School of Mechanical and Manufacturing Engineering

Faculty of Engineering

The University of New South Wales

Experimental Modelling and Control of a Robotic Arm

by

Mei Yan Tang

Thesis submitted as a requirement for the degree of

Bachelor of Engineering in Mechatronic Engineering

Submitted: November 20, 2020 Student ID: z5129009

Supervisor: Dr. Mohammad Deghat (UNSW)

Originality Statement

'I hereby declare that this submission is my own work and to the best of my knowledge it contains no materials previously published or written by another person, or substantial proportions of material which have been accepted for the award of any other degree or diploma at UNSW or any other educational institution, except where due acknowledgement is made in the thesis. Any contribution made to the research by others, with whom I have worked at UNSW or elsewhere, is explicitly acknowledged in the thesis. I also declare that the intellectual content of this thesis is the product of my own work, except to the extent that assistance from others in the project's design and conception or in style, presentation and linguistic expression is acknowledged.'

Signed: 

Date: November 20, 2020

Abstract

Robot manipulators with highly flexible joints are widely researched as it remains as one of the most challenging systems to control. Other factors such as model uncertainties and external disturbances, could also increase the difficulty of the control problem. Hence, robust and adaptive advanced control algorithms are designed to overcome these problems. Four controllers with robust/adaptive features were designed and tested on a 2 DoF serial robot manipulator with highly flexible joints. Results concluded that robust/adaptive controllers were able to solve the regulation and tracking problem in the presence of uncertainties. The controllers were also tested on the same type of robot with rigid joints to highlight the effects of joint flexibility.

Acknowledgements

Firstly, I would like to thank my supervisor, Dr Mohammad Deghat, who provide constant support, guidance and advice in completing this thesis.

I would also like to thank the technical officer of the Control and System Lab from the School of Electrical Engineering and Telecommunications, Zhengyu Liu, for granting me remote lab access and providing support during my experiments.

Thank you to Dr Arash Khatamianfar for providing useful resources from the EET Control Systems Course (ELEC3114).

Thanks to all my friends that helped me along the way.

Special thanks to my mom and dad for *attempting* to read my thesis.

Abbreviations

DoF Degrees of Freedom

PD Proportional-Derivative

SMC Sliding Mode Controller

MRAC Model Reference Adaptive Control

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In the industry of automation, robot manipulators have come a long way since its breakthrough of the first industrial robot arm, Unimate in the 1961. Since then, millions of jobs have been replaced by robots and productivity in many industries have improved greatly. Robots are able to do boring and tedious tasks efficiently that are refused by humans or even work in dangerous environment that pose a hazard to the human health. As the field of robotics have advanced, robot arm manipulators can be used in many different applications. From desktop robot manipulators for educational and recreational purposes to precision industry robots, the potential of robots is endless. This means that the robots of many different characteristics and designs are being invented for different purposes. One of the common characteristics of most of the robots is the presence of flexibility in the joints.

Joint flexibility is needed as it provides some compliance when in contact with unspecified environment and allow human interaction with the robot safely [1]. However, it is important to highlight the significance of flexible joints as it could contribute to unwanted vibrations and position errors if it is not acknowledged. Although, this issue can be solved if the accurate representation of the dynamics model of the robot is available, robots that possess flexibility have very complex model and it is hardly ever accessible. Moreover, robots work in environments that are not ideal, uncertainties like disturbances are likely to affect the performance of the robot. These errors could cause the robot to lose its precision and perform poorly compared to

its expected outcome. Hence, research have been done to compensate these errors by designing controllers that are more robust and less susceptible to the external nonlinearities. It is useful to discover the capabilities of a controller as the flexible joint robot could adopt the precision of a rigid joint robot and still possess its flexibility.

The aim of this thesis is to investigate how advanced control algorithms can be used to compensate the uncertainties and nonlinearities in the system and also the external disturbance. In addition, the motion of the robot arm will be carefully monitored and analysed when the controller is applied to understand the purpose of the control algorithm. This research also aims to identify the significance of elasticity in the joints by comparing the results between rigid joint robots and flexible joint robots. The performance of different advanced control algorithms will also be compared.

1.1 Outline

Chapter 2 presents the literature review for this thesis. Chapter 3 will be on the methodology, where the derivation of the robot models, controller design, and description and configuration of the physical robot used. Chapter 4 will be mainly discussing the results obtained from the simulations and experiments. Also, the results will be analysed and compared. Chapter 5 will provide a conclusion of the thesis and potential future works.

Chapter 2

Literature Review

2.1 Types of Robot Manipulator

Robot arm manipulators have a long history of assisting mankind in industries like manufacturing, transport and medical. It has been almost 60 years since the first robot manipulator has been deployed to work in industries and since then many different kinds of robot have been created [2]. Subsequently, the field of robot arm manipulators have been researched extensively.

Rigid robot manipulators usually incorporate heavy materials to maximize stiffness which reduces vibration of the end-effector, hence leads to better accuracy [3]. However, the drawback of using heavy material is the inefficiency of power consumption and low speed which defies industrial productivity [3]. As technology advances, the performance, efficiency, complexity, and flexibility of robot manipulator increases. Flexibility in robot manipulators can be categorized in two groups, link flexibility and joint flexibility. Robots with link flexibility have a great advantage of lighter arms and therefore higher in speed and lower in cost operation, but, vibration due to low stiffness will occur [3]. On the other hand, joint flexibility is currently common in industrial robots where motion transmission and reduction elements are used [4]. Although, previously joint flexibility was considered as a limiting factor for the performance of the manipulator [5], it is now an important factor as it improves dynamic efficiency [4]. Besides that, the added flexibility between the motor and the load is beneficial to avoid large impact

against the target, which will either damage the target or the robot, when there is lack of accuracy in the robot [2].

In a study by Zhang and Zhou [6], flexible robot arm manipulator two flexible links and one rigid link connected by flexible joints were simulated and compared against its equivalent of a traditional rigid robot manipulator. It was concluded the presence of flexibility in the joint and links results in fluctuations in the angular velocity of the links which ultimately cause tool tip position and velocity deflections while there were negligible fluctuations in rigid robot arms [6]. This study supports the fact that vibrations are an occurring issue for flexible robots due to its physical properties.

In this thesis, a 2 DoF serial flexible joint robot and the equivalent 2 DoF rigid robot is studied.

2.2 Modelling Robot Manipulator

Modelling is important in simulation and controller design. When modelling a system, the model should be as similar representation of the actual system to achieve accurate simulation and effective controller design, especially in a model-based controller. In the study by Zhang and Zhou [6] mentioned before, it outlined the importance of applying the suitable dynamical model by illustrating the significance of flexibility, as the rigid and flexible robot manipulator yielded different outcomes with the same input. However, the exact model of the robots is not always available, but there are a few general types of dynamic model that are widely used.

There are three types of dynamics models that are considered this thesis, the rigid dynamics model, the linear joint dynamics model and the non-linear joint dynamics model. All three dynamic models have been used by fellow researchers in their studies.

2.2.1 Rigid Dynamics Model

The rigid dynamic model can be derived by using the Euler-Langrange method, utilizing the kinetic and potential energies stored in the system. A typical dynamics model of a rigid robot

which can be found in [7], is described by the equation below.

$$M(q)\ddot{q} + C(q,\dot{q}) + G(q) = \tau \tag{2.1}$$

M is the robot inertia matrix, C is the Coriolis and centrifugal forces matrix, G represents the gravitational terms, q is the vector of joint variables, and τ represents the vector of input torques. This model assumes that the coupling between the actuator and rigid links are absolutely rigid. Although, perfectly rigid robot does not exist, this model approximation is sufficient to describe rigid robots. In this thesis, the rigid dynamics model will be used for the rigid joint robot used.

2.2.2 Linear Joint Dynamics Model

In the case of flexible joints robot, the rigid dynamics model is not a good approximation of the robot due to the added flexibility in the joint. Hence, there is a better model that also utilizes the Euler-Langrange method to obtain the dynamic model, which is the linear joint dynamics model, which some refers to the simplified flexible joint model, that was suggested in a research by Spong [8]. The general dynamics equation is given as,

$$M(q_1)\ddot{q_1} + C(q_1,\dot{q_1}) + G(q_1) + K(q_1 - q_2) = 0 \tag{2.2}$$

$$J\ddot{q_2} - K(q_1 - q_2) = \tau \tag{2.3}$$

There is an additional set of joint variables that is introduced in this equation, q_1 represents the joint variables of the link, while q_2 represents the joint variables of the actuator. In addition to the M, C and G matrix mentioned in equation (2.1), K is the diagonal matrix of joint stiffness coefficients, J is the diagonal matrix of actuator inertia. The linear joint dynamics model includes joint flexibility by modelling each joint as a linear torsional spring of constant stiffness. Although, it is an approximate, it will still contribute to the improvement of the performance of the robot.

2.2.3 Nonlinear Joint Dynamics Model

In addition, studies showed that nonlinear effects exist, for example, friction, shaft windup and nonlinear stiffness. Even though, the previously mentioned models could still provide promising

results if the controller design could compensate the nonlinear deviations, it is always better to obtain the dynamics model that is closest to the actual robot. Therefore, in order to model a greater representation of the robot manipulators, research have been done to include nonlinearities. Tuttle et al. [9] showed that harmonic drives exhibit nonlinear dynamic behaviours, such as transmission friction, compliance, and kinematic errors. It is crucial to note this as harmonic drives are commonly used in robot manipulators as it provides high torque transmission and zero backlash. In the study by Ulrich et al. [10], nonlinearities like friction, nonlinear stiffness, soft-windup and inertial cross-coupling is taken into consideration for the dynamics model to further understand the nonlinear effects that can be observed in the experiment. It was mentioned that the understanding of flexible effects in joints will help in developing an accurate model which leads to a higher success in controller design [10].

It is important to note that nonlinear effects exist in any type of robots, even rigid joint robots. These effects could be a potential concern in this study or probable future work idea to improve performance of the robot. Despite presence of nonlinear effects, it is understood that controllers can compensate these effects. Hence, one of the aims of this thesis is to identify and design controllers that are capable to make up these nonlinearities without the exact representation of the robot model.

2.3 Control Algorithms

There are mainly two classes of control objectives, which are the regulation or point-to-point control and the tracking control. The difference between the two objectives is that the goal of regulation is to control the joint variable about the desired position without considering torque disturbances and initial conditions. Therefore, transients and overshooting behaviour might occur. On the other hand, for tracking control, the aim is to follow a desired trajectory while taking disturbances, maximum velocity, acceleration, and other unmodelled factors into account. In this section, an overview of different type of controllers that has been developed since the earlier days to the most recent research will be described.

2.3.1 Regulation

Regulators are often used when the only goal is to approach the desired position, hence, unwanted transient behaviour might affect the motion of the robot arm during the trajectory. An example of a regulator is the proportional-derivative (PD) controller, which is one of the basic controllers that has been researched intensively. One of the earlier studies done by Tomei [11] has shown that a simple PD controller with gravity compensation that is used for rigid joint manipulators, can achieve global asymptotic stability on flexible joint robots and link errors will converge to zero. The controller includes a linear constant feedback from the position and velocity of the motor rotors, and the gravity term. The PD controller is also known to be robust with respect to some model uncertainties, especially with inertial and frictional parameters of the robot [11]. Even without the knowledge of these parameters, the PD controller can still achieve asymptotic stability [11]. On the other hand, stability will be affected when uncertainties exist in the gravitational or elastic term as it is directly related to the control law [11]. However, further studies [12] have highlighted that the disadvantage of previously mentioned PD controller is the noises in the velocity measurement could affect the performance. Additionally, even if the velocity is obtained by numerical differentiation, it is still susceptible to errors in low and high speed [12].

Hence, Kelly et al. [12] suggested replacing the motor rotor velocity measurement by approximate differentiation by using a high-pass filter. Results show that in the similar setting in [11], with added velocity noise, global asymptotic stability was still preserved and better tracking performance than Tomei's controller [11]. The tracking performance was compared by comparing the integral of square error, which the error was 0.571 rad^3 without using approximate differentiation, and 0.284 rad^3 by using approximate differentiation. In another study, Son et al. [13] demonstrated a regulation control law that only requires the position measurements similar to the research done by Kelly et al. [12]. Son et al. had proposed that a parallel connection of the robot system and an input-dimensional additional system can provide the effect of a desired differentiator to obtain the velocity measurements [13]. It was concluded that the control law suggested by Son et al. [13] surpass Kell et al.'s [12] method as it had a better approximation of the velocity measurement.

On the other hand, in the research of Tomei [11], a constant gravity term has been used for gravity compensation. However, the gravitational term depends on the link position variable which is not measurable in [11], and might not be available in practice. De Luca et al. [14] showed in their research that this issue can be solved with the on-line gravity compensation, where the link position variable is replaced by a gravity-biased modification of the measured motor position. At steady state, this method will approach to the correct gravity compensation, without the measurement of the link position [14]. The results proved that the controller with online gravity compensation surpassed the performance of the controller with the constant gravity compensation as transient behaviour have been reduced [14]. Besides that, the PD controller with on-line gravity compensation has an advantage of a larger range of workable proportional gain [14]. For example, in lower gains, the on-line gravity compensation method still achieved the desired configuration while with the constant gravity compensation, steady state errors are likely to happen [14].

So far, the controllers mentioned previously were done in the joint space where the joint configuration was considered. A study shows that a compliance controller which includes PD actions with either constant or on-line gravitational compensation can also be done in the Cartesian space or also known as task space [15]. Furthermore, this research also considered unforeseen contact between the environment and the robot, where it aims to minimize vibrational and chattering effects of the joint elasticity. As expected, the compliance controller with the on-line gravitational compensation have better performance in terms of transient behaviour. In the experiment, a load cell was present to measure the force course by an impact in the z-direction and it was concluded that the controller was able to regulate the robot compliance when a suitable proportional gain is selected.

In a recent study, Sun et al. [16] mentioned that there were some drawbacks on the on-line gravity compensation-based PD control law. The three main drawbacks were the functional non-collocated information was not utilized, the lack of overshoot analysis, and no effective damping term that could counter external torque [16]. In order to overcome these drawbacks, Sun et al. [16] have developed a nonlinear state feedback controller which is based on on-line gravity compensation with residual vibration suppression. There are two terms in the controller that were responsible to reduce overshoots and suppress residual vibrations [16]. This controller

	Joint 1			Joint 2		
Indexes	σ [%]	t_s [s]	e_{q1} [rad]	σ [%]	t_s [s]	e_{q2} [rad]
PBC	3.55	1.227	0.009	2.82	0.447	0.003
ERC	2.65	0.635	0.004	0.60	0.349	0.004
OCG	15.24	2.009	0.003	4.29	0.536	0.010
Proposed	0.50	0.376	0.002	2.01	0.253	0.003

Table 2.1: Performance Indices for First and Second Joint [16]

was also proved to be asymptotically stable like the previously mentioned controllers [16]. The study made a comparison with different control laws, for example, Proxy-based control (PBC), Energy-based control (EBC), and most importantly, PD control with on-line gravity compensation (OGC) [16]. Two experiments were conducted on a self-build flexible joint manipulator to compare these control algorithms, one without and one with external disturbance during the transient state, and the task of the experiment was to regulate on the desired joint position [16]. Table 2.1 shows the results of the first experiment without the external disturbance where σ is the percentage overshoot, t_s is the settling time and e_q is the steady state error [16]. It was obvious that the proposed nonlinear controller had better performance than the other controllers in terms of overshoot, settling time and steady state error. Moreover, other controllers have shown signs of oscillatory behaviour due to vibration, but the proposed controller have suppressed the vibrations successfully [16]. In the second experiment where external disturbance was introduced during the transient state, it was observed that the proposed controller, PBC and OCG showed good robustness against the external disturbance, but violent oscillations were experienced in PBC and OCG [16]. Meanwhile, although ERC does not show undesired oscillations, its steady state error was unacceptable [16]. Therefore, in conclusion of this research, the proposed nonlinear controller has illustrated superior performance against the compared control algorithms in terms of all aspects including overshoot, settling time, vibration suppression, and robustness against external disturbance [16]. However, an important note was that the control input was susceptible to noise for the proposed method, due to the one of the terms in the controller which amplifies the noises in the velocity reading [16].

