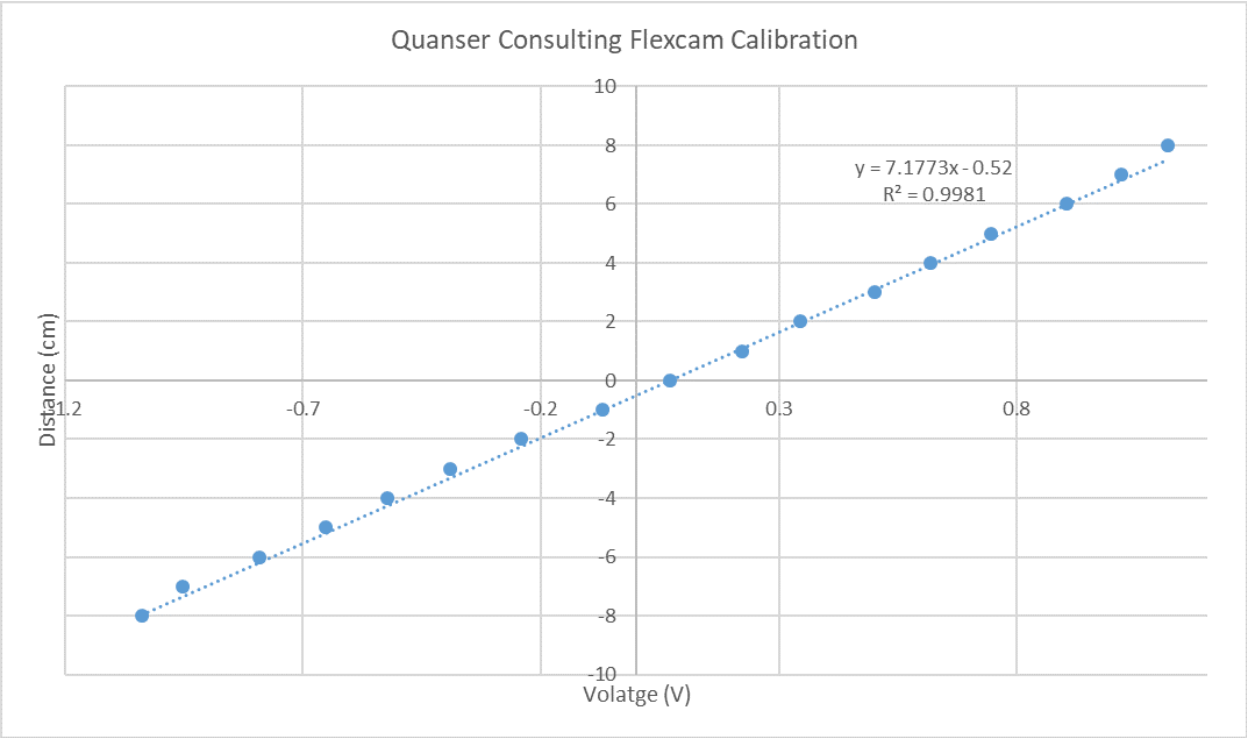

Figure 5.29: Flexcam Calibration Graph With the Light Source 0.395 m From The Hub

5.4.3 Flexcam Results

The link was then tested with a proportional controller and then again with the Fuzzy Logic Controller designed from this research. Similarly to the previous link, the proportional gain value was determined heuristically with the block diagram seen in Figure 5.30 while adjusting the torque gain value. The proportional gain value was chosen to have a value of 5 so that the link would reach its steady-state position within 2.5 seconds.

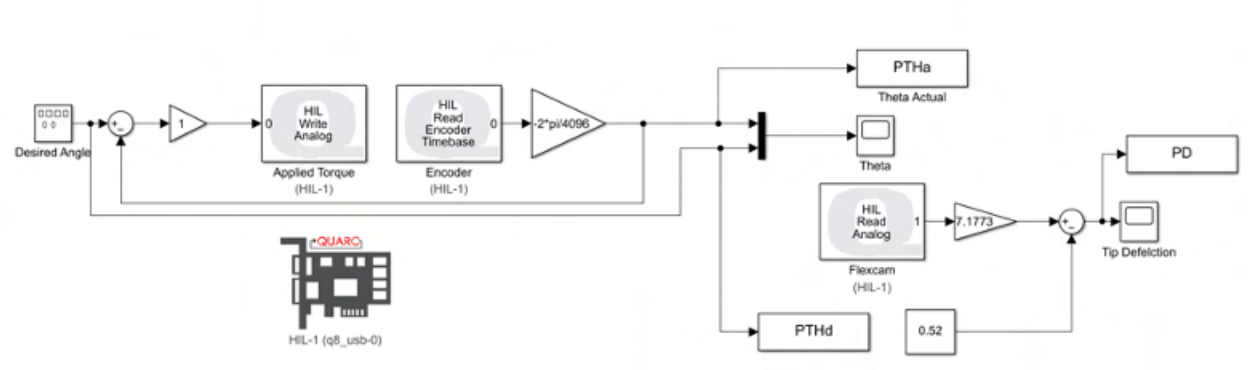

Figure 5.30: Simulink Block Diagram for Proportional Gain Determination for the Flexcam Link

The Flexcam link was given a square pulse input with amplitudes of $\pi/16$, $\pi/8$, and $\pi/4$ rads with a frequency of 0.02 Hertz. The Flexcam link first ran using a proportional controller as seen in Figure 5.31.

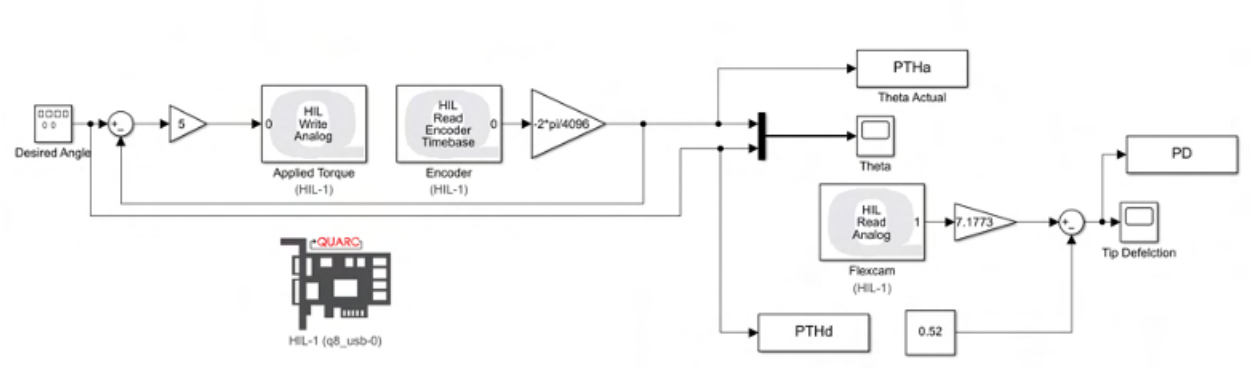

Figure 5.31: Flexcam Link Simulink Block Diagram with Proportional Controller

The Flexcam link then ran under the same loading conditions but with the Fuzzy Logic Controller as seen in Figure 5.32 with an angular position error gain value of 75, a tip deflection gain value of 0.1 and a torque gain value of 0.25. These scaling factors were chosen as the same scaling factors from the aluminum link with the exception of the torque

scaling factor which started at 0.5 due to the smaller link requiring less torque and then being adjusted such that the maximum tip deflection was less than 2 cm. The corresponding θ and δ responses can be seen in Figures 5.33 through 5.38.