On the other hand, another recent study by Kumar et al. [17] had made comparisons between different control techniques on a single-link flexible joint robot manipulator. The control techniques

includes proportional-integral-derivative (PID) controller, pole-placement, and linear-quadratic regulator (LQR) methods. It is known that tuning PID type controllers to its optimal have always been an occuring challenge, and many studies have used different strategies to properly tune a PID controller. One of the examples are the Fuzzy self-tuning PID semiglobal regulator by Meza et al. [18] which fuzzy logic was used to determine the nonlinear gains of the PID controller while considering several practical features in the real robot. In the study by Kumar et al. [17], the genetic algorithm methodology was used for self-tuning the PID controller. The basic concept of genertic algorithm used in this context for the optimization problem was similar to the biological explanation where 'the fittest survive to reproduce' [17]. While for pole-placement method, the desired pole locations was identified and the feedback gain was obtained by using the MATLAB command 'place' [17]. As for the LQR method, the weightage matrix was tuned by genetic algorithm and the controller feedback gain can be obtained [17]. The controller that performed the best wasthe genetic algorithm tuned PID controller, followed by LQR method, then pole-placement method [17]. It was observed that the PID controller showed faster response time and allows better control in flexible joint robots [17].

In this section, the evolution of controllers based on PD controllers have been described for the regulation objective. The change in the gravity term and noisy velocity measurements have greatly impact the performance of the robot. Additionally, the presence of joint flexibility introduces undesired oscillatory behaviours in the links, where this issue was eventually tackled by Sun et al. [16] by suppressing vibrations. On the other end of the spectrum, PID controllers showed promising results if it is optimumly tuned [17].

2.3.2 Tracking Control

It is known that tracking control has higher complexity than the regulation problem as a desired trajectory is followed with respect to the desired velocity and acceleration. This type of control is more often used in tasks that require more precision such as wielding, drawing and painting.

One of the classic examples of tracking control is the inverse dynamics control. The idea of this controller is to linearize and decouple robot manipulator dynamics and compensating nonlinearities by adding the required force into the control input [19]. It is known that the drawbacks

of the inverse dynamics control is the requirement of the exact robot model parameters and the high amount of computational burden [19]. Numerous researches have tackled this issue in various different techniques. In one of the earlier researches, Krzysztof et al. [20] have proposed an inverse dynamic control technique based on discrete time. It was suggested that backward differentiation formula methods from numerical methods can be used to solve differential-algebraic equations systems [20]. This approach provides a prediction scheme that was one-and-a-half-step ahead which leads to a less complex control process where a basic dynamic model can be used [20]. The decrease in the complexity of the control process could lessen the computational burden [20]. Another advantage of this approach was the support of low sampling frequencies [20]. The results of this study has deemed successful as it has showed good tracking properties and there was no overshoot, which is common in controllers like PID [20].

Besides that, to solve the tracking control problem, a dynamic feedback controller can be utilised. In an earlier study of Chang et al. [21] a dynamic feedback controller that only requires position measurements was proposed. The velocity was estimated by a Reduced-Order Observer which has half the dimension of the flexible joint robot [21]. By only taking the position measurements into account, velocity noises can be neglected. The simulation results showed that the trajectory was smooth and the errors in the trajectory converged to zero exponentially with a predetermined convergence rate that can be assigned arbitrarily [21]. In addition, robustness to uncertainties can be achieved by tuning the controller gains [21]. However, the exact knowledge of the physical robot must be known [21]. A more recent study has shown that robot state vector can be estimated along with the uncertainties in the physical robot by using Extended-State-Observer [22]. Talole et al. [22] have proposed an approach to utilize the states and uncertainties estimated by the Extended-State-Observer, then applying it to a feedback linearization-based controller, where all states and exact knowledge of the robot must be known. A Quanser flexible joint module was put to the test and has showed accurate tracking of the reference actual trajectory [22]. Moreover, the estimated uncertainties, states and integrated states and uncertainties were very similar to the actual which proves that Extended-State-Observer method is valid [22].

Currently, advanced controllers are either robust, adaptive or both. There are many uncertainties and knowledge that is not available in a control problem. Hence, robustness and adaptability

are required to solve these issues. For example, unknown and unpredictable uncertainties can be compensated with the controller proposed in [21]. Moreover, the study by Talole et al. [22] have dealt with the difficulty on obtaining the exact knowledge of the robot by prediction by using Extended-State-Observer. In the next two sections, robust and adaptive control will be discussed further.

2.3.3 Robust Control

Robot modelling errors and unmodelled dynamics are common obstacles faced by many researchers due to the complexity of the robot and external uncertainties such as friction and simplified dynamic model [23]. Additionally, neglecting modelling errors and unmodelled dynamics will result in poor performance in tracking precision [23]. Hence, robust controllers have been developed to handle these uncertainties while still preserve precision tracking. However, there is always a trade-off between robustness and performance, therefore a balance should exist between these two factors.

Robustness analysis is often done to test its performance with presence of uncertainties. For example, some of the studies mentioned previously in this literature review have proved robustness in the proposed controllers [16, 21, 22]. In this section, more robust controllers will be discussed.

One of the studies have demonstrated a simple structured, easily tuned robust controller design which can compensate disturbances and model uncertainties [24]. The controller design consist of modelled based computed torque control and robust nonlinear H-infinity control [24]. In addition, the model of the robot was obtained by using MATLAB/SimMechanics toolbox [24]. Yeon et al. [24] have used transformed dynamics for the dynamics model by substituting the elastic forces of the singular perturbation form of the link dynamics into the singular perturbation form of the motor dynamics. The experiment was implemented on a 6-DoF flexible joint robot with 30% model uncertainties of either mass or stiffness and have shown promising results with position errors less than 1.5mm [24]. It was concluded that not only the proposed controller was robust against mass uncertainties and disturbances, it is also robust against stiffness [24]. It is shown that the tracking performance can be maintained by the nonlinear H-infinity control

which provides robustness to the uncertainties.

In another example of a widely researched robust nonlinear controller is the sliding mode control. One of the properties of the sliding mode control is its insensitivity towards modelling errors, unmodelled dynamics and external disturbances, which makes this controller robust. The sliding mode control works by having two phases, where the first phase is forcing the system trajectory to achieve a sliding manifold, which is called reaching phase, and second phase is driving the states towards the desired equilibrium point, which is called sliding phase [25]. However, the widely known problem about this controller is the undesired phenomenon called 'chattering', which is known to affect control accuracy, wearing of mechanical components and heat loss in electrical components [26]. Chattering usually occurs when there is oscillation about the sliding manifold [27]. One of the causes of chattering is the discontinuous nature of the control law that contributes to the robustness of sliding mode control. Studies have also shown that the 'chattering' phenomenon is caused by the fast dynamics which are neglected in the robot model and the usage of digital controllers with finite sampling rate which leads to 'discretization chatter' [26]. Since then, many researches have been done to alleviate the chattering phenomenon by using different techniques.

In sliding mode control, there were a few reaching laws and some have proved to alleviate chattering. A reaching law can be described as a differential equation in which the dynamics of a switching function is specified [28]. Particularly, the exponential reaching law was used in many research papers [25, 28, 29, 30], which consist of a constant and proportional rate of reaching the sliding manifold. To minimize chattering, the gain for the proportional rate term is designed to be larger than the gain for the constant term, which includes a *sign* function that causes chattering. Another popular method to minimize chattering caused by the constant discontinuous term which contains a *sign* function, is to introduce a boundary layer [31]. The boundary layer replaces the sign function by a saturation function which smooths the control signal. Generally, the thicker the boundary layer, the lesser the chattering [31]. However, it comes with a cost of the sliding condition will not be guaranteed [31].

In the study of Soltanpour et al. [29], a voltage-based sliding mode control proposed utilizes both of the previously mentioned techniques. In the controller design, three sliding surfaces

are defined for each stage of the flexible joint robot manipulator to design the control law. The advantage of this approach is that it guarantees asymptotic stability for each stage of the robot and global asymptotic stability of the closed-loop system in the presence of structured and unstructured uncertainties [29]. The experiment was implemented on a single DoF flexible joint robot with structure and unstructured uncertainties [29]. Physical shock was also applied by trying to change the path of the robot arm with the hand. It was concluded that the experimental results had achieved good tracking performance despite the uncertainties and physical shock [29]. There was also no chattering in the simulation and experimental results [29].

Modification can also be done in the sliding manifold. In this study, an integral term was added to the sliding mode control to overcome the chattering problem and improve robustness and performance of the system [25]. The benefit of the integral sliding mode control is that the reaching phase is not needed as the system trajectory always starts from the sliding manifold and the integral action drives the states to the desired equilibrium point [25]. In the simulation and experimental results, both controllers were observed to achieve good performance in terms of settling time and zero steady state. One of the drawbacks was the presence of the chattering phenomenon in the sliding mode controller, but, the integral sliding mode controller showed great improvements compared to the sliding mode controller [25]. However, the case where mismatch perturbations are being subjected into the system was not studied. Therefore, in a recent paper by Alam et al. [30], a comparison between the traditional sliding mode control, integral sliding mode control and the disturbance observer-based sliding mode control was being investigated when mismatch perturbations were subjected. The disturbance observer-based sliding mode controller that Alam et al. [30] proposed deal with mismatch perturbations by estimating the instantaneous mismatched uncertainties and updates the parameters in the control law. This method would guarantee robustness and nominal control performance is secured. Figure 2.1 summarizes the performance of each controller in the simulation done by Alam et al. [30] when the disturbance was inserted. From the figure, it was obvious that traditional sliding mode control did not react well to the mismatched disturbance and hence resulted in a constant steady-state error. Besides that, the integral version has a high percentage overshoot and longer settling time. This research concluded that the disturbance observer-based sliding mode control outperforms the other controllers in all three aspects, steady-state error, settling

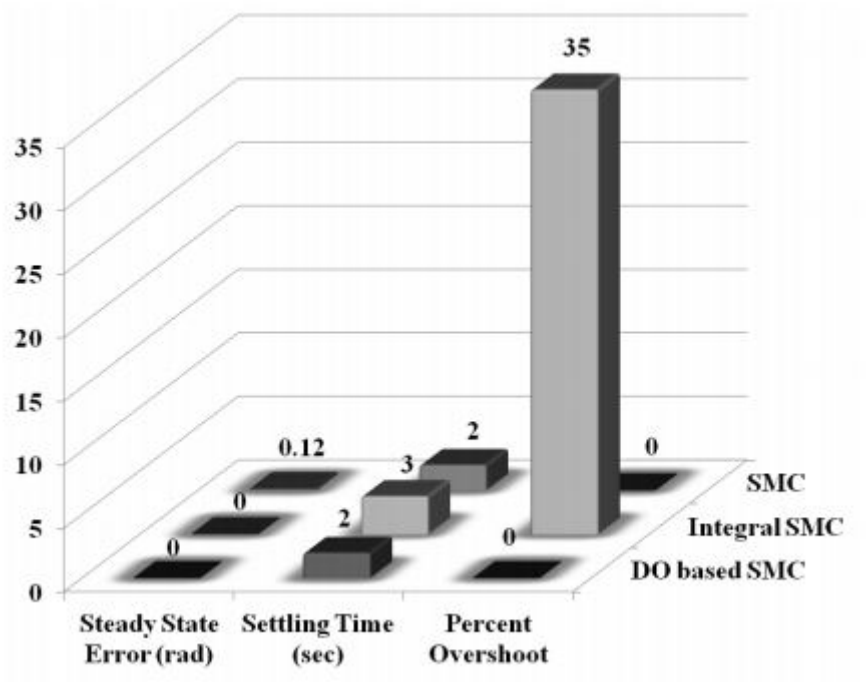

Figure 2.1: Performance comparison between all control strategies [30]

time and percentage overshoot [30]. It was also mentioned that the estimated disturbance approaches the actual disturbance and the trajectory converges to the desired equilibrium point asymptotically [30]. Needless to say, chattering problem was also improved. On the other hand, both of the studies [25, 30] utilized voltage as its input as it takes actuator dynamics into account.

2.3.4 Adaptive Control

Adaptive controllers are controllers that can perform regulation and tracking problems while adapting to the unknown or varying process parameters. It is useful when there are no knowledge or varying system parameters such as unpredictable loads and flexibilities. However, complexity and computational time are the drawbacks for this type of controller.

Model Reference Adaptive Control (MRAC) is a type of adaptive control scheme which has been studied for some time. In Ulrich's [10] study, a comparison between an adaptive and

a nonadaptive control scheme have been investigated. The adaptive and nonadaptive control were the MRAC and Transpose Jacobian control strategy, respectively. The general idea of the MRAC is to reduce tracking error between actual and the desired system outputs specified in a reference model from the robot system by varying control gains in real-time to the decentralized modified simple adaptive control law [10]. Results have indicated that MRAC showed good adaptivity, even against complex nonlinear dynamics model which includes uncertainties like nonlinear stiffness, soft-windup and friction [10]. In addition, MRAC have performed better in tracking a square trajectory in a given timeframe than the Transpose Jacobian method with a faster settling time to steady state and lesser overshoot at the corners of the square [10].

Another approach of an adaptive controller was by utilizing a neural-network observer to obtain unmeasurable parameters which was studied in a recent research by Liu et al. [32]. The study has proposed an observer based on Radial Basis Function neural network to estimate system states [28]. The controller is then designed according to the dynamic surface method on a single link flexible joint robot where the initially unknown model is constructed by Radial Basis Function neural network [32]. Moreover, the stability of the proposed controller and observer were established by using the Lyapunov method. Furthermore, the simulation have achieved good performance despite the absence of the model [32]. This proves that controllers can still be designed even without the knowledge of the robot model, however, it is unsure whether it is capable to handle multiple link robot with greater complexity.

2.4 Summary and Research Gaps

Overall, the presence of uncertainties is common in any system. Joint flexibilities can contribute to undesirable vibrations which could lead to unstable system or even system crash. Furthermore, flexible joint robot model can be simplified to a linear model where flexibility is modelled as a linear torsional spring. Nonlinearities can also be considered in the dynamics model, but it is known to have greater control challenges but better performance. Moreover, there are two control objectives, regulation, and tracking control. The tracking control task is known to be more complex than the regulation task. Different types of controllers are used to perform different control objectives and operating conditions. However, the controller design plays a greater

role than the type of controllers as combination of different controllers and techniques could be altered to overcome certain problems. For example, by adding an observer to predict system states, a controller could become more robust towards uncertainties. Additionally, advanced control algorithm can be either robust, adaptive or both. Robust controllers are useful when there is presence of uncertainties in the system while adaptive controllers are used when system parameters are unknown. Many of these controllers have utilized more complex techniques such as neural network [32] and prediction of state variables [30] to compensate greater uncertainties. On the other hand, some researchers suggest using voltage as the input, as it includes the actuator dynamics for better results, rather than torque input when the electromechanical system is significant.