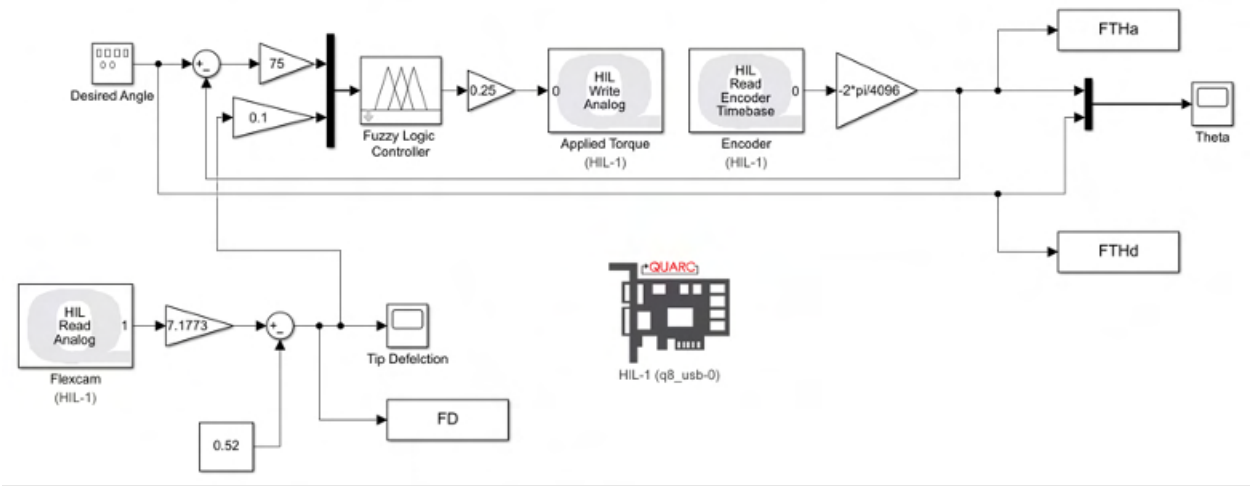

Figure 5.32: Flexcam Simulink Setup with Fuzzy Logic Controller

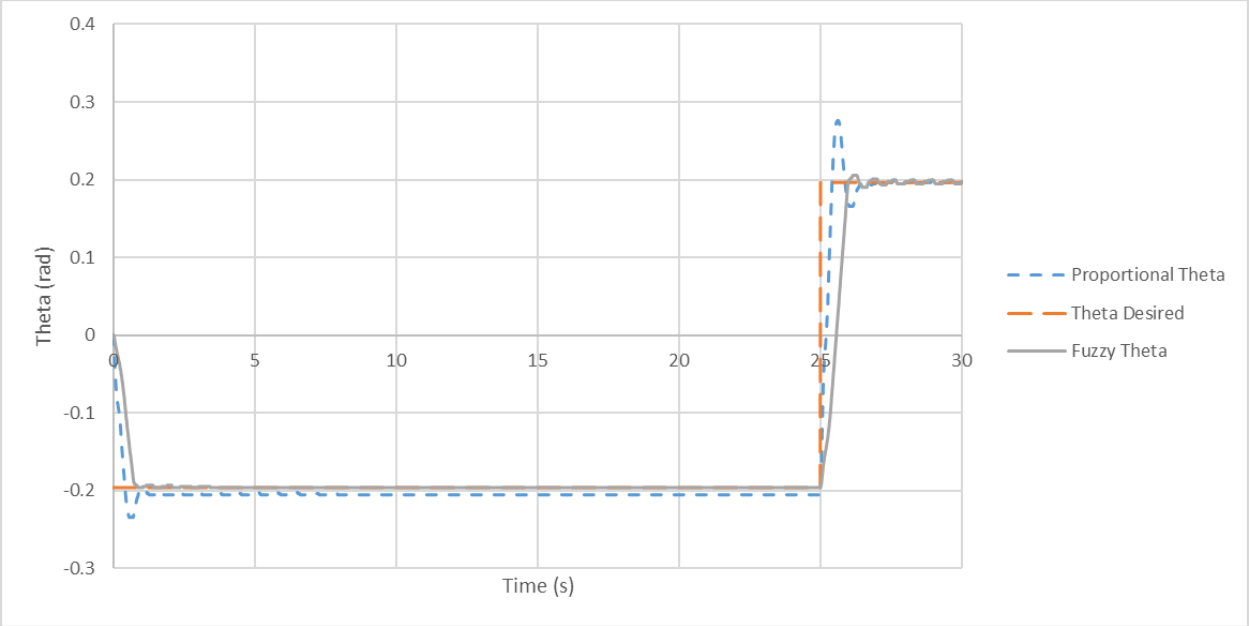

Figure 5.33: Flexcam Link θ Response with a $\pi/16$ Input

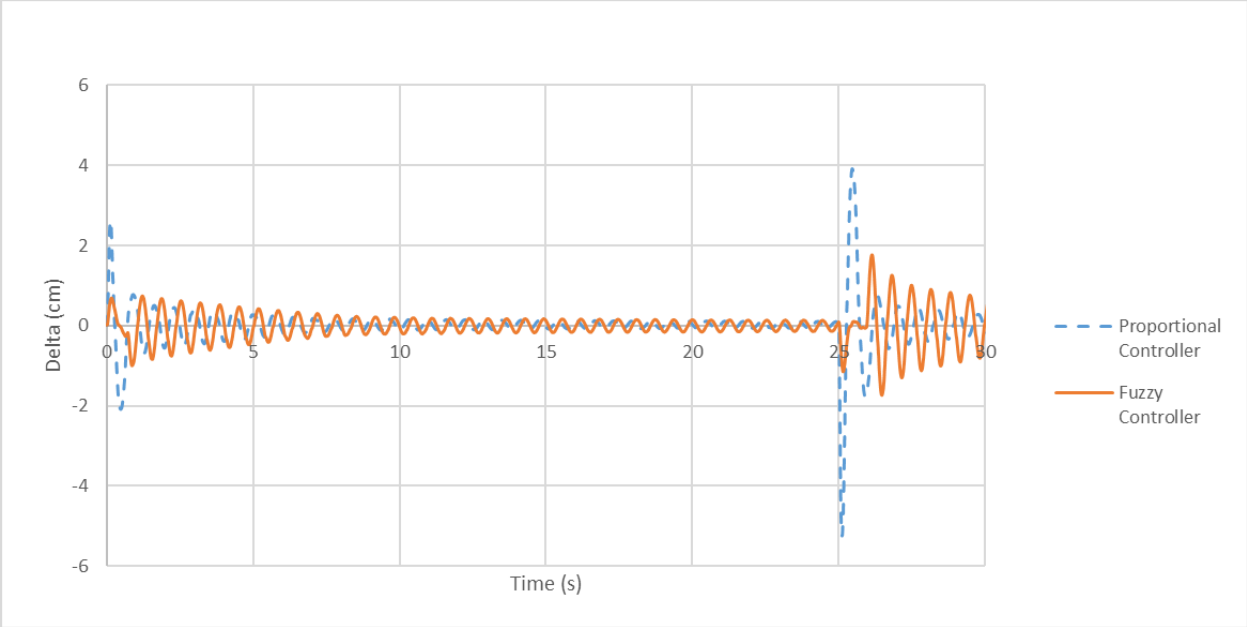

Figure 5.34: Flexcam Link δ Response with a $\pi/16$ Input

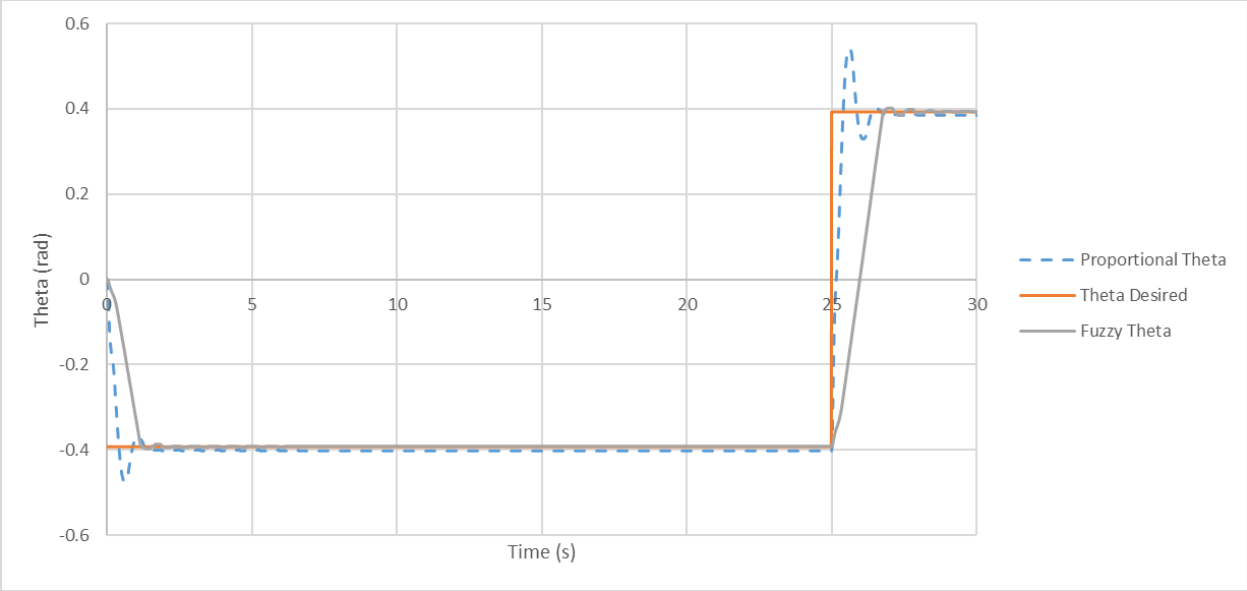

Figure 5.35: Flexcam Link θ Response with a $\pi/8$ Input

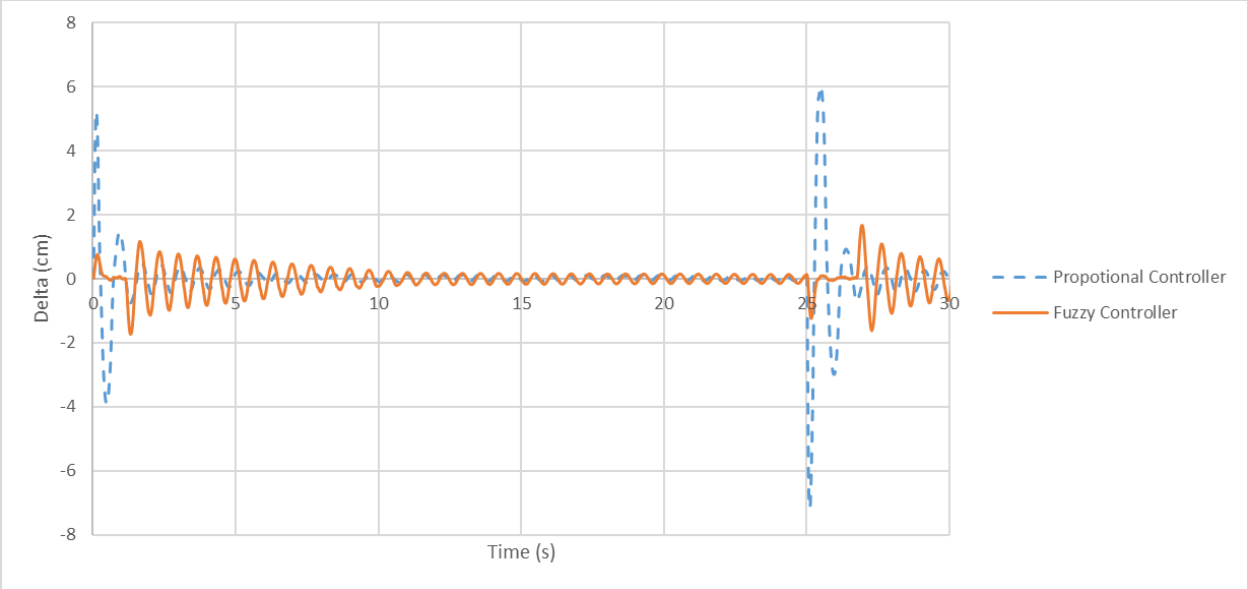

Figure 5.36: Flexcam Link δ Response with a $\pi/8$ Input

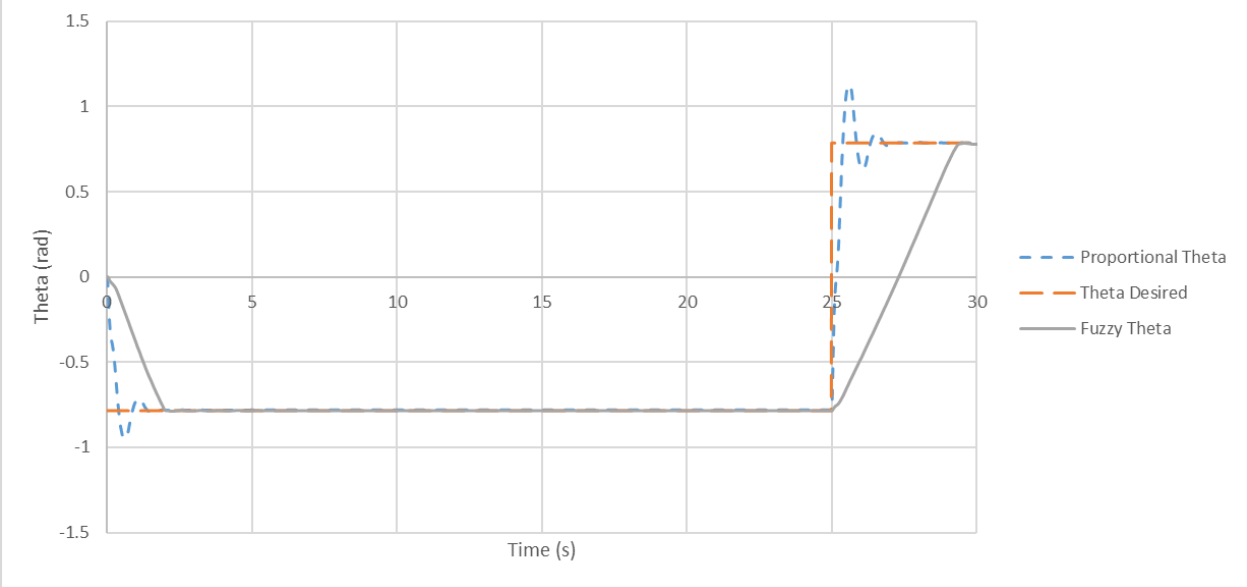

Figure 5.37: Flexcam Link θ Response with a $\pi/4$ Input

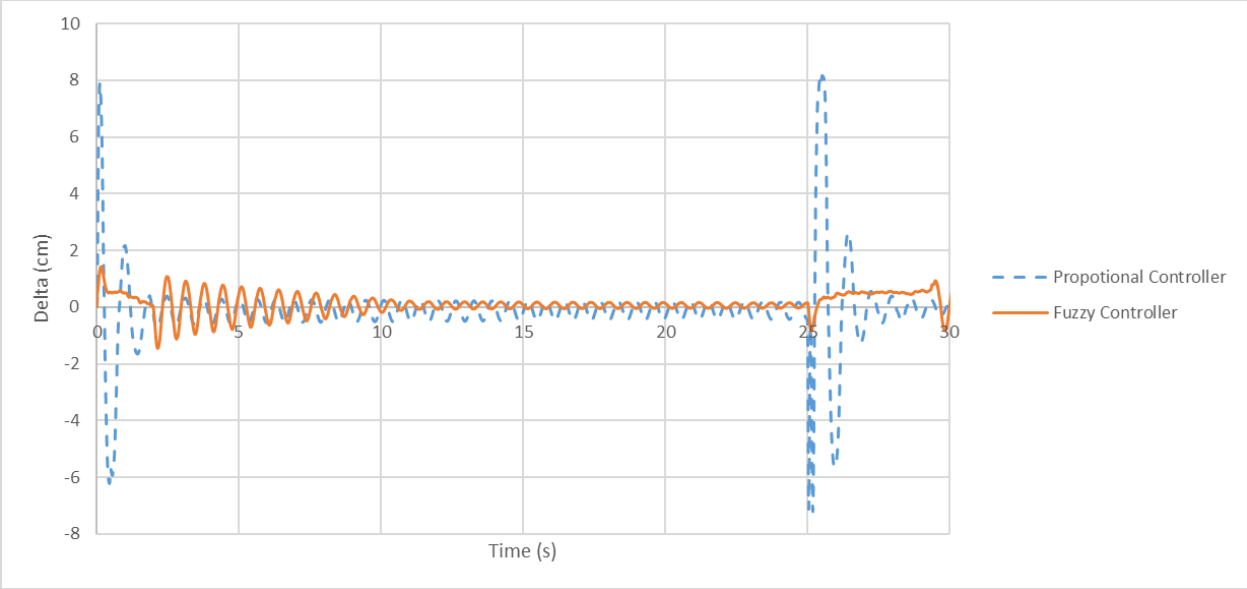

Figure 5.38: Flexcam Link δ Response with a $\pi/4$ Input

The Fuzzy Logic Controller was able to reduce the tip deflection and eliminate the steady-state error of the link. The results of the testing can be seen in Table 12. The Fuzzy Logic Controller reduced the end point deflection by up to 81.65% and was able to eliminate the steady-state error which ranged from 0.0031 rad to 0.0092 rad with a proportional controller.

This shows that the Fuzzy Logic Controller was able to reduce the tip deflection of the link while moving the link to its desired position with minimal tuning required. A model of the system was not required and the Fuzzy Logic Controller was able to effectively be incorporated into the system and achieve satisfactory results.