In this literature review, a variety of controllers were reviewed to better understand the studies have been done in the robotics field of flexible joint manipulators and the purpose of each controllers were also investigated. One of the research gap is that most of the recently published papers that were reviewed have only tested the controllers in simulation [10, 17, 30, 32]. Additionally, most of the recent studies have applied advanced techniques in the controller design, which this thesis is particularly interested. Therefore, experimental results might deviate from the simulation results due to unexpected factors in the working environment that were not being considered in the simulation. Hence, in this thesis, the design of the controllers is aimed to be implemented on a 2 DoF robot manipular with flexible joints with real-world setting. Another research gap is that most of the flexible joint robots that were tested in the studies have very high joint stiffness [12, 14, 15, 16, 20, 21]. For example, in [12, 14, 20], the joint stiffness was over 1000 Nm/rad. However, this thesis will be investigating robot manipulator with highly flexible joints where the joint stiffness will be less than 10 Nm/rad. Although most of the research paper mentioned only tested the controllers for flexible joint robots, but, in this thesis, the controllers will also be tested on rigid joint robots to compare the difference.

Chapter 3

Methodology

3.1 Overview

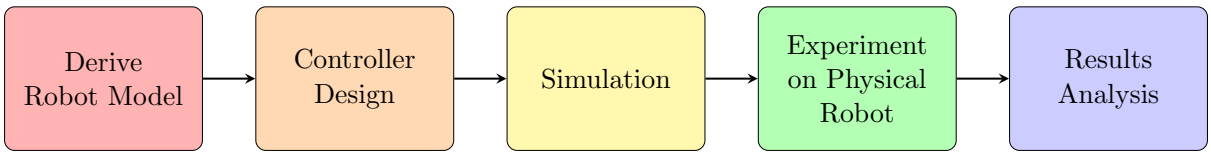

Figure 3.1: Overall methodology in conducting the experiment

Although modelling and control of a robot arm manipulator is a well-established field, and advancements have been made in both the model of the robot arm manipulator and the controller design, there is still some research gaps that this thesis aims to achieve. The objective of this thesis is to conduct experiments on a 2 DoF serial highly flexible joint and rigid joint robot arm manipulator to highlight the effects and significance of flexibility in robot joints. Besides that, as the control problem and the dynamics of the robot became more complex, numerous studies have been done to solve the arising problems due to complexities and uncertainties by applying advanced control algorithms. Unfortunately, experimental validation was not provided from recent research papers which studied advanced control algorithms. In this thesis, the controller designs from past literature will be studied, modified and applied to the Quanser 2 DoF Serial Robot. Simulations and experiments are conducted to explore the behaviour and simultaneously verifying the results of the previous work. Different advanced control algorithm will also

be compared while the robot manipulator performs the same task. The overall approach in conducting the whole experiment is shown in the flow chart in Figure 3.1.

3.2 Robot Manipulator Model

There are two types of dynamic robot models that are used in this thesis, which are the rigid joint and flexible joint robot dynamic model. The dynamic model is needed in the simulation to simulate the dynamics of the actual robot and also required in controller design.

3.2.1 Rigid Joint Robot Dynamic Model

The dynamic model of a rigid joint robot is obtained by using the Euler-Lagrange method, which is an energy-based approach of deriving the dynamics of a system. Firstly, the total kinetic energy, K, which includes rotational and translational kinetic energy, and potential energy, P is calculated. The Lagrangian, L can then be calculated by using equation (3.1), where L and K depend on the angular position, q and angular velocity, $\dot{q}$, while P only depends on the angular position, q. However, since it is a planar robot, there will be no gravity acting on the system, and hence, the potential energy is zero. The Langrangian is used to obtain the equation of motion shown in equation (3.2), in which $i = (1, ..., n)$ where n is the number of links and i is the i_{th} link.

$$L(q, \dot{q}) = K(q, \dot{q}) - P(q) \tag{3.1}$$

$$\tau_i = \frac{d}{\mathrm{dt}} \frac{\partial}{\partial \dot{q}_i} L - \frac{\partial}{\partial q_i} L \tag{3.2}$$

It is then rearranged to acquire the form of equation (3.3).

$$M(q)\,\ddot{q} + C(q, \dot{q})\,\dot{q} + f_v \dot{q} = \tau \tag{3.3}$$

In which $q = (q_1, ..., q_n)^T$ and $\tau = (\tau_1, ..., \tau_n)^T$ are both $(n \times 1)$ matrix, where q is the angular position and τ is the input torque. M, C, and f_v are the inertia matrix, Coriolis and centrifugal matrix and a diagonal matrix of viscous damping coefficient respectively where all matrices are

$(n \times n)$ matrices. Note that the actual robot utilises current as input, therefore, $\tau_i = K_{m_i} I_{m_i}$, where K_{m_i} and I_{m_i} is the torque constant and the input current of link i respectively. The torque constant converts current to torque.

In this thesis, a planar robot manipulator with two links is used. The step by step calculations of the Euler-Lagrange method is done in MATLAB which can be found in Appendix A. The M, C and f_v matrices are defined in equation (3.4, 3.5, 3.6), and are all (2×2) matrices. The description and values of the model parameters are shown in Table 3.1, where the values are obtained from the Quanser robot user manual [33] and provided by the School of Electrical Engineering and Telecommunications at the University of New South Wales.

$$M = \begin{bmatrix} m_2\, {L_1}^2 + 2\, b\, m_1\, {\mathrm{Lc}_1}^2 \cos(q_2) + a + I_1 & a + b\cos(q_2) \\ a + b\cos(q_2) & a \end{bmatrix} \tag{3.4}$$

$$\begin{aligned} a =& m_2\, {\mathrm{Lc}_2}^2 + I_2 \\ b =& m_2\, L_1\, \mathrm{Lc}_2 \end{aligned}$$

$$C = \begin{bmatrix} -L_1\, \mathrm{Lc}_2\, m_2\, \sin(q_2)\, \dot{q_2} & -L_1\, \mathrm{Lc}_2\, m_2\, \sin(q_2)\, (\dot{q_1} + \dot{q_2}) \\ L_1\, \mathrm{Lc}_2\, m_2\, \sin(q_2)\, \dot{q_1} & 0 \end{bmatrix} \tag{3.5}$$

$$f_v = \begin{bmatrix} B1 & 0 \\ 0 & B2 \end{bmatrix} \tag{3.6}$$

Parameters	**Values**	**Units**
Torque constant for joint 1, K_{m_1}	8.925	Nm/A
Torque constant for joint 2, K_{m_2}	0.87	Nm/A
Length of link 1, L_1	0.343	m
Length of link 2, L_2	0.267	m
Distance to centre of gravity of link 1, Lc_1	0.1590	m
Distance to centre of gravity of link 2, Lc_2	0.0550	m
Mass of link 1, m_1	1.51	kg
Mass of link 2, m_2	0.87326	kg
Moment of inertia at z axis of link 1, I_1	0.0392	kgm^2
Moment of inertia at z axis of link 2, I_2	0.0081	kgm^2
Viscous damping coefficient of link 1, B_1	0.070364	Nms/rad
Viscous damping coefficient of link 2, B_2	0.028211	Nms/rad

Table 3.1: Parameters Values for Quanser 2 DoF Rigid Joint Robot

3.2.2 Flexible Joint Robot Dynamic Model

Similar to the rigid joint robot dynamic model, the Euler-Lagrange method is used to derive the model. Since the joints are flexible, the potential energy includes the elastic potential energy of the springs. The equation of motion of the flexible joint robot can be expressed in the form of equation (3.7).

$$\begin{aligned} D(q)\,\ddot{q} + C(q,\dot{q})\,\dot{q} + K(q-\phi) + f_{v1}\dot{q} &= 0 \\ J\ddot{\phi} + f_{v2}\dot{\phi} + K(\phi - q) &= \tau \end{aligned} \tag{3.7}$$

As shown in equation (3.5), the presence of flexibility introduced additional matrices and state variable, ϕ. Where $q = (q_1, ..., q_n)^T$ and $\phi = (\phi_1, ..., \phi_n)^T$ are link angular position and motor angular position respectively and n is the number of joints in the robot. D is the inertia matrix of the n rigid links, and J is a diagonal matrix of the inertia of the motors, where both are $(n \times n)$ matrices. C, K, f_{v1} and f_{v2} are the Coriolis and centrifugal matrix, a diagonal matrix of joint stiffness coefficient, and diagonal matrices of viscous damping coefficient respectively, and are also $(n \times n)$ matrices. As mentioned previously, $\tau_i = K_{m_i} I_{m_i}$, which converts current to torque.

With the calculations done in MATLAB, the D, J, C, K, f_{v1} and f_{v2} matrices can be defined as shown below. Since the robot used is a two linked planar flexible joint robot manipulator, n is 2. Besides that, the D, C and f_{v1} matrices are the same as the M, C, and f_v matrices defined in the rigid joint model. The values of model parameters are showed in Table 3.2 and obtained from the same sources from the case of the rigid joint robot.

$$D = \begin{bmatrix} m_2\,{L_1}^2 + 2\,b\,m_1\,{\mathrm{Lc}_1}^2\cos(q_2) + a + I_1 & a + b\cos(q_2) \\ a + b\cos(q_2) & a \end{bmatrix} \tag{3.8}$$

$$\begin{aligned} a &= m_2\,{\mathrm{Lc}_2}^2 + I_2 \\ b &= m_2\,L_1\,\mathrm{Lc}_2 \end{aligned}$$

$$J = \begin{bmatrix} Ir_1 & 0 \\ 0 & Ir_2 \end{bmatrix} \tag{3.9}$$

$$C = \begin{bmatrix} -L_1 \, \mathrm{Lc}_2 \, m_2 \, \sin(q_2) \, \dot{q_2} & -L_1 \, \mathrm{Lc}_2 \, m_2 \, \sin(q_2) \, (\dot{q_1} + \dot{q_2}) \\ L_1 \, \mathrm{Lc}_2 \, m_2 \, \sin(q_2) \, \dot{q_1} & 0 \end{bmatrix} \tag{3.10}$$

$$K = \begin{bmatrix} K_{s1} & 0 \\ 0 & K_{s2} \end{bmatrix} \tag{3.11}$$

$$f_{v1} = \begin{bmatrix} B1 & 0 \\ 0 & B2 \end{bmatrix} \tag{3.12}$$

$$f_{v2} = \begin{bmatrix} Bm1 & 0 \\ 0 & Bm2 \end{bmatrix} \tag{3.13}$$

Parameters	**Values**	**Units**
Torque constant for joint 1, K_{m_1}	8.925	Nm/A
Torque constant for joint 2, K_{m_2}	0.87	Nm/A
Length of link 1, L_1	0.343	m
Length of link 2, L_2	0.267	m
Distance to centre of gravity of link 1, Lc_1	0.1590	m
Distance to centre of gravity of link 2, Lc_2	0.0550	m
Mass of link 1, m_1	1.51	kg
Mass of link 2, m_2	0.87326	kg
Moment of inertia at z axis of link 1, I_1	0.0392	kgm^2
Moment of inertia at z axis of link 2, I_2	0.0081	kgm^2
Moment of inertia at z axis of motor 1, Ir_1	0.0628	kgm^2
Moment of inertia at z axis of motor 2, Ir_2	0.0026	kgm^2
Viscous damping coefficient of link 1, B_1	0.070364	Nms/rad
Viscous damping coefficient of link 2, B_2	0.028211	Nms/rad
Viscous damping coefficient of motor 1, Bm_1	4.5	Nms/rad
Viscous damping coefficient of motor 2, Bm_2	0.5	Nms/rad
Torsional stiffness of spring 1, Ks_1	9.0	Nm/rad
Torsional stiffness of spring 2, Ks_2	4.0	Nm/rad

Table 3.2: Parameters Values for Quanser 2 DoF Flexible Joint Robot

3.3 Controller Design

In this thesis, the controllers designed will be tested on either a regulation and tracking problem in joint space on a 2 DoF rigid and flexible joint robot manipulator. Joint space is the space which a vector of joint variables of each joint in the robot is defined [19]. A total of four controllers were prepared, which are the Proportional-Derivative (PD) controller, Adaptive PD controller, Conventional Sliding Mode Controller (SMC) made for rigid joint robot manipulator, and a Modified SMC for the flexible joint robot manipulator. The PD controllers are designed to tackle the regulation problem, while the SMC controllers are designed to overcome the tracking problem. The derivation and calculations of the controllers are all done in MATLAB. Additionally, the simulation of all the controllers is done in MATLAB/Simulink. The block diagram of the simulation for each controller is located in Appendix A.

3.3.1 PD Controller

The Proportional-Integral-Derivative (PID) type controllers are commonly used in the industry and have a simple control structure. A simple PD controller was designed for both rigid and flexible joint robot manipulator. The PD controller is not only robust against model uncertainties but also do not require any knowledge of the model structure and parameters [19]. The PD controller can be shown in equation (3.14) and are the same for the rigid and flexible joint robot manipulator where the only difference is the joint variables.

$$u = K_P(q_d - q) - K_D\dot{q} \tag{3.14}$$

In which u is the (2×1) vector of control input current, K_P and K_D are the proportional and derivative gain respectively and are a constant, (2×2) positive definite diagonal matrices. q_d and q is a (2×1) vector of desired and actual joint variables. However, for the flexible joint robot, the motor joint variables are used as it is used in the stability analysis.

Besides that, the PD controller is proved to be globally asymptotically stable for both rigid and flexible joint robot manipulators by applying Laselle's theorem as long as the K_P and K_D are positive definite [11, 19]. However, in some cases, K_P has to be large enough, but measurement

noise and unmodelled dynamics will limit the value of the gains; hence, steady-state error can occur [19].

3.3.2 Adaptive PD

Since the actual robot manipulator has many uncertainties like a non-constant dead zone in the motors, the PD controller from the previous section can be improved by adding an adaptive mechanism. The adaptive ability was acquired by introducing the Model Reference Adaptive Control (MRAC) strategy using the modified MIT rule inspired by Jain et al. [34] into the control strategy. A brief explanation of how the Modified MIT Rule works is described below.