Table 12: Flexcam Link Experimental Data

	π/16 Input	π/8 Input	π/4 Input
Proportional Controller θ Error (rad)	0.0092	0.0092	0.0031
Proportional Controller Percent Overshoot (%)	19.53	20.31	18.75
Fuzzy Logic Controller Percent Overshoot (%)	0	1.17	0.39
Proportional Controller Max Tip Displacement (cm)	2.62	5.23	7.90
Fuzzy Logic Controller Max Tip Displacement (cm)	0.99	1.73	1.45
δ Percent Improvement (%)	62.21	66.92	81.65

5.5 Controller Adjustments

After completing the experimental verification, it was noted that a very large e_θ gain value of 75 was needed in order to reduce the steady-state error of the system to zero as the encoder is able to measure very small differences in angular position. This caused the controller to still produce a torque of zero if the e_θ was within 0.006 rads unless a large gain value was used. The membership functions for the e_θ were then adjusted as seen in Figure 5.39 and the new controller was compared against the original controller on the Flexcam link with a $\pi/4$ input and an e_θ gain of 23, a δ gain of 0.1, and a torque gain of 0.25 which produced the results shown in Figure 5.40, Figure 5.41, and Table 13.

The key changes to the membership functions were that the zero membership function was limited to only cover from -0.08 to 0.08 to limit the effect of the zero membership function at all unless the e_θ is within 0.0035 rads with the gain of 23. Next the small positive and negative membership functions were changed to trapezoidal membership functions that extend with values of 1 up to an e_θ of 0.1 and then moves to a value of 0 when e_θ is 0 linearly afterwards. While this change does not affect the membership function's overlap between the small and medium membership functions, as e_θ becomes smaller, it maintains a larger value in the small membership function, meaning a torque is applied at smaller e_θ values.

The new controller moved into position slightly faster than the original controller as seen in Figure 5.40. It was also able to completely eliminate any steady-state error despite the e_θ gain only being 23. This is compared with the original controller which maintained a steady-state error of 0.0061 rads. The updated controller had more residual motion after reaching its desired position which was caused by the controller attempting to eliminate the e_θ more precisely which took extra time. The updated controller also was able to reduce the steady-state vibration in the link more effectively than the original controller, reducing the vibration to 0.10 cm within 12 seconds while the original controller was still oscillating at 0.25 cm at 12 seconds.