3.3.2.1 Modified MIT Rule

Generally, in the MIT rule, a cost function is defined as shown in equation (3.15), where e is the error which is the actual plant output minus the reference model output shown in equation (3.16) and θ is the adjustable parameter that is adapted in the controller.

$$J(\theta) = \frac{1}{2}e^2 \tag{3.15}$$

$$e = y - y_m \tag{3.16}$$

Since the aim is to minimize the cost function which corresponds to the error, the change of θ shown in equation (3.17) will be the negative gradient of J [34].

$$\frac{d\theta}{dt} = -\gamma\frac{\partial J}{\partial \theta} = -\gamma\, e\frac{\partial e}{\partial \theta} \tag{3.17}$$

where γ is the positive definite adaptation gain and $\frac{\partial e}{\partial \theta}$ is the sensitivity derivative which describes the changes of the error with respect to θ. In the Modified MIT rule, the Normalized algorithm is used with the MIT rule, which results in equation (3.18). The Normalized algorithm is introduced to increase stability as the MIT rule is very sensitive to changes in reference input [34]. In order to visualize how the Modified MIT rule works, a block diagram is illustrated in Figure 3.1.

$$\frac{d\theta}{dt} = -\frac{\gamma e\phi}{\alpha + \phi'\phi} \tag{3.18}$$

$$\phi = \frac{\partial e}{\partial \theta}, \quad \alpha > 0 \quad \text{(small constant value to avoid zero division)}$$

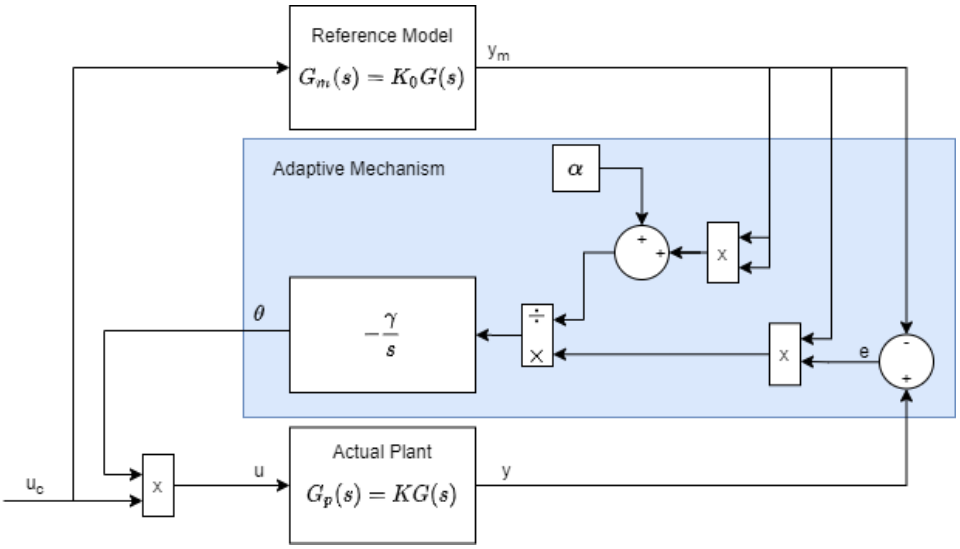

Figure 3.2: Block Diagram of Modified MIT Rule

From Figure 3.1 and equation (3.16), the equation below can be derived.

$$\begin{aligned} E(s) =& KG(s)U(s) - K_0 G(s)U_c(s) \\ =& KG(s)\theta U_c(s) - K_0 G(s)U_c(s) \end{aligned} \tag{3.19}$$

Where K is an unknown constant parameter and K_0 is a known constant parameter. With equation (3.19), the sensitivity derivative in terms of the model output can be obtained by taking the partial derivative of θ.

$$\begin{aligned} \frac{\partial e}{\partial \theta} =& KGU_c \\ =& \frac{K}{K_o} y_m \end{aligned} \tag{3.20}$$

By substituting equation (3.20) into (3.18), it results in the law for adjusting parameter θ.

$$\frac{d\theta}{dt} = -\frac{\gamma e \left(\frac{K}{K_o} y_m\right)}{\alpha + \left(\frac{K}{K_o} y_m\right)' \left(\frac{K}{K_o} y_m\right)} \tag{3.21}$$

3.3.2.2 Proposed Controller Design

In addition to the MRAC with Modified MIT rule, dead-zone inverse is also introduced. Since the dead-zone is non-constant in this case, the dead-zone inverse alone will not provide a satis-

factory result. Hence, an adaptive mechanism is included to adapt to the changing dead-zone and other unmodelled dynamics. The adaptive PD controller is shown in equation (3.22).

$$\begin{aligned} u =& v\theta \\ v =& DZ^1(u_{pd}) = \begin{cases} u_{pd} + b_r & , u_{pd} > 0 \\ 0 & , u_{pd} = 0 \\ u_{pd} - b_l & , u_{pd} < 0 \end{cases} \end{aligned} \tag{3.22}$$

where u_{pd} is the PD controller described in equation (3.14), b_r and b_l are the approximated start and end of the dead-zone parameters. These values can be approximated as θ, the adjustable parameter from the adaptive mechanism will adjust itself to minimize the error between the actual and reference model. In terms of stability, the MIT rule does not guarantee convergence or stability of the state variables [35]. Although stability is not guaranteed, experimental results in Chapter 4 showed there exist values of control parameters that lead to stability in the state variables. The Simulink block diagram of the proposed adaptive PD controller in equation (3.22) can be shown in Figure 3.4.

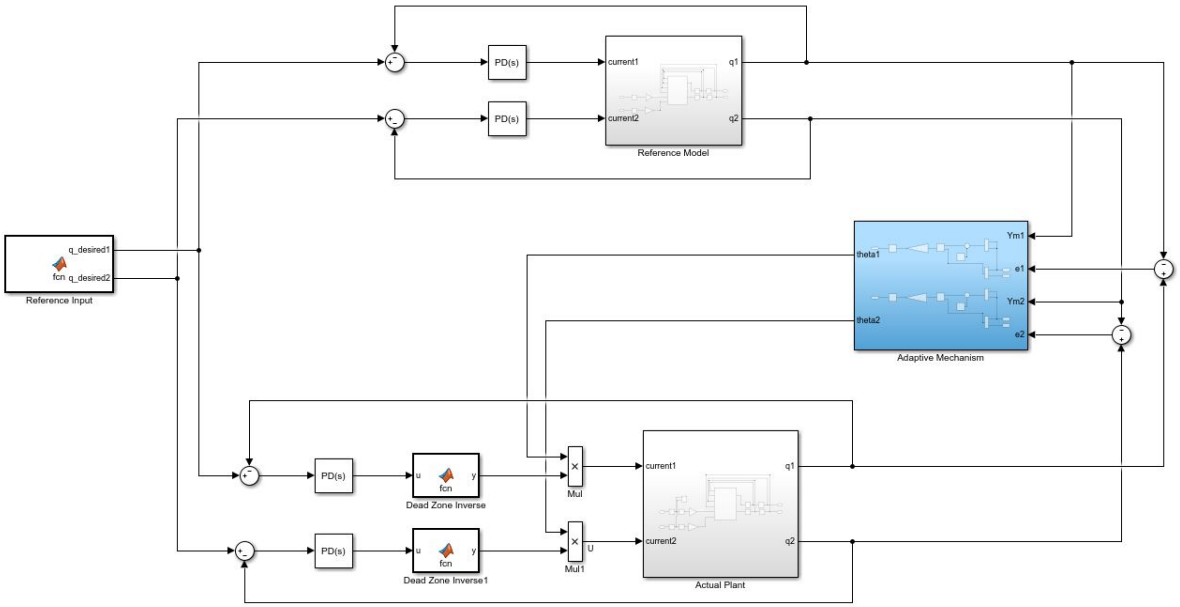

Figure 3.3: Simulink Block Diagram of Adaptive PD Controller

3.3.3 Conventional Sliding Mode Controller

A robust conventional SMC controller is design for the rigid joint robot manipulator to overcome the tracking problem. The SMC controller designed is based on the exponential reaching law, inspired by Moldoveanu [28] and Liu [27] mentioned previously in the literature review. The sliding mode function, $s = (s_1, s_2)^T$, used in this controller can be shown in equation (3.23).

$$\begin{aligned} s =& ce + \dot{e} \\ e =& q - q_d \\ \dot{e} =& \dot{q} - \dot{q_d} \end{aligned} \tag{3.23}$$

where c is a (2×2) diagonal matrix of positive constants, $q = (q_1, q_2)^T$ and $q_d = (q_{1d}, q_{2d})^T$ are the matrices of angular position and desired angular position of the two links. e and $\dot{e}$ are the (2×1) matrix of tracking error of the angular position and velocity respectively. By taking the time derivative of equation (3.23) and substituting the rigid robot dynamics (3.3) with bounded unknown disturbance, $|d(t)| \leq D$ into the equation, it results in the equation below.

$$\begin{aligned} \dot{s} =& c\dot{e} + \ddot{e} \\ =& c\dot{e} + \ddot{q} - \ddot{q_d} \\ =& c\dot{e} + M(q)^{-1}(\tau - C(q, \dot{q})\dot{q} - f_v\dot{q}) + d - \ddot{q_d} \end{aligned} \tag{3.24}$$

The exponential reaching law can be described as,

$$\dot{s} = -K_1 sign(s) - K_2 s \tag{3.25}$$

In which K_1 and K_2 are (2×2) diagonal matrices of the positive constants for the constant rate terms and the exponential term respectively. Furthermore, $sign(s) = (sign(s_1), sign(s_2))^T$ and $s = (s_1, s_2)^T$. In the exponential reaching law, the proportional rate term, $-K_2 s$, will force the state to approach the switching manifold faster and the $-K_1 sign(s)$ term is used when s is near zero. Since the main contributor to the chattering phenomenon is the $-K_1 sign(s)$ term, by choosing a smaller K_1 and larger K_2 will alleviate chattering. With equation (3.24) and (3.25) while assuming that there is no disturbance, the control law can be defined as,

$$\tau = M(q)[-K_1 sign(s) - K_2 s - c\dot{e} + \ddot{q_d}] + C(q, \dot{q})\dot{q} + fv\dot{q} \tag{3.26}$$

As for the stability analysis, a Lyapunov function, V, can be defined as $V = \frac{1}{2}s^2$, which is positive definite. When taking the time derivative of V,

$$\begin{aligned}\dot{V} =& s\dot{s} \\ =& s(-K_1 sign(s) - K_2 s + d) \\ =& -K_1|s| - K_2 s^2 + Ds \leq 0, \quad \text{if } K_1 \geq D, K_2 > 0\end{aligned} \tag{3.27}$$

By applying LaSalle's Theorem, $\dot{V} = 0$ and hence $s = 0$. This means that, the system will converge to the set M, where $M = \{s = 0\}$ as time approaches infinity. Moreover, when $s = ce + \dot{e} = 0$, it is shown that $e(t) = e(0)exp(-ct)$, which means as time approaches infinity, the position and velocity tracking error will tend to zero exponentially with the value c [27].

In order to minimize the chattering that is mainly caused by the sign function in the $-K_1 sign(s)$ term, the boundary layer method described in literature review was applied. The sign function is modified to a saturation function, $sat(s)$, which can be defined in equation (3.28) [31]. ψ is a small constant and characterizes the error associated with smooth approximation of the $sign(s)$.

$$sat(s) = \begin{cases} 1 & , \psi > 0 \\ \frac{s}{\psi} & , -\psi < s \leq \psi \\ -1 & , s \leq -\psi \end{cases} \tag{3.28}$$

3.3.4 Modified Sliding Mode Controller

Since the conventional SMC is designed for rigid joint robot manipulators, only the link variables are taken into consideration. In this modified SMC designed for flexible joint robot manipulators, both motor and link variables are included in the control law. The proposed controller is inspired by the work of Soltanpour et al. [29], which is previously mentioned in the literature review. However, in this thesis, instead of voltage-based control input, torque is used. This controller was also not tested on a physical 2 DoF flexible joint robot, which is a good opportunity to verify this controller with increased complexity.

The approach of the proposed controller utilises three sliding surfaces for each stage of the robot manipulator. Moreover, similar to the conventional SMC, the robot dynamics is also used to

formulate the control input. But, the robot dynamics depends the control input torques, u, the motor variables, ϕ and the link deflections, α instead of the link variables, which can be described in equation (3.29). For conciseness, the actual and desired state variables is defined as $x = (\phi_1, \phi_2, \alpha_1, \alpha_2, \dot{\phi_1}, \dot{\phi_2}, \dot{\alpha_1}, \dot{\alpha_2})^T$ and $x_d = (\phi_{d_1}, \phi_{d_2}, \alpha_{d_1}, \alpha_{d_2}, \dot{\phi_{d_1}}, \dot{\phi_{d_2}}, \dot{\alpha_{d_1}}, \dot{\alpha_{d_2}})^T$ respectively.

$$\begin{aligned} \alpha =& q - \phi \\ \ddot{x_i} = \ddot{\phi_i} =& f_i(\phi, \alpha, \dot{\phi}, \dot{\alpha}) + b_i u_i + d_i(t) \\ \ddot{x}_{n+i} = \ddot{\alpha_i} =& f_{i+n}(\phi, \alpha, \dot{\phi}, \dot{\alpha}) + b_{i+n} u_i + d_{n+i}(t) \end{aligned} \tag{3.29}$$

In which $i = 1, 2$ and $n = 2$, which is the number of subsystems. In addition, $f_i(\phi, \alpha, \dot{\phi}, \dot{\alpha})$ and $f_{i+n}(\phi, \alpha, \dot{\phi}, \dot{\alpha})$ are nonlinear functions representing the flexible joint robot dynamics, b_i and b_{i+n} are coefficients for the control input, u, and d_i and d_{n+i} are bounded disturbance, where $|d_i| \leq D_i$ and $|d_{n+i}| \leq D_{n+i}$. The three sliding surfaces is shown in equation (3.30), where e_i is the error of each state variable.

$$\begin{aligned} s_{1i} =& c_{1i} e_i + e_{2n+i}, \quad i = 1, 2 \quad n = 2 \\ s_{2i} =& c_{2i} e_{n+i} + s_{1i} \\ s_{3i} =& c_{3i} e_{3n+i} + s_{2i} \\ e_j =& x_j - x_{d_j}, \quad j = 1, 2.., 8 \end{aligned} \tag{3.30}$$

It is important to note that the third sliding surface contains all the state variables of the corresponding subsystem. Hence, the condition for stability is the convergence to zero for the third sliding surface [29]. By taking the time derivative of the third sliding surface, s_{3i},

$$\begin{aligned} \dot{s}_{3i} =& \; c_{3i} \dot{e}_{3n+i} + \dot{s}_{2i} \\ =& \; c_{3i}(\ddot{x}_{n+i} - \ddot{x}_{d_{n+i}}) + c_{2i}(\dot{x}_{n+i} - \dot{x}_{d_{n+i}}) + c_{1i}(\dot{x}_i - \dot{x}_{d_i}) + (\ddot{x}_i - \ddot{x}_{d_i}) \\ =& \; c_{3i}(f_{i+n}(\phi, \alpha, \dot{\phi}, \dot{\alpha}) + b_{i+n} u_i + d_{n+i} - \ddot{x}_{d_{n+i}}) + c_{2i}(\dot{x}_{n+i} - \dot{x}_{d_{n+i}}) + c_{1i}(\dot{x}_i - \dot{x}_{d_i}) \\ & + (f_i(\phi, \alpha, \dot{\phi}, \dot{\alpha}) + b_i u_i + d_i - \ddot{x}_{d_i}) \end{aligned} \tag{3.31}$$

With the adoption of the exponential reaching law,

$$\dot{s}_{3i} = \; -\eta_i sign(s_{3i}) - K_i s_{3i} \tag{3.32}$$

To obtain the total control law, u_i, equate equation (3.31) and (3.32), while assuming that there

is no presence of disturbance.

$$\begin{aligned} u_i &= u_{eq} + u_{sw} \\ u_{eq} &= -\frac{c_{3i}(f_{i+n}(\phi,\alpha,\dot{\phi},\dot{\alpha}) - \ddot{x}_{d_{n+i}}) + c_{2i}(\dot{x}_{n+i} - \dot{x}_{d_{n+i}}) + c_{1i}(\dot{x}_i - \dot{x}_{d_i}) + f_i(\phi,\alpha,\dot{\phi},\dot{\alpha}) - \ddot{x}_{d_i}}{c_{3i}b_{i+n} + b_i} \\ u_{sw} &= -\frac{\eta_i sign(s_{3i}) + K_i s_{3i}}{c_{3i}b_{i+n} + b_i} \end{aligned} \tag{3.33}$$

In which u_{eq} represents the equivalent control law where it is assumed that it is free of external disturbances or unmodelled dynamics, and u_{sw} is the control law that handles the presence of uncertainties in the system with the condition where η_i and K_i must be positive constants.

In order to examine the stability of this controller, the Lyapunov function for the third sliding surface can be defined as $V_{3i} = \frac{1}{2}S_{3i}^2$. After taking the time derivative of V_{3i}, equation (3.31) is first substituted into the equation followed by (3.33), which yields (3.34).

$$\begin{aligned} \dot{V_{3i}} &= S_{3i}\dot{S}_{3i} \\ &= S_{3i}(C_{3i}d_{n+i}(t) + d_i(t) - \eta_i sign(S_{3i}) - K_i S_{3i}) \end{aligned} \tag{3.34}$$

Since the disturbances are bounded, the equation (3.34) can be represented as,

$$\begin{aligned} \dot{V_{3i}} &\leq S_{3i}(C_{3i}Dn + i + D_i - \eta_i sign(S_{3i}) - K_i S_{3i}) \\ &\leq (C_{3i}D_{n+i} + D_i)|S_{3i}| - \eta_i|S_{3i}| - K_i S_{3i}^2) \end{aligned} \tag{3.35}$$

By selecting $\eta_i > (C_{3i}D_{n+i} + D_i)$, it can be concluded that $\dot{V}_{3i} \leq 0$. Based on the LaSalle's Theorem, when $\dot{V}_{3i} = 0$, S_{3i} will equate to zero, which means that the system will converge to set M, where S_{3i} will approach zero as time approaches infinity. Hence, it is proved that in the presence of disturbances, the control law in equation (3.33) converges the third sliding surface which consist of all the state variables to zero. In the detailed stability analysis done by Soltanpour et al. [29], the closed-loop system was proved to achieve global asymptotic stability in the presence of uncertainties and all sliding surfaces and state variables will converge to zero at a finite time.

In addition, the boundary layer method mentioned in the Conventional SMC was also applied in this controller to reduce chattering.

3.4 Experiment on Physical Robot

The controllers are tested on the Quanser 2 DoF Serial Flexible Joint Robot located in the System and Control Lab(109) in the School of Electrical Engineering and Telecommunications at the University of New South Wales. To conduct the experiment on the robot, the MATLAB Simulink Desktop Real-Time Toolbox allows running Simulink block diagrams which are connected to the physical system and allows collecting data in real-time. A Simulink template that consists of real-time Simulink blocks connected to the robot terminal was provided by the School of Electrical Engineering and Telecommunications.

Since the Quanser robot takes in current, I_{m_i} as input, the torque constant, K_{m_i} is used to convert torque to current for torque-based controllers like the Conventional SMC and Modified SMC.

3.4.1 Robot Manipulator Configuration

The original configuration of the robot manipulator was mounted with springs in each joint showed in Figure 3.4. However, the rigid joint configuration can be represented by mounting rigid bars on the inner holes of the opposed support bars of the flexible joint illustrated in Figure 3.5. Additionally, the detailed description of the Quanser robot can be found in [33].

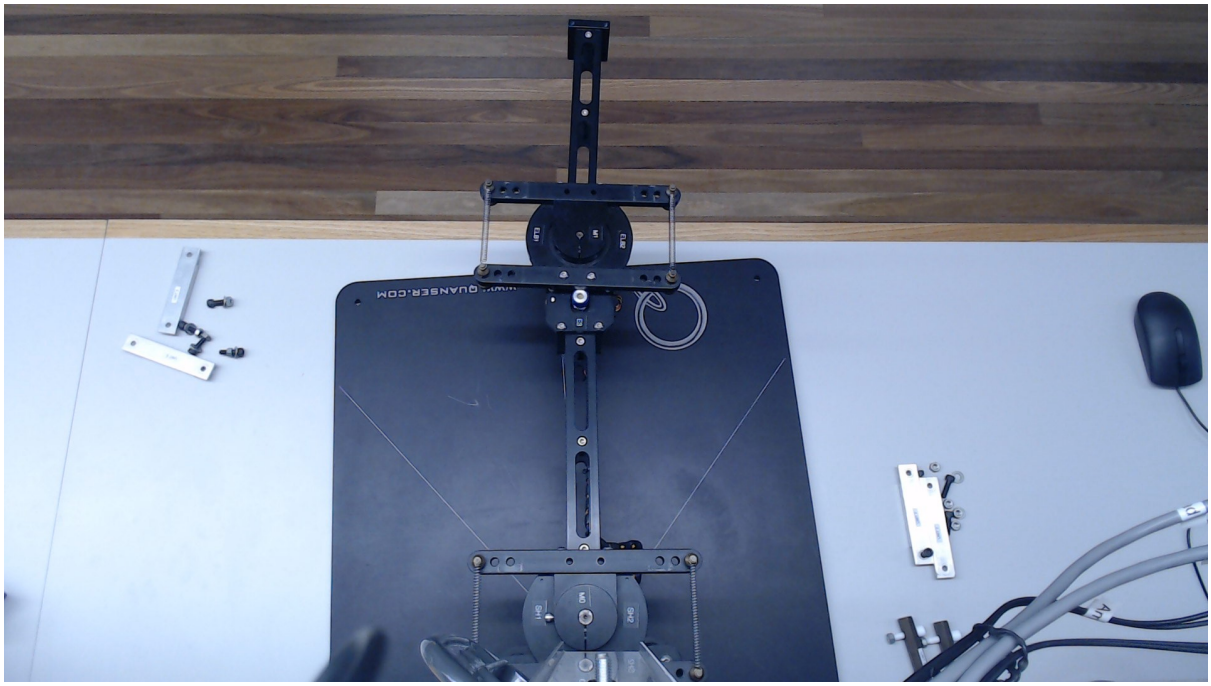

Figure 3.4: Flexible Joint Configuration of Quanser 2 DoF Serial Link Robot

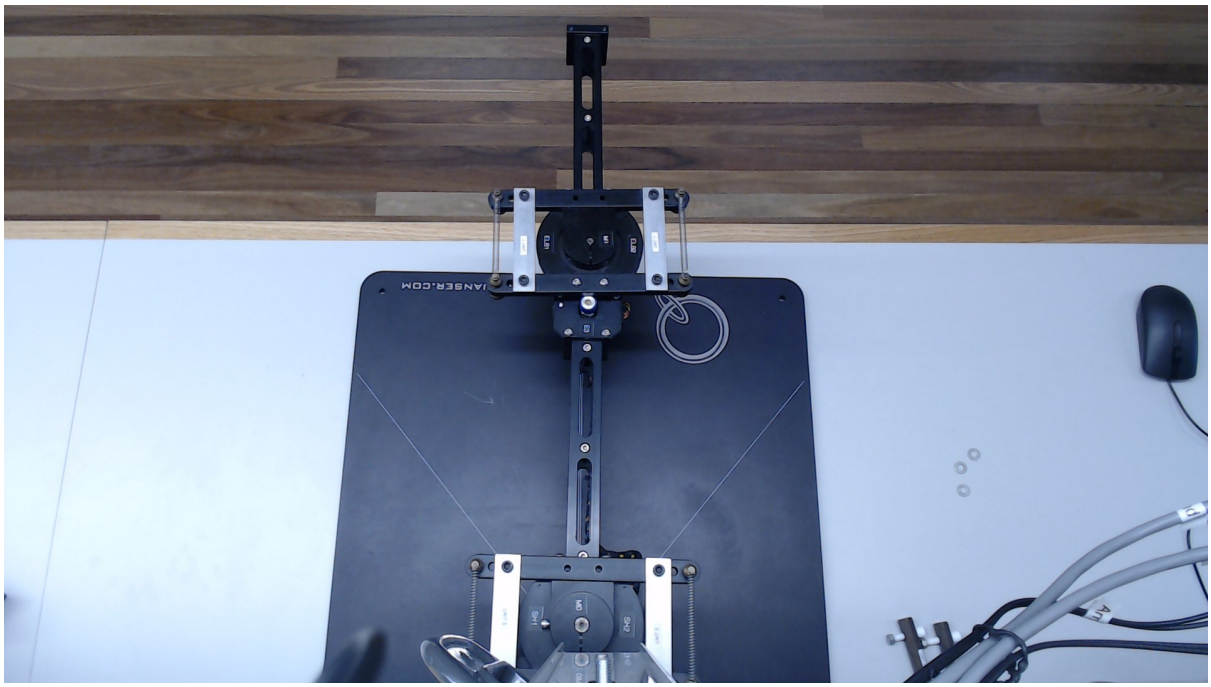

Figure 3.5: Rigid Joint Configuration of Quanser 2 DoF Serial Link Robot

Chapter 4

Results and Discussion

In this chapter, the results will be separated into two sections, where the first section will be discussing the results of the rigid joint robotic arm and the second section will be discussing the results of the flexible joint robotic arm. Each section will have the controllers as a subsection which illustrate the simulation and experimental results. Each subsection, there will be a table of control parameters used in the simulation and experiment for the respective controller.

The regulation problem was tested on the PD and Adaptive PD controllers, where the desired angular position was set to $\frac{\pi}{4}\,rad$ for both links. In addition, the tracking problem was tested on the Conventional and Modified SMC, where the robot manipulator will follow the desired polynomial trajectory in joint space. The goal is to achieve an angular position of $\frac{\pi}{2}\,rad$ and $\frac{\pi}{4}\,rad$ for the first and second link respectively, with respect to the desired angular velocities.

On the other hand, the input currents were saturated to $\pm 1.0\,A$ and $\pm 0.5\,A$ for first and second joint of the robot manipulator to protect itself from overloading. The saturation of the currents were also applied in the simulation.

Throughout the experiment, it was found that the 2 DoF Quanser Serial Flexible Joint Robot used has several limitations. For example, there are state-dependent dead-zone in the motors of both joints, where the value changes with respect to its position and how warmed up are the motors. Hence, it is not feasible to estimate a fixed dead-zone inverse to compensate the

dead-zone completely. Moreover, the sample time for the sensors and encoders are fixed to $0.002s$. It was also observed that there was always a noticeable deflection between the motor and link angular positions for the first joint of the flexible joint robot. To find out the cause of this issue, the robot was subjected to different positive and negative fixed step input current. Results shown that the further the robot moves, the deflection accumulates, regardless of which direction was the robot moving. A possible hypothesis is that there is an issue with the scaling of encoder data or the encoder measuring the link position uses auxiliary ring which might affect the accuracy of the measurement. In addition, there could be backlash and friction in the joint.

In the simulation, the simulation for Adaptive PD controller, Conventional SMC and Modified SMC were all subjected to a dead-zone between $-0.15A$ to $0.15A$.

4.1 Rigid Joint Robot Manipulator

4.1.1 PD Controller

Table 4.1: Simulation and Experiment Proportional and Derivative Gains for Rigid Joint Robot

	Joint 1 Gains		Joint 2 Gains	
	K_{p1}	K_{d1}	K_{p2}	K_{d2}
Simulation	3	0.6	8	0.8
Experiment 1	3	0.6	8	0.8
Experiment 2	10	0.6	9	0.8

4.1.1.1 Simulation

In this simulation, the robot model was assumed to be ideal without any uncertainties or disturbances. As shown in figure 4.1 and 4.2, the angular position error for both links converge to 0, but, there is a large pre-shoot and a small overshoot at the start of the simulation in link 2. Efforts have been made to tune the PD gains to eliminate the pre-shoot, however, the response in figure 4.2 was the best while minimizing steady-state error.

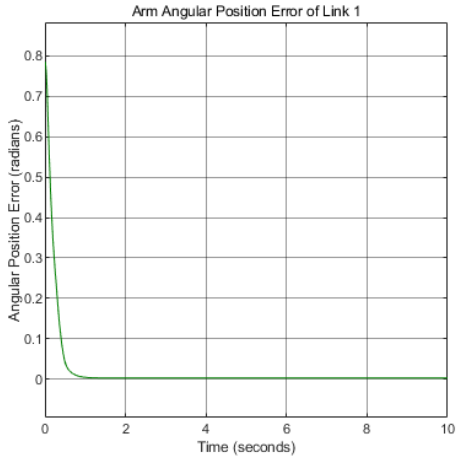

Figure 4.1: Simulation on Rigid Joint Robot: Joint Error of Link 1 with PD Controller

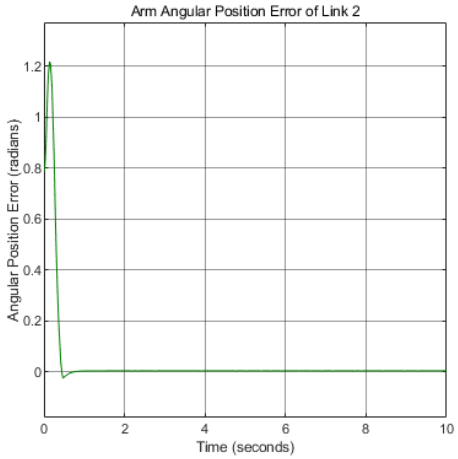

Figure 4.2: Simulation on Rigid Joint Robot: Joint Error of Link 2 with PD Controller

4.1.1.2 Experiment

The PD gains from the simulation were used in this experiment. However, results showed that steady-state error occurs. This was due to the presence of uncertainties in the actual system, which was mainly caused by the dead-zone in motors. From the input current illustrated in figure 4.5 and 4.6, although the input current was not zero, the angular position of the links was still stationary. The steady-state error caused by dead-zone is called the dead-zone effect.

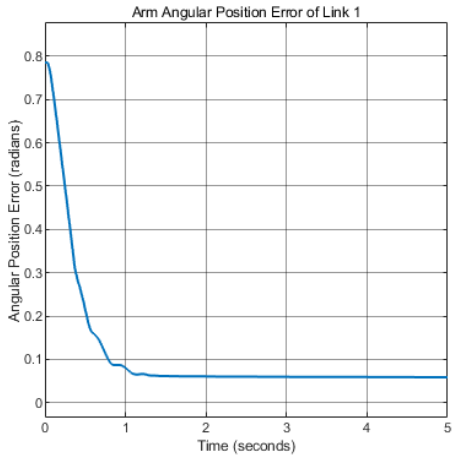

Figure 4.3: Experiment 1 on Rigid Joint Robot: Joint Error of Link 1 with PD Controller

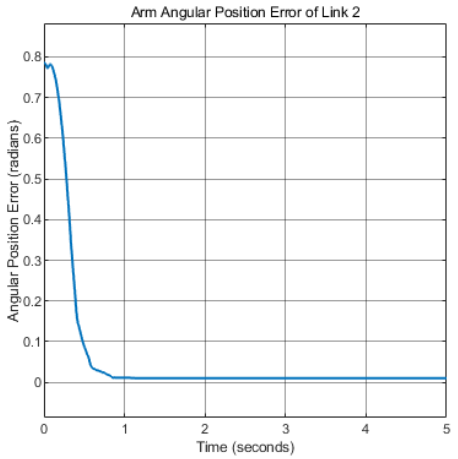

Figure 4.4: Experiment 1 on Rigid Joint Robot: Joint Error of Link 2 with PD Controller

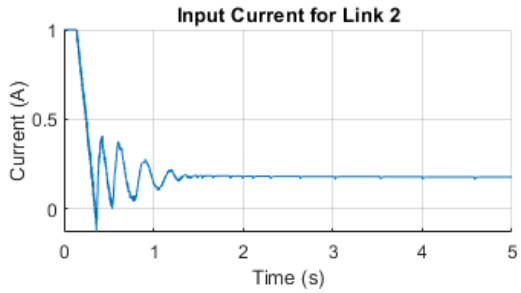

Figure 4.5: Experiment 1 on Rigid Joint Robot: Input Current of Link 1 with PD Controller

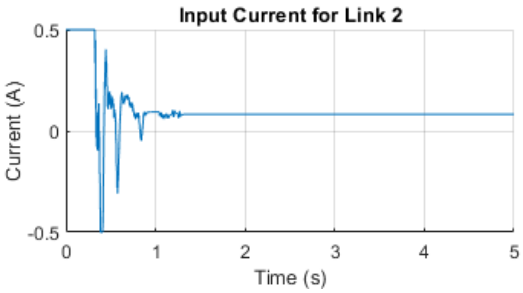

Figure 4.6: Experiment 1 on Rigid Joint Robot: Input Current of Link 2 with PD Controller

However, by increasing the proportional gain to the values showed in Experiment 2 in table 4.1, the steady-state error could be minimize. This is because, in the presence of uncertainties and measurement noise, the angular position error is inversely proportional to the proportional gain, which is proved in the stability analysis in [19].