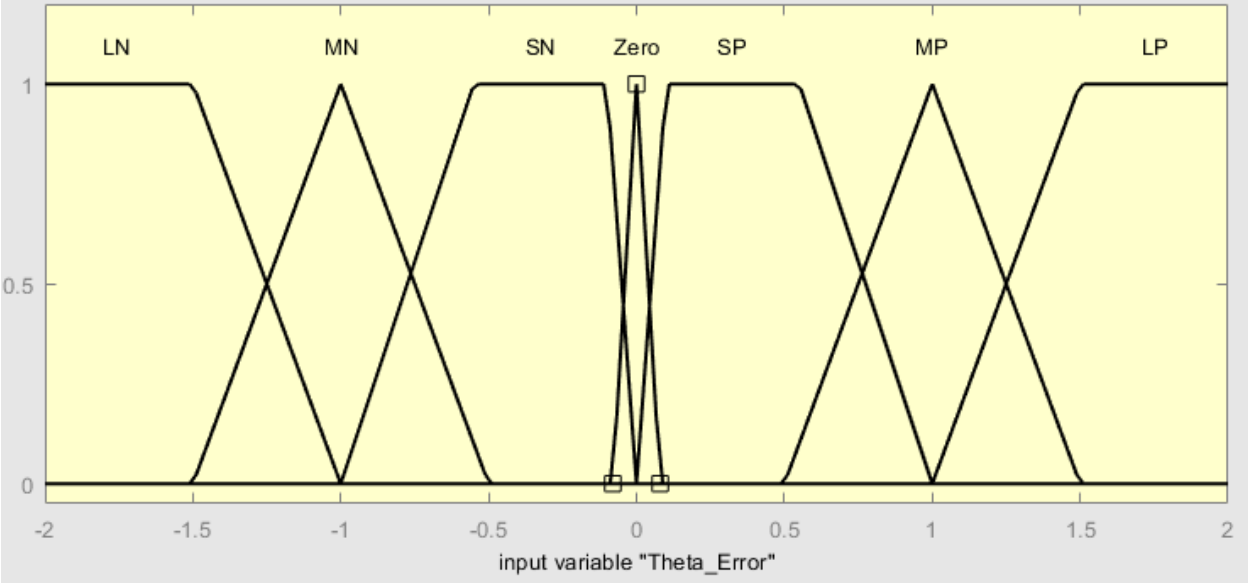

Figure 5.39: Updated e_θ Membership Functions

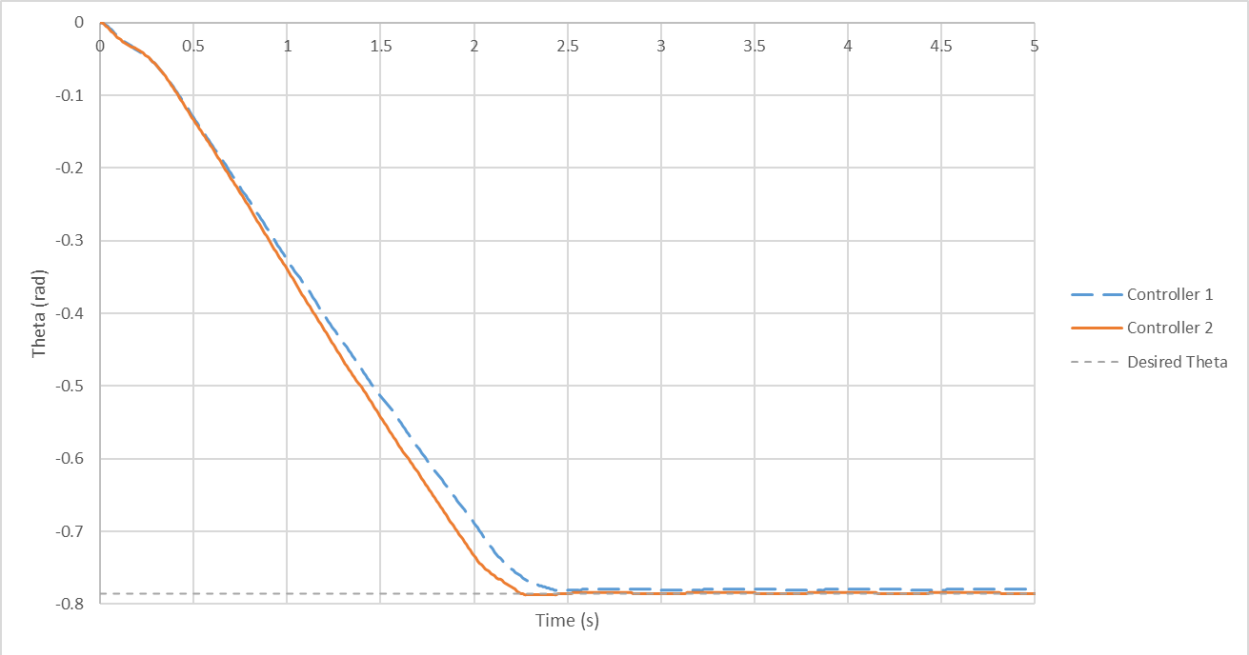

Figure 5.40: Fuzzy Logic Controller Comparison θ Response

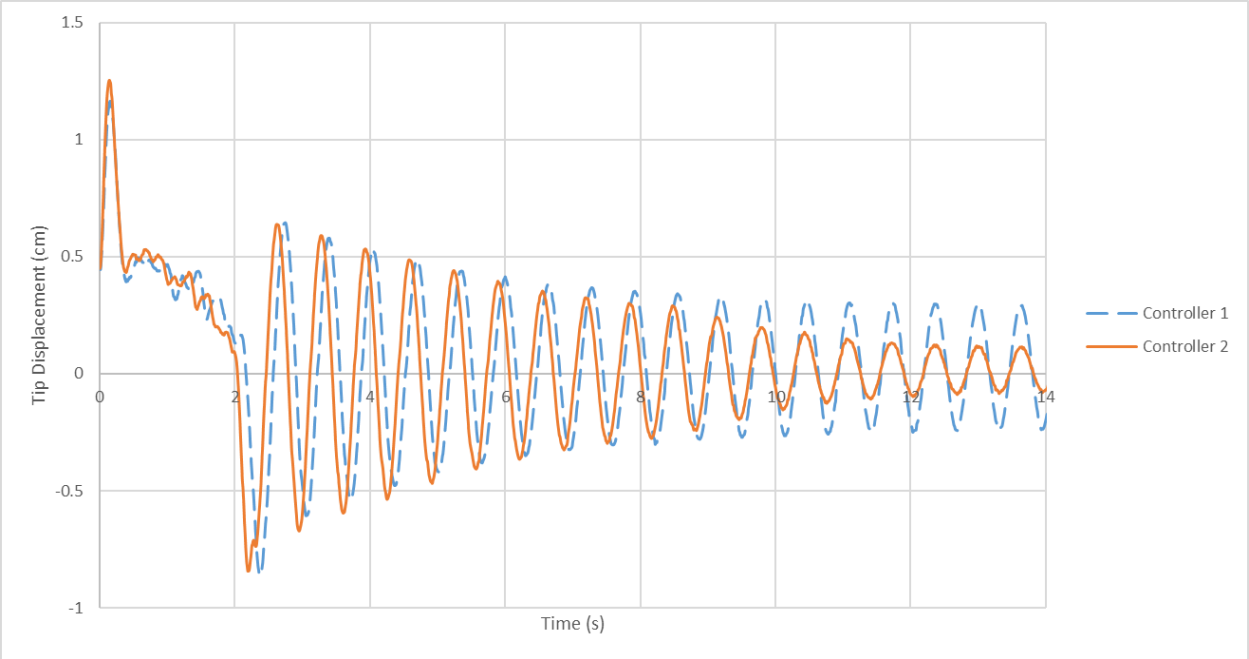

Figure 5.41: Fuzzy Logic Controller Comparison δ Response

Table 13: Updated Fuzzy Logic Controller Comparison Data

	Original Fuzzy Logic Controller	**Updated Fuzzy Logic Controller**
e_θ (rad)	0.0061	0
Percent Overshoot (%)	0	.20
Max Tip Displacement (cm)	1.16	1.25
Tip Displacement After 12 s (cm)	0.25	0.10

6 Conclusions and Recommendations

In this research, a Fuzzy Logic Controller was developed and implemented to control the end point vibration of a flexible link on a rotating hub. The objective was to design a controller that could effectively dampen the vibrations of a flexible link, thereby enhancing the stability and performance of the system. The results obtained in Chapters 4 and 5 demonstrate the effectiveness of this Fuzzy Logic Controller at dampening vibration.

The performance of the Fuzzy Logic Controller was evaluated through simulations and experiments. The simulations were conducted using MATLAB Simulink and MATLAB Simscape while the experiments were conducted using a physical test rig. The results obtained from both the simulations and experiments demonstrated that the Fuzzy Logic Controller was able to effectively dampen the vibrations of the flexible links.

The success of this research highlights the potential of Fuzzy Logic Controllers as a viable option for controlling the dynamics of flexible structures. The use of Fuzzy Logic provides a means of incorporating expert knowledge into the control process more easily and intuitively than it has been previously possible to do, which can lead to improved performance in flexible structures.

The Fuzzy Logic Controller was able to be incorporated within two different links that had very different properties with minimal tuning required before effective results could be obtained. This emphasizes one of the key benefits of Fuzzy Logic Controllers, being that the non-modal nature allows them to be quickly and easily incorporated into just about any system that operates in the same (or similar) manner to the system the controller was designed for. This enables an economy of effort where controllers can be designed for broader purposes and over time be fine tuned and tested in multiple environments to

obtain scaling factors.

In conclusion, the development and implementation of a Fuzzy Logic Controller for the control of the end point vibration of two different flexible links has been demonstrated in this research. The success of the controller in effectively dampening the vibrations of the flexible links highlights the potential of Fuzzy Logic Controllers for the control of flexible structures.

6.1 Recommendations for Future Work

More research should be completed to determine the optimal methods of determining the scaling factors to be used with a Fuzzy Logic Controller as they often need to be adjusted heuristically. Once this research has been completed, an area for further study should be to take this existing controller and apply it to a wide range of links. These links should vary in length, thickness, material, and any other properties that could affect the performance of the controller. Testing should then be completed to evaluate what gains should be used for the angular position error, tip deflection, and torque to achieve optimal results. These results can then be compiled and extrapolated from to develop charts and tables for quick reference. Thereby making it so that if someone is looking to control a link using this Fuzzy Logic Controller, instead of going through the tuning process, they would simply have to look up the properties of their link in the table to find the ideal gain values. This would save time for others in the future who want to incorporate this Fuzzy Logic Controller into their flexible structure.

Also, there was no comparison between the Fuzzy Logic Controller and alternative controllers. Another area for development could be to compare the effectiveness of this controller against many different types of controllers. This will help whoever is deciding what type of controller to use in the future make a more informed choice about what

controller is the best choice base on their specific priorities.

References

[1] Canadian Space Agency. Canadarm2. https://www.asc-csa.gc.ca/eng/iss/canadarm2/about.asp.

[2] Dong Y. Dynamic analysis and position control of a single flexible-link flexible-joint robot manipulator using time delay. Master's thesis, Royal Military College of Canada, National Library of Canada, Ottawa, ON, 2007.

[3] Theodore R. and A. Ghosal. Comparison of the assumed modes and finite element models for flexible multilink manipulators. *The International Journal of Robotics Research*, 14(2):91–111, 1995.

[4] K.V. Shihabudheen Reddy M.P.P. and J. Jacob. Precise non linear modeling of flexible link flexible joint manipulator. *International Review on Modelling and Simulations*, 5(3):1368–1374, 2012.

[5] Subudhi B. and A.S. Morris. Dynamic modelling, simulation and control of a manipulator with flexible links and joints. *Robotics and Autonomous Systems*, 41:257–270, 2002.

[6] Deepshikha and R. Kumar. Comparative study of conventional controllers. *International Journal of Electrical, Electronics and Data Communication*, 2(10):17–20, 2014.

[7] S. Le. Active vibration control of a flexible beam. Master's thesis, San Jose State University, 2009.

[8] A.R. Plummer Manning W.J. and M.C. Levesley. Vibration control of a flexible beam with integrated actuators and sensors. *Smart Materials and Structures*, 9:932–939, 2000.

[9] Jnifene A. and A. Fahim. A computed torque/time delay approach to the end point control of a one-link flexible manipulator. *Dynamics and Control*, 7:171–189, 1997.

[10] Lotfi Zadeh. Fuzzy sets. *Information and Control*, 8:338–353, 1965.

[11] Zimmermann H.G. *Fuzzy Set Theory and its Applications*. New York: Springer Science+Business Media, 2001.

[12] E.H. Mamdani. Application of fuzzy algorithms for control of simple dynamic plant. *Proceedings of the Institution of Electrical Engineers*, 121(12):1585 – 1588, 1974.

[13] M.J. Fricker. Adaptive control and vibration suppression of a flexible-link manipulator using fuzzy logic. Master's thesis, Royal Military College of Canada, National Library of Canada, Ottawa, ON, 2003.

[14] W. Andrews. Active vibration control of a flexible structure using tools of fuzzy logic and neural networks. Master's thesis, Royal Military College of Canada, National Library of Canada, Ottawa, ON, 2001.

[15] Jnifene A. and W. Andrews. Fuzzy logic control of the end-point vibration in an experimental flexible beam. *Journal of Vibration Control*, 10(4):493–506, 2004.

[16] Sun C. H. Gao, W. He and Y. Yu. Fuzzy neural network control of a flexible robotic manipulator using assumed mode method. *TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS*, 29(11):5214–5227, 2018.

[17] H.H. Cudney Cao W. and R. Waser. Smart materials and structures. *Proc. Natl. Acad. Sci. USA*, 96:8330–8331, 1999.