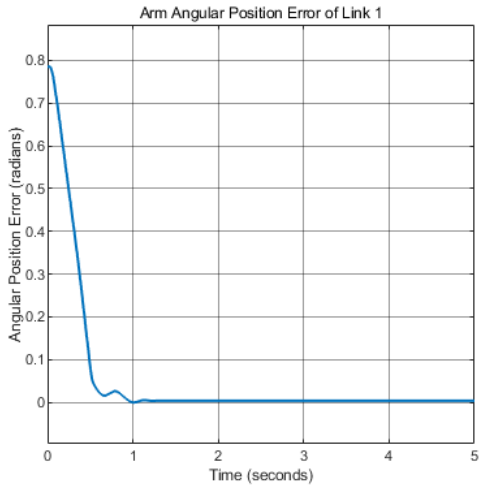

Figure 4.7: Experiment 2 on Rigid Joint Robot: Joint Error of Link 1 with Higher Gain PD Controller

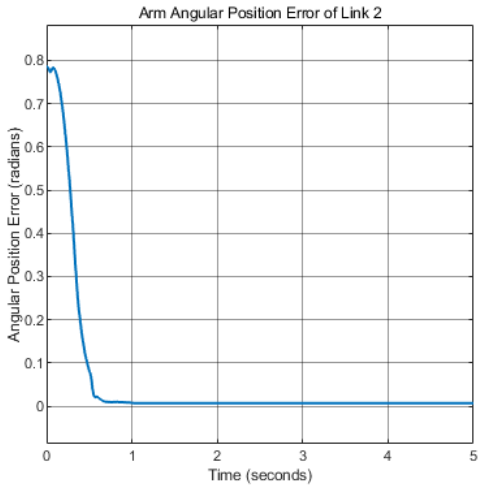

Figure 4.8: Experiment 2 on Rigid Joint Robot: Joint Error of Link 2 with Higher Gain PD Controller

4.1.2 Adaptive PD Controller

Since the dead-zone is state-dependent, an adaptive controller was designed and tested to compensate the uncertainties in an adaptive manner.

Table 4.2: Simulation and Experiment Control Parameters for Adaptive PD Controller on Rigid Joint Robot

	Joint 1 Gains					Joint 2 Gains				
	K_{p1}	K_{d1}	γ_1	b_r	b_l	K_{p2}	K_{d2}	γ_2	b_r	b_l
Simulation	3	0.6	5	0.1	0.1	8	0.8	5	0.1	0.1
Experiment	3	0.6	2	0.08	0.08	8	0.8	2	0.03	0.03

4.1.2.1 Simulation

In the simulation, the robot model was subjected a fixed value dead-zone of $\pm 0.15A$ for the ease of simulation. In order to verify this controller, the values of the dead-zone inverse used in the controller structure was a different value than the fixed dead-zone value, which is shown in table 4.2. This is because in reality, the dead-zone values are unknown and always changing and an approximation can only be made.

In figure 4.9 and 4.10, the original PD controller were compared against the adaptive PD controller, where both controllers were using the same PD gains. It was observed that the performance have improved by a decrease in overshoot and steady-state error in both links.

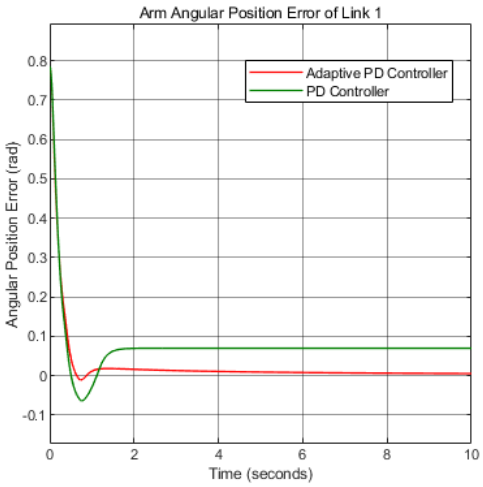

Figure 4.9: Simulation on Rigid Joint: Joint Error of Link 1 with Adaptive PD Controller

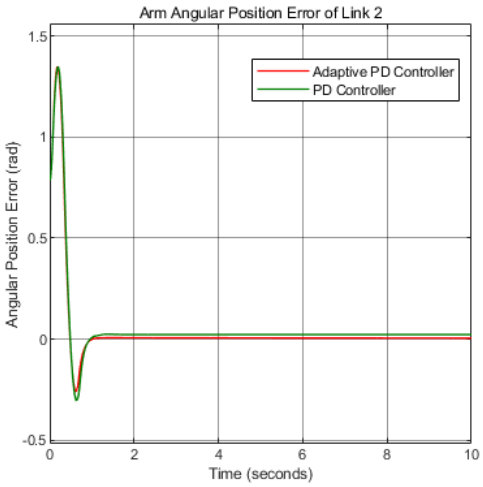

Figure 4.10: Simulation on Rigid Joint: Joint Error of Link 2 with Adaptive PD Controller

4.1.2.2 Experiment

The experimental results showed good results where the steady-state error approaches zero with minimal overshoot. However, the selection of γ and dead-zone inverse is highly sensitive and will affect the stability of the system. In some of the attempts where γ or dead-zone inverse parameters, b_r and b_l were not balanced, the system became unstable where the oscillations grow larger with respect to time.

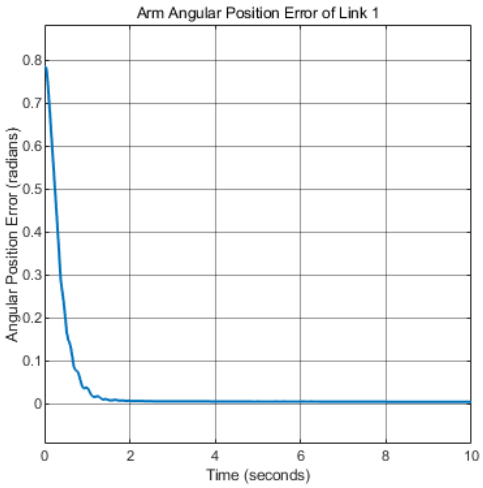

Figure 4.11: Experiment on Rigid Joint: Joint Error of Link 1 with Adaptive PD Controller

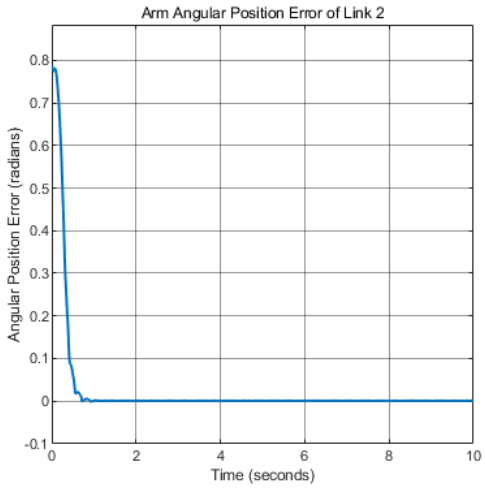

Figure 4.12: Experiment on Rigid Joint: Joint Error of Link 2 with Adaptive PD Controller

4.1.3 Conventional Sliding Mode Control

Theoretically, it is known that the PD controller does not work well with a tracking problem [19]. Hence, SMC was chosen to perform a time-varying trajectory and its successive derivative.

Table 4.3: Simulation and Experiment Control Parameters for Conventional Sliding Mode Control on Rigid Joint Robot

	Joint 1 Gains				Joint 2 Gains			
	K_{1_1}	K_{1_2}	λ_1	ψ_1	K_{2_1}	K_{2_2}	λ_2	ψ_2
Simulation	10	15	2	0.02	15	10	2	0.02
Experiment	10	15	2	0.02	15	10	2	0.02

4.1.3.1 Simulation

From the simulation results in figure 4.13-4.18, the controller did well in tracking the desired angular position and velocity in the presence of a fixed value dead-zone, except for the error between 3 to 5s in the joint velocity of link 2. Other disturbances and uncertainties were not included. There was no chattering as a boundary layer was introduced.

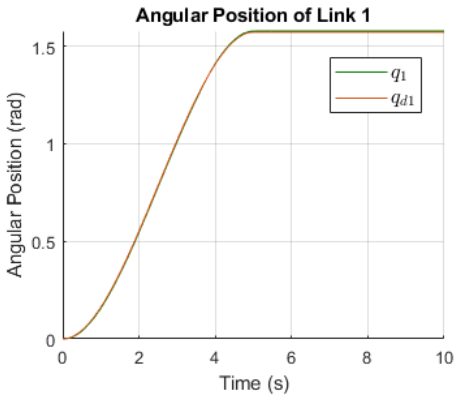

Figure 4.13: Simulation on Rigid Joint: Joint Position of Link 1 with Conventional SMC

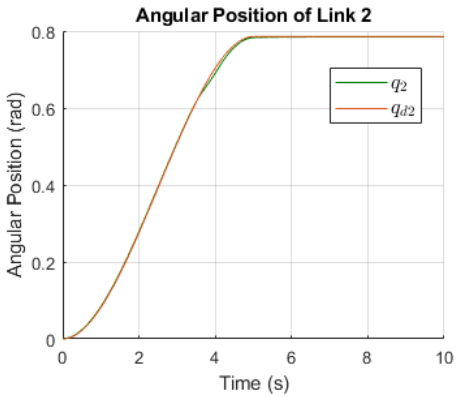

Figure 4.14: Simulation on Rigid Joint: Joint Position of Link 2 with Conventional SMC

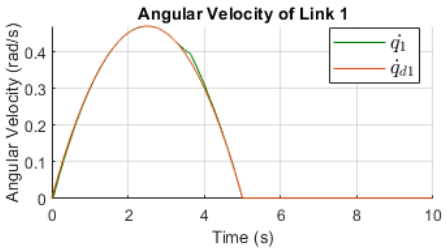

Figure 4.15: Simulation on Rigid Joint: Joint Velocity of Link 1 with Conventional SMC

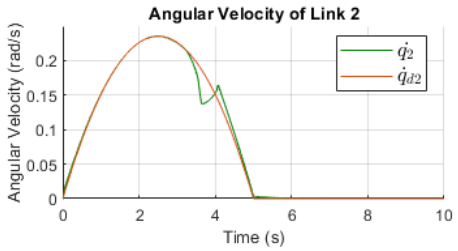

Figure 4.16: Simulation on Rigid Joint: Joint Velocity of Link 2 with Conventional SMC

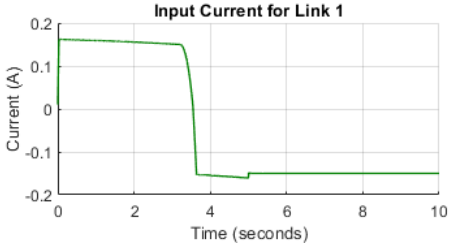

Figure 4.17: Simulation on Rigid Joint: Input Current of Link 1 with Conventional SMC

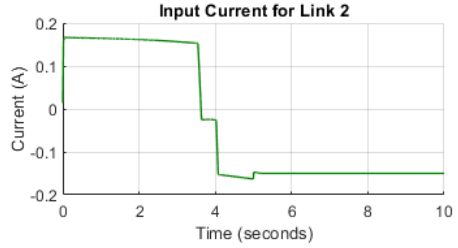

Figure 4.18: Simulation on Rigid Joint: Input Current of Link 2 with Conventional SMC

4.1.3.2 Experiment

According to the figures below, there is tracking error in both links. However, in link 1, the error converges to zero, while in link 2, there is a steady-state error of $0.024rad$. Although this controller is robust against uncertainties and disturbances, other factors might cause this error, such as mismatched perturbations, selection of control parameters, fixed sample time etc.

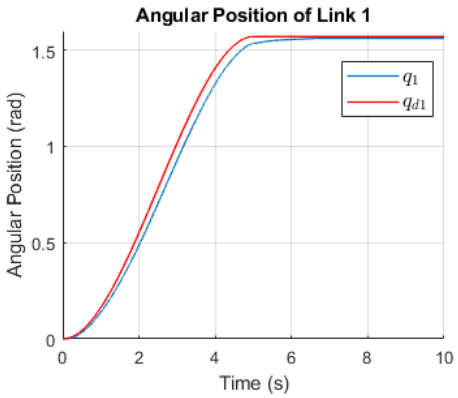

Figure 4.19: Experiment on Rigid Joint: Joint Position of Link 1 with Conventional SMC

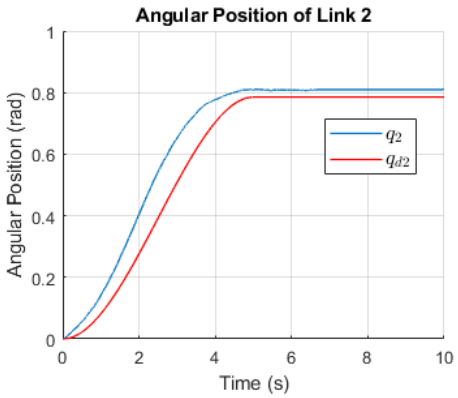

Figure 4.20: Experiment on Rigid Joint: Joint Position of Link 2 with Conventional SMC

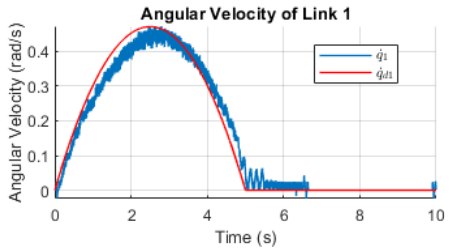

Figure 4.21: Experiment on Rigid Joint: Joint Velocity of Link 1 with Conventional SMC

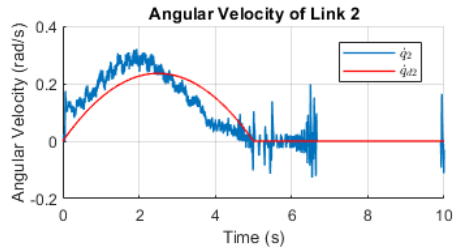

Figure 4.22: Experiment on Rigid Joint: Joint Velocity of Link 2 with Conventional SMC

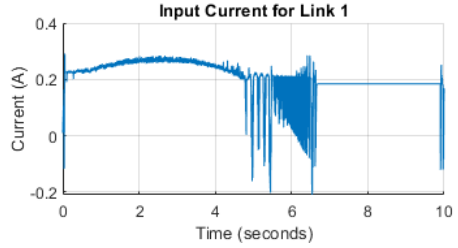

Figure 4.23: Experiment on Rigid Joint: Input Current of Link 1 with Conventional SMC

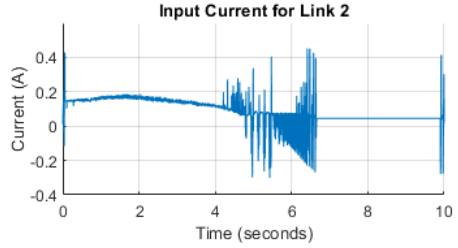

Figure 4.24: Experiment on Rigid Joint: Input Current of Link 2 with Conventional SMC

4.2 Flexible Joint Robot Manipulator

4.2.1 PD Controller

Table 4.4: Simulation and Experiment Control Parameters for PD Controller on Flexible Joint Robot

	Joint 1 Gains		Joint 2 Gains	
	K_{p1}	K_{d1}	K_{p2}	K_{d2}
Simulation	0.6	0.05	0.65	0.05
Experiment 1	0.6	0.05	0.65	0.05
Experiment 2	1.0	0.065	0.7	0.05

4.2.1.1 Simulation

As expected from the theory, the joint error for both links eventually converge to zero.