[18] Li Y.F. and Chen X.B. End-point sensing and state observation of a flexible-link robot. *TRANSACTIONS ON MECHATRONICS*, 6(3):351–356, 2001.

[19] D.J. Jehan. Effect of sensor placement on measuring higher modes of vibration for a thin flexible cantilever beam in free vibration. Master's thesis, Royal Military College of Canada, National Library of Canada, Ottawa, ON, 2008.

[20] Singiresu S. Rao and Philip Griffin. *Mechanical Vibrations*. Pearson, 2018.

[21] John R. Taylor. *Classical Mechanics*. University Science Books, 2005.

[22] Katalin M. Hangos; József Bokor and Gábor Szederkényi. *Analysis and Control of Nonlinear Process Systems*. Springer, 2004.

[23] William L. Brogan. *Modern Control Theory*. UQuantum Publishers, Inc., 1974.

[24] O.A. Bauchau and J.I. Craig. *Structural Analysis - With Applications to Aerospace Structures*. Springer, 2009.

[25] Vinesh V. Nishawala. A study of large deflection of beams and plates. Master's thesis, Rutgers University, New Brunswick, New Jersey, 2011.

[26] University of Iowa Physics and Astronomy. Small angle formula. `https://itu.physics.uiowa.edu/glossary/small-angle-formula`.

[27] Guanrong Chen and Trung Tat Pham. *Introduction to Fuzzy Sets, Fuzzy Logic, and Fuzzy Control Systems*. CRC Press, 2001.

[28] Balasem Salem Sumait Omar Adil M. Ali1, Aous Y. Ali1. Comparison between the effects of different types of membership functions on fuzzy logic controller performance. *International Journal of Emerging Engineering Research and Technology*, 3(3):76–83, 2015.

[29] Amrit Kaur Arshdeep Kaur. Comparison of mamdani-type and sugeno-type fuzzy inference systems for air conditioning system. *International Journal of Soft Computing and Engineering*, 2(2):323–325, 2012.

[30] Nacy S. O.F. Fakhri and R. Khaleel. Tip deflection control for a rotating flexible link. *American Journal of Mechanical Engineering*, 4:163–168, 08 2016.

[31] Vitaliy Sergeevich Geraschenko, Andrey Sergeevich Grishin, Nataly Igorevna Gartung. Approaches for the Calculation of Rayleigh Damping Coefficients for a Time–History Analysis. *WIT Transactions on The Built Environment*, 180:227–237, 2018.

[32] Athol J. Carr. *Ruaumoko Manual, Volume 1: Theory*. University of Canterbury, 2007.

[33] Sourabha Havaldara and Uday N Chate. Estimation of modal damping ratio from specific shear modulus for monolithic materials and hybrid cored multilayer composites. *Procedia Materials Science*, 10:124–138, 2015.

[34] Noor Dhaher. *MATERIALS SCIENCE AND ENGINEERING*. International Energy and Environment Foundation, 07 2019.

[35] AMETEK Haydon Kerk Pittman. Ametek pittman gm14902s020-r1 lo-cog dc servo gearmotor. `https://cdn.shopify.com/s/files/1/0568/0716/2056/files/Ametek_Pittman_GM14902S020.pdf?v=1658776403`.

[36] R. C. Hibbeler. *Mechanics of Materials*. Pearson, 2015.

[37] Kapp C. Actuators and sensors in mechatronics - optical encoders. `http://engineering.nyu.edu/mechatronics/Control_Lab/Criag/Craig_RPI/SenActinMecha/S&A_Optical_Encoders.pdf`.

[38] Omega. Compact strain gauge bridge completion module. `https://www.omega.ca/en/force-and-strain-measurement/strain-gauges/strain-gauge-accessories/p/BCM-2-Series`.

[39] Alan V. Oppenheim and Alan S. Willsky with S. Hamid Nawab. *Signals and Systems*. Prentice Hall, 1997.

A Matrices

A.1 Mass Matrix

$$[M] = \frac{\rho Al}{420} \begin{bmatrix}
(\frac{420 J_{base}}{\rho Al} + 4l^2) & 13l & -3l^2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
13l & 312 & 0 & 54 & -13l & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-3l^2 & 0 & 8l^2 & 13l & -3l^2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 54 & 13l & 312 & 0 & 54 & -13l & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -13l & -3l^2 & 0 & 8l^2 & 13l & -3l^2 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 54 & 13l & 312 & 0 & 54 & -13l & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -13l & -3l^2 & 0 & 8l^2 & 13l & -3l^2 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 54 & 13l & 312 & 0 & 54 & -13l & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & -13l & -3l^2 & 0 & 8l^2 & 13l & -3l^2 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 54 & 13l & 312 & 0 & 54 & -13l \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -13l & -3l^2 & 0 & 8l^2 & 13l & -3l^2 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 54 & 13l & 156 & -22l \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -13l & -3l^2 & -22l & 4l^2
\end{bmatrix}$$

A.2 Stiffness Matrix

$$[K] = \frac{EI}{l^3}\begin{bmatrix}
4l^2 & -6l & 2l^2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
-6l & 24 & 0 & -12 & 6l & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
2l^2 & 0 & 8l^2 & -6l & 2l^2 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & -12 & -6l & 24 & 0 & -12 & 6l & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 6l & 2l^2 & 0 & 8l^2 & -6l & 2l^2 & 0 & 0 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & -12 & -6l & 24 & 0 & -12 & 6l & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 6l & 2l^2 & 0 & 8l^2 & -6l & 2l^2 & 0 & 0 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & -12 & -6l & 24 & 0 & -12 & 6l & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 6l & 2l^2 & 0 & 8l^2 & -6l & 2l^2 & 0 & 0 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & -12 & -6l & 24 & 0 & -12 & 6l \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 6l & 2l^2 & 0 & 8l^2 & -6l & 2l^2 \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & -12 & -6l & 12 & -6l \\
0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 6l & 2l^2 & -6l & 4l^2
\end{bmatrix}$$