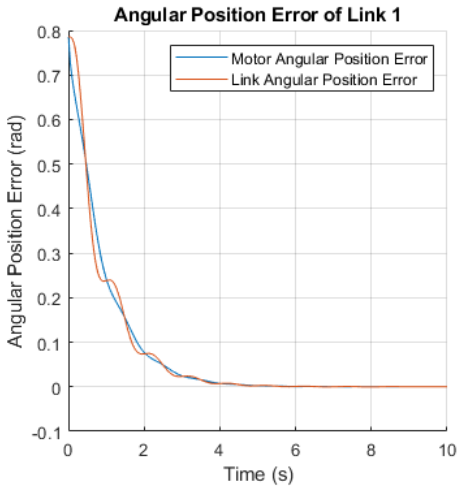

Figure 4.25: Simulation on Flexible Joint Robot: Joint Error of Link 1 with PD Controller

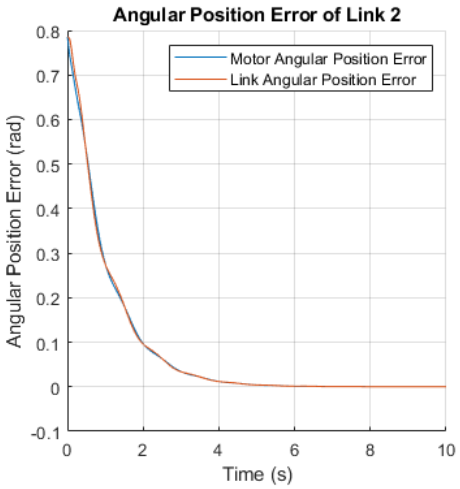

Figure 4.26: Simulation on Flexible Joint Robot: Joint Error of Link 2 with PD Controller

4.2.1.2 Experiment

Similar to the rigid joint configuration, the dead-zone effects occurs, which is shown in figure 4.27 and 4.28. The experimental results also showed oscillations of the link angular position while there was no oscillations in the simulation results.

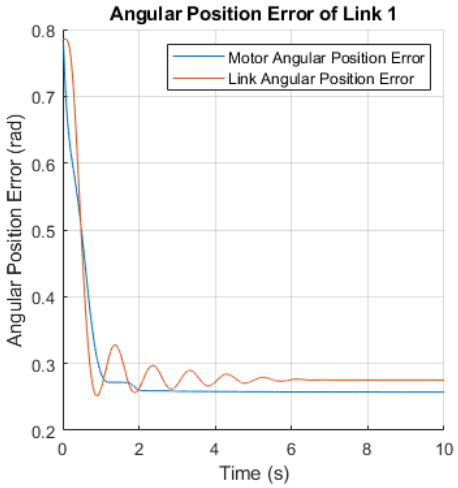

Figure 4.27: Experiment 1 on Flexible Joint Robot: Joint Error of Link 1 with PD Controller

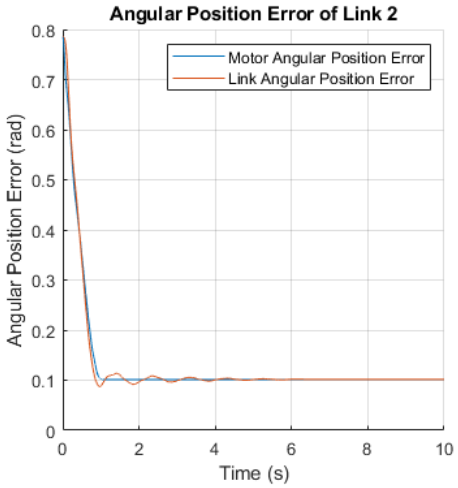

Figure 4.28: Experiment 1 on Flexible Joint Robot: Joint Error of Link 2 with PD Controller

Besides that, the steady-state error decreases as the proportional gain, K_p increases, which is illustrated in figure 4.29 and 4.30. However, for the flexible joint robot, increasing the K_p gain also leads to a more underdamped response as more oscillations are needed to achieve steady-state.

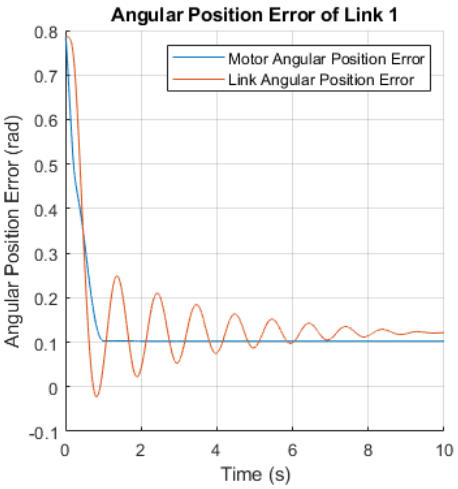

Figure 4.29: Experiment 2 on Flexible Joint Robot: Joint Error of Link 1 with Higher Gain PD Controller

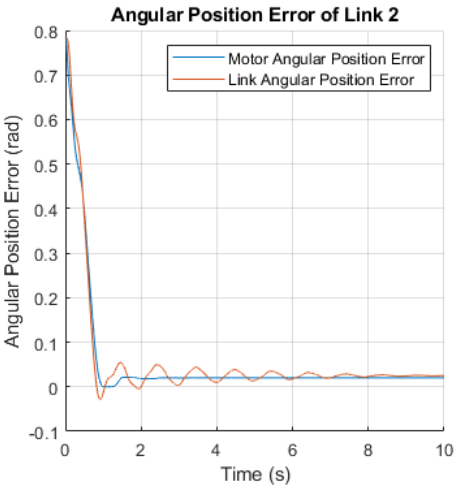

Figure 4.30: Experiment 2 on Flexible Joint Robot: Joint Error of Link 2 with Higher Gain PD Controller

4.2.2 Adaptive PD Controller

Table 4.5: Simulation and Experiment Control Parameters for Adaptive PD Controller on Flexible Joint Robot

	Joint 1 Gains					Joint 2 Gains				
	K_{p1}	K_{d1}	γ_1	b_r	b_l	K_{p2}	K_{d2}	γ_2	b_r	b_l
Simulation	0.6	0.05	5	0.1	0.1	0.65	0.05	5	0.1	0.1
Experiment	0.6	0.05	2.5	0.1	0.1	0.65	0.05	2.5	0.08	0.08

4.2.2.1 Simulation

By comparing the results of PD and Adaptive PD controller illustrated in figure 4.31 and 4.32, the simulation proved that the Adaptive PD controller can compensate a fixed dead-zone. However, the settling time has increased compared to the PD controller without dead-zone in figure 4.25 and 4.26. Although the settling time can be decreased by increase γ, or increasing the range of the dead-zone inverse parameters, the system might become unstable. Hence, the control parameters in table 4.25 were chosen.

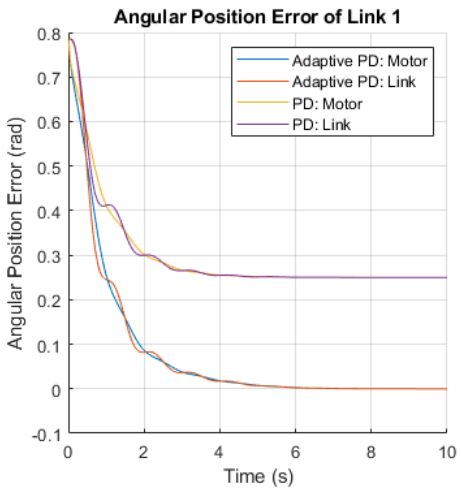

Figure 4.31: Simulation on Flexible Joint Robot: Joint Error of Link 1 with Adaptive PD Controller and PD Controller

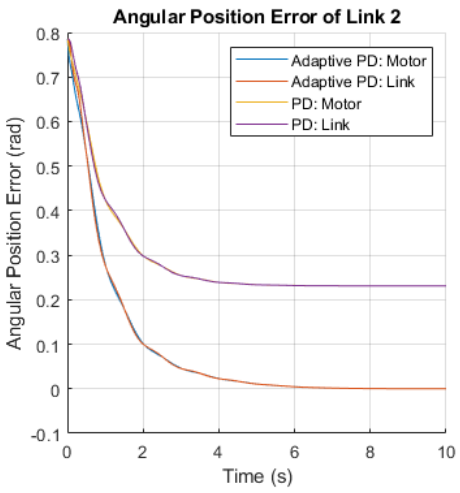

Figure 4.32: Simulation on Flexible Joint Robot: Joint Error of Link 2 with Adaptive PD Controller and PD Controller

4.2.2.2 Experiment

The transient response of the experimental results shown in figure 4.33 and 4.34 were very different from the simulation. A reason for this behaviour was due to the value of the control parameters, γ and the range of the dead-zone inverse. Besides that, the actual system is also subjected to more uncertainties and disturbances, which is why only a handful of control parameters that would provide stability in this system. However, results showed that the angular position error eventually approached zero.

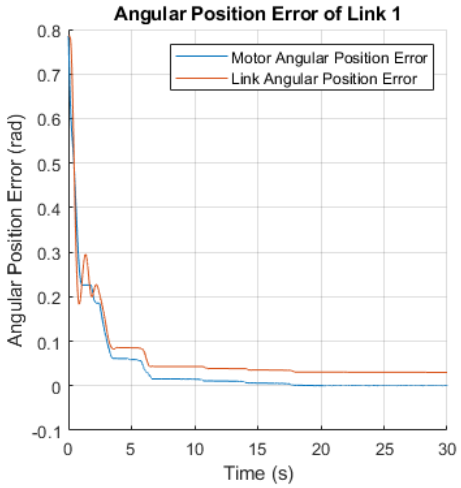

Figure 4.33: Experiment on Flexible Joint Robot: Joint Error of Link 1 with Adaptive PD Controller

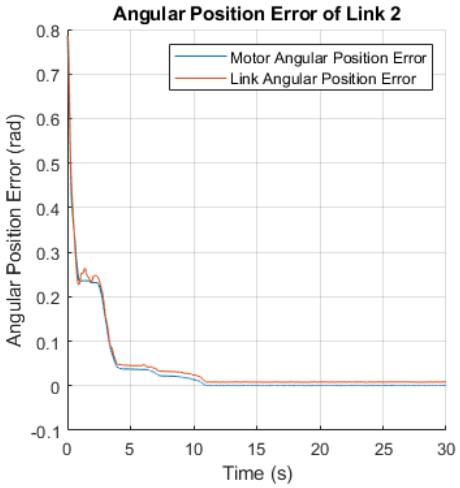

Figure 4.34: Experiment on Flexible Joint Robot: Joint Error of Link 2 with Adaptive PD Controller

4.2.3 Conventional SMC

In order to test the robustness of the conventional SMC, the exact controller used in the rigid joint robot was tested on the flexible joint robot with different sets of control parameters.

Table 4.6: Simulation and Experiment Control Parameters for Conventional SMC on Flexible Joint Robot

	Joint 1 Gains				Joint 2 Gains			
	K_{1_1}	K_{1_2}	λ_1	ψ_1	K_{2_1}	K_{2_2}	λ_2	ψ_2
Simulation	8	8	2	0.02	12	10	6	0.02
Experiment	8	8	2	0.02	12	10	6	0.02

4.2.3.1 Simulation

Since the controller was designed based on the dynamics of the rigid joint robot, the joint flexibility, other important dynamics, and the all the state variables were not included in the control law. As expected, the controller could not completely overcome the joint flexibility and results in poor tracking results in link 1 shown in figure 4.35 and 4.37. Although the tracking performance in link 2 is decent, there was a small steady-state error. In addition, there are high frequency response on the motor joint velocities. Even by tuning the control parameters, the controller still could not achieve a satisfactory tracking performance, there is always a trade off between the control parameters. For example, by increasing the proportional term K_{i_2} will lead to oscillations in the transient response, but zero steady-state error, and decreasing K_{i_2} will result in no oscillation in the transient phase but unacceptable steady-state error.

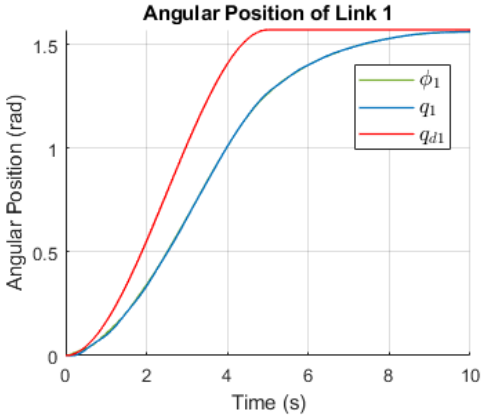

Figure 4.35: Simulation on Flexible Joint Robot: Joint Position of Link 1 with Conventional SMC

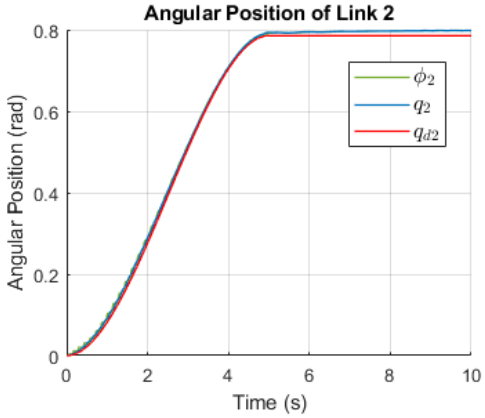

Figure 4.36: Simulation on Flexible Joint Robot: Joint Position of Link 2 with Conventional SMC

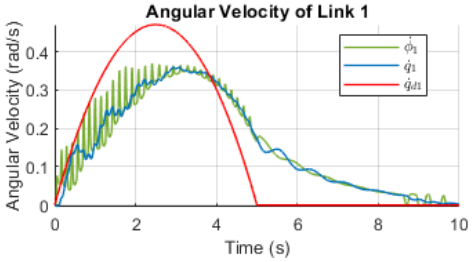

Figure 4.37: Simulation on Flexible Joint Robot: Joint Velocity of Link 1 with Conventional SMC

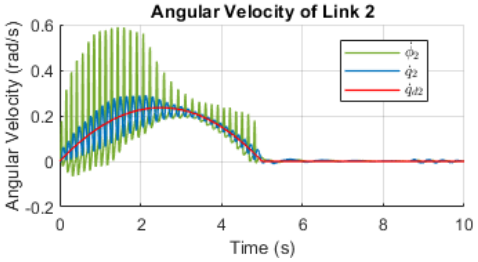

Figure 4.38: Simulation on Flexible Joint Robot: Joint Velocity of Link 2 with Conventional SMC

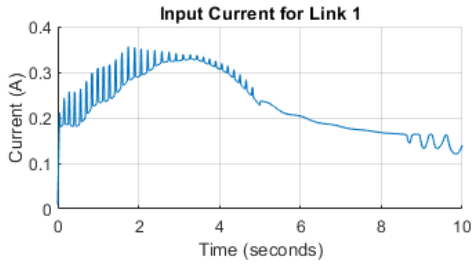

Figure 4.39: Simulation on Flexible Joint Robot: Input Current of Link 1 with Conventional SMC

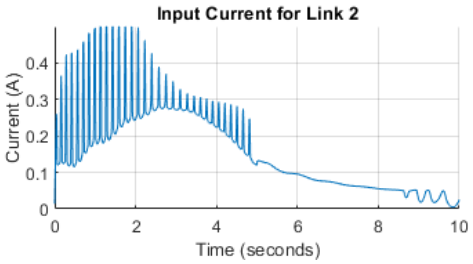

Figure 4.40: Simulation on Flexible Joint Robot: Input Current of Link 2 with Conventional SMC

4.2.3.2 Experiment

Similarly to the simulation, the tracking performance of this controller is poor with unacceptable tracking error. In figure 4.41 and 4.42, the actual trajectory of the motor and link position for both links were not smooth, which is most likely due to the ragged input current. Despite attempts in tuning the control parameters, the results did not improve, where there is a trade off between the oscillations in the transient phase and steady-state error.