B Link Model Matlab Code

```
clc
clear all

rho=2700;           %beam density
E=70*10^9;          %Youngs Modulus of beam
G=25*10^9;          %Shear Modulus of beam
L=1.12;             %beam length
l=L/6;              %element length
h=0.0508;           %height of beam
w=0.00228;          %thickness of beam
A=h*w;              %beam cross sectional area
I=((w^3)*h)/12;     %moment of inertia of beam
Jbase=0.00012;      %mass moment of inertia of the motor
%Jbase=0;

Z=zeros(13);        %Zero matrix
ZZ=zeros(26,13);    %Zero matrix number 2
II=eye(13);         %Identity matrix

%Mass Matrix
M=((rho*A*l)/420)*[((Jbase*420/(rho*A*l))+4*l^2) 13*l -3*l^2 0 0 0 0 0 ...
    0 0 0 0 0; 13*l 156*2 0 54 -13*l 0 0 0 0 0 0 0 0; -3*l^2 0 8*l^2 13*l ...
    -3*l^2 0 0 0 0 0 0 0 0; 0 54 13*l 156*2 0 54 -13*l 0 0 0 0 0 0; 0 ...
    -13*l -3*l^2 0 8*l^2 13*l -3*l^2 0 0 0 0 0 0; 0 0 0 54 13*l 156*2 ...
    0 54 -13*l 0 0 0 0; 0 0 0 -13*l -3*l^2 0 8*l^2 13*l -3*l^2 0 0 0 0; ...
    0 0 0 0 0 54 13*l 2*156 0 54 -13*l 0 0; 0 0 0 0 0 -13*l -3*l^2 0 ...
    8*l^2 13*l -3*l^2 0 0; 0 0 0 0 0 0 0 54 13*l 2*156 0 54 -13*l; 0 ...
    0 0 0 0 0 0 -13*l -3*l^2 0 8*l^2 13*l -3*l^2; 0 0 0 0 0 0 0 0 0 ...
    54 13*l 156 -22*l; 0 0 0 0 0 0 0 0 0 -13*l -3*l^2 -22*l 4*l^2];

```

```
%Stiffness Matrix
K=((E*I)/l^3)*[4*l^2 -6*l 2*l^2 0 0 0 0 0 0 0 0 0 0; -6*l 24 0 -12 ...
    6*l 0 0 0 0 0 0 0 0; 2*l^2 0 8*l^2 -6*l 2*l^2 0 0 0 0 0 0 0 0; ...
    0 -12 -6*l 24 0 -12 6*l 0 0 0 0 0 0; 0 6*l 2*l^2 0 8*l^2 -6*l ...
    2*l^2 0 0 0 0 0 0; 0 0 0 -12 -6*l 24 0 -12 6*l 0 0 0 0; 0 0 0 ...
    6*l 2*l^2 0 8*l^2 -6*l 2*l^2 0 0 0 0; 0 0 0 0 0 -12 -6*l 24 0 ...
    -12 6*l 0 0; 0 0 0 0 0 6*l 2*l^2 0 8*l^2 -6*l 2*l^2 0 0; 0 0 0 ...
    0 0 0 -12 -6*l 24 0 -12 6*l; 0 0 0 0 0 0 0 6*l 2*l^2 0 8*l^2 ...
    -6*l 2*l^2; 0 0 0 0 0 0 0 0 0 -12 -6*l 12 -6*l; 0 0 0 0 0 0 0 ...
    0 0 6*l 2*l^2 -6*l 4*l^2];

%Torque Vector
Tau=[1; 0; 0; 0; 0; 0; 0; 0; 0; 0; 0; 0; 0];

zeta=1.2*((G/rho)^(-1/3));
omegamax=476.30;
omegamin=41.19;

alpha=2*zeta*(omegamax*omegamin)/(omegamax+omegamin);
beta=2*zeta/(omegamax+omegamin);

C=alpha*M+beta*K;

AA=K*inv(M);
V=eig(AA);
Vhertz=sqrt(V)/(2*pi);

fuzzy;
%select file -> import -> import from file -> CONTROLLER
%select file -> export -> export to workspace -> CONTROLLER
```

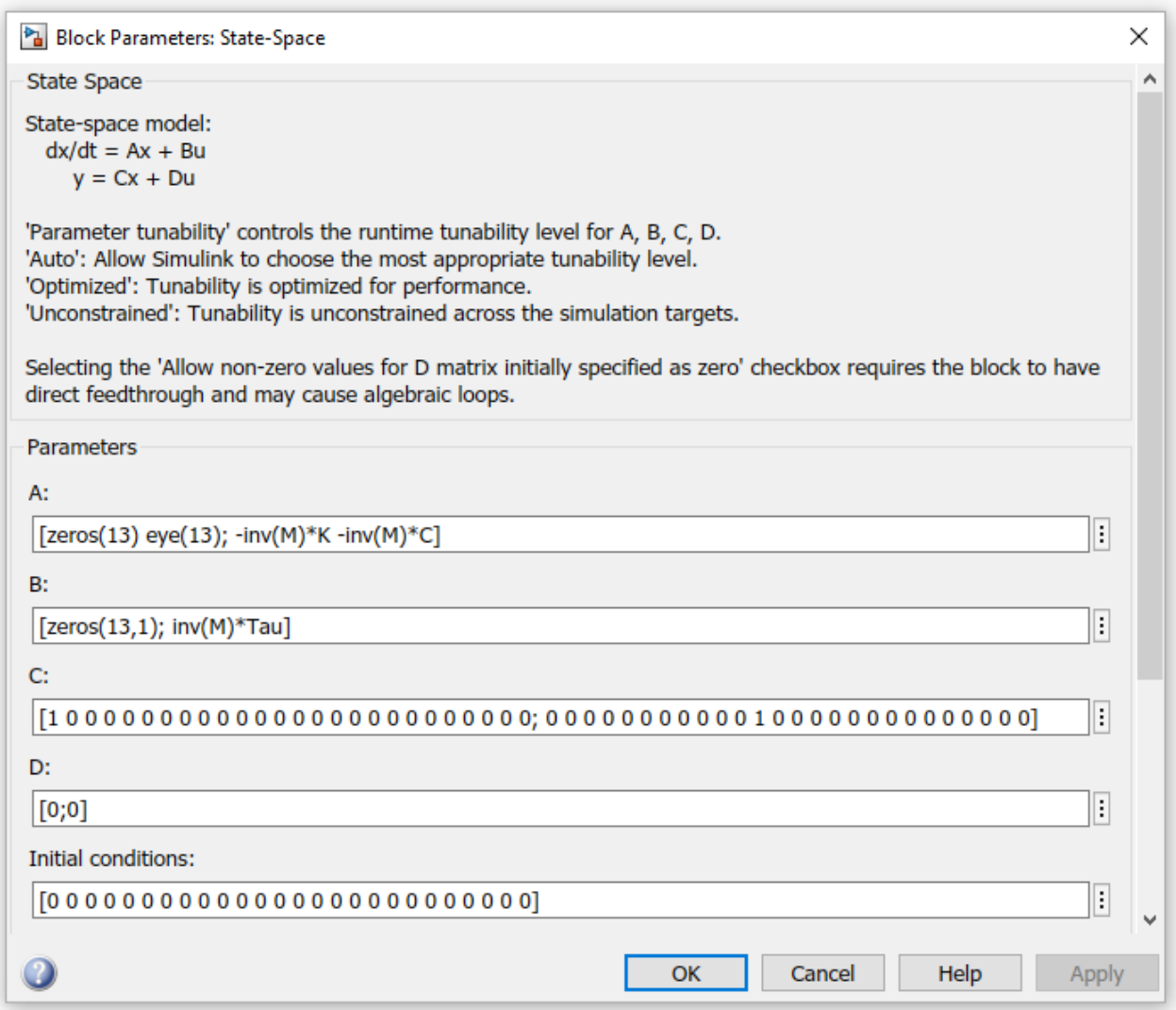

Figure B.1: State-Space Simulink Block Matrix Inputs