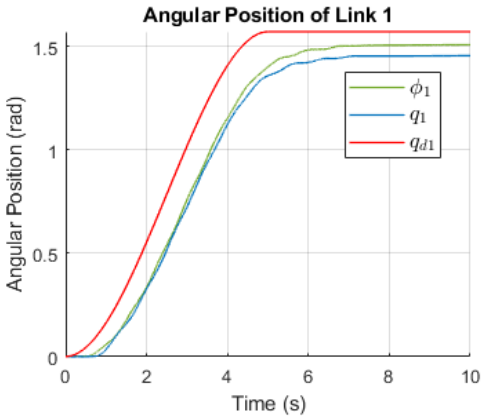

Figure 4.41: Experiment on Flexible Joint Robot: Joint Position of Link 1 with Conventional SMC

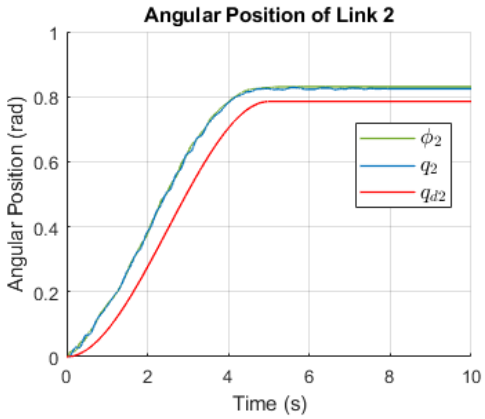

Figure 4.42: Experiment on Flexible Joint Robot: Joint Position of Link 2 with Conventional SMC

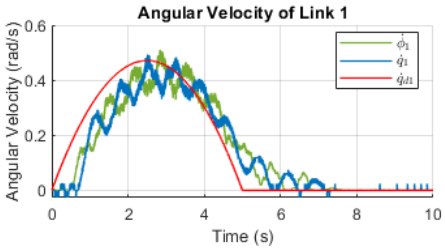

Figure 4.43: Experiment on Flexible Joint Robot: Joint Velocity of Link 1 with Conventional SMC

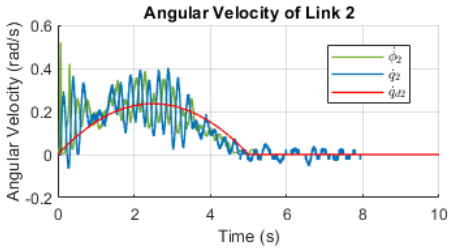

Figure 4.44: Experiment on Flexible Joint Robot: Joint Velocity of Link 2 with Conventional SMC

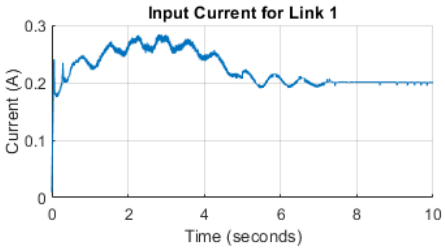

Figure 4.45: Experiment on Flexible Joint Robot: Input Current of Link 1 with Conventional SMC

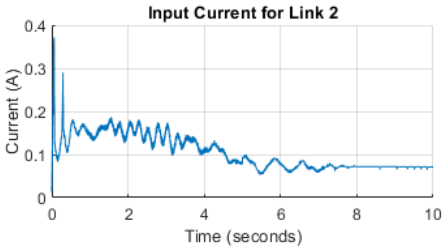

Figure 4.46: Experiment on Flexible Joint Robot: Input Current of Link 2 with Conventional SMC

4.2.4 Modified SMC

To improve the performance of the tracking problem in flexible joint robot, a modified SMC was introduced which uses three sliding surfaces to include all the state variables in the system.

Table 4.7: Simulation and Experiment Control Parameters for Modified SMC on Flexible Joint Robot

	Joint 1						Joint 2					
	K_{1_1}	K_{2_1}	c_{1_1}	c_{2_1}	c_{3_1}	ψ_1	K_{1_2}	K_{2_2}	c_{1_2}	c_{2_2}	c_{3_2}	ψ_2
Simulation	1	8	11	5	0.96	0.01	1	15	20	0.5	0.95	0.01
Experiment	1	8	11	5	0.96	0.01	1	15	20	0.5	0.86	0.01

4.2.4.1 Simulation

The simulation shows that this controller follows the given trajectory very well with a slight oscillation in the motor and link angular velocity on both links at $5s$, which is when the robot

stops moving. There is also no chattering and high frequency response in either the angular velocity or input current like the simulation of the conventional SMC.

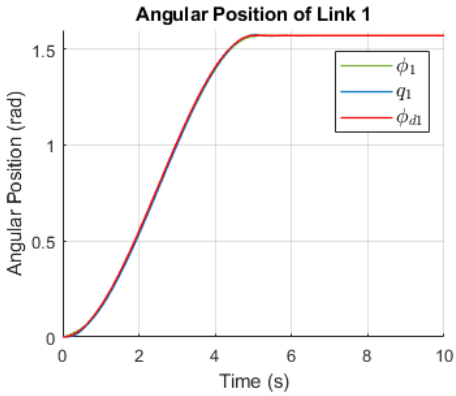

Figure 4.47: Simulation on Flexible Joint Robot: Joint Position of Link 1 with Modified SMC

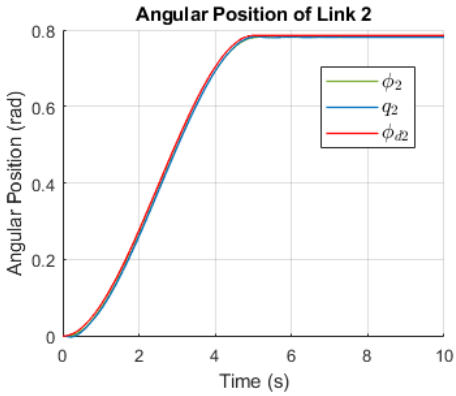

Figure 4.48: Simulation on Flexible Joint Robot: Joint Position of Link 2 with Modified SMC

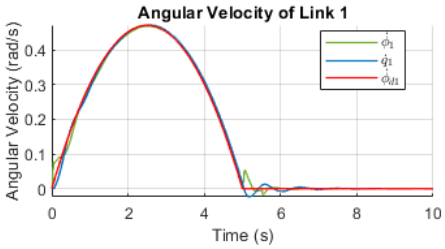

Figure 4.49: Simulation on Flexible Joint Robot: Joint Velocity of Link 1 with Modified SMC

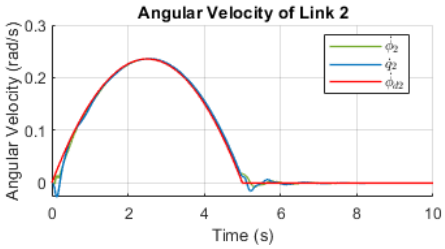

Figure 4.50: Simulation on Flexible Joint Robot: Joint Velocity of Link 2 with Modified SMC

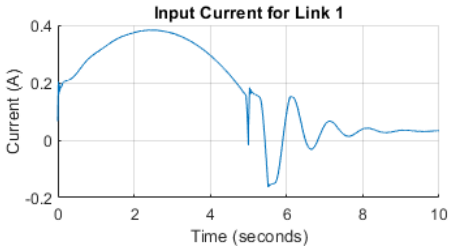

Figure 4.51: Simulation on Flexible Joint Robot: Input Current of Link 1 with Modified SMC

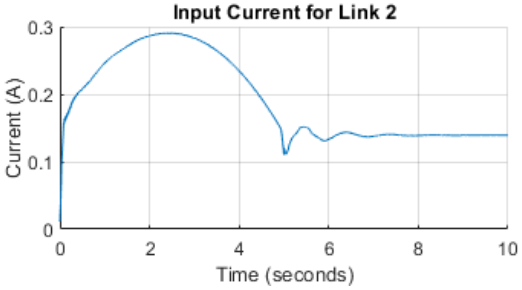

Figure 4.52: Simulation on Flexible Joint Robot: Input Current of Link 2 with Modified SMC

4.2.4.2 Experiment

According to figure 4.53-4.58, the experiment result showed promising results as the tracking error was relatively small. As mentioned before, there is a significant deflection between the motor and link angular position in link 1 that is not related to the execution of the controller. Hence, it can be assumed that the link position will converge to the motor position as the robot stops moving. However, there were still other factors like measurement noise that contributes to the high frequency response of the input current and the angular velocity.

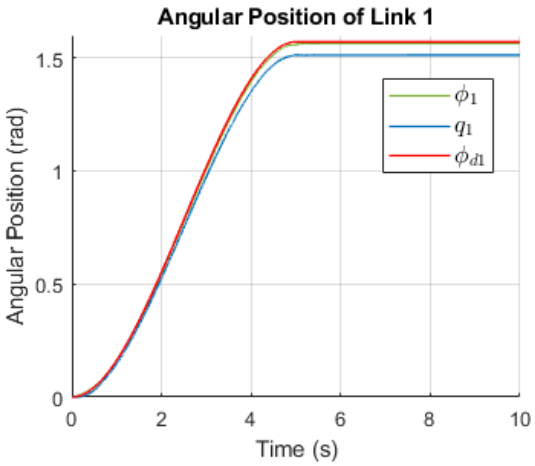

Figure 4.53: Experiment on Flexible Joint Robot: Joint Position of Link 1 with Modified SMC

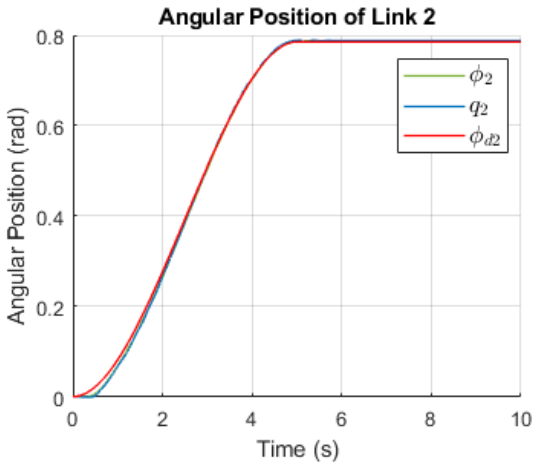

Figure 4.54: Experiment on Flexible Joint Robot: Joint Position of Link 2 with Modified SMC

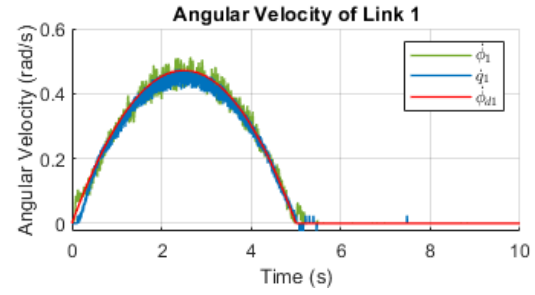

Figure 4.55: Experiment on Flexible Joint Robot: Joint Velocity of Link 1 with Modified SMC

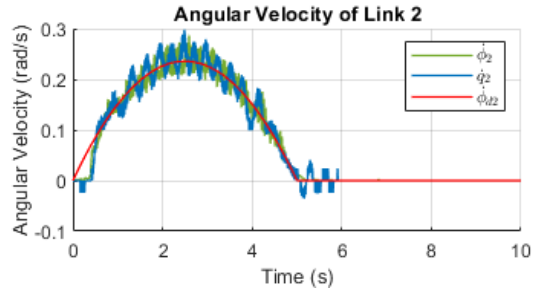

Figure 4.56: Experiment on Flexible Joint Robot: Joint Velocity of Link 2 with Modified SMC

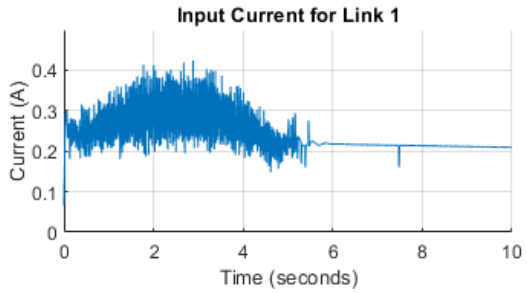

Figure 4.57: Experiment on Flexible Joint Robot: Input Current of Link 1 with Modified SMC

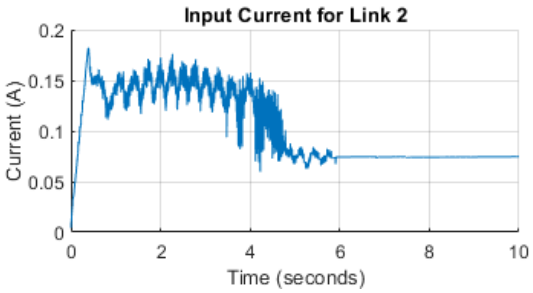

Figure 4.58: Experiment on Flexible Joint Robot: Input Current of Link 2 with Modified SMC

4.3 Overall Discussion

Table 4.8: Experimental Performance comparison between PD and Adaptive PD Controllers

Type of Controller (Regulation)	Type of Robot	Joint 1			Joint 2		
		Settling Time	Percentage Overshoot	Steady-state Error	Settling Time	Percentage Overshoot	Steady-state Error
PD Controller	Rigid	-	0	0.0589	0.8245	0	0.0103
	Flexible	-	0	0.2573	-	0	0.1013
PD Controller (High Gain)	Rigid	0.8745	0.0671	0.0033	0.6345	0	0.0071
	Flexible	-	0	0.1023	-	0	0.0201
Adaptive PD Controller	Rigid	1.2765	0	0.0038	0.6625	0.2808	0.00043143
	Flexible	6.60	0.0176	0.0002301	9.79	0.0273	0.0001534

As shown in table 4.8, the experimental performance of the controllers for the regulation problem can be compared. For rigid joints, it is evident that the PD controller with high gain performs the best in terms of settling time. and the steady-state errors were tiny, which is less than 0.5 degrees. The adaptive PD controller also ends with really small steady-state error, but the settling time was longer. As for flexible joints, the PD controller fails to converge to an acceptable range of steady-state error. However, the adaptive PD controller was able to achieve that with very small steady-state error but the downside was the very long settling time and very small percentage overshoot.

For the tracking problem, the comparison can be only made with the Conventional SMC and Modified SMC on the flexible joint robot. It is obvious that the Modified SMC will work better as it was designed for the flexible joint case. By only comparing the motor positions, there is

always an error in the trajectory for the Conventional SMC while the motor position in the Modified SMC follows the trajectory perfectly. The performance in velocity tracking in the Modified SMC is better than the Conventional SMC. However, chattering is more prominent in the Modified SMC than the Conventional SMC as there is more high frequency input current. An explanation for this might be due to the value of ψ used in the boundary layer, where the value of ψ used in the Conventional SMC is 0.02, while the value used for the Modified SMC is 0.01. Another reason for the high frequency response in all the experiments of the sliding mode control is related to the fixed sample time of $0.002s$, where this limitation does not apply in the simulation.

From the results above, it can be concluded that the simulation results behaved quite differently compared to the experimental results. This is because the dynamical model was not modelled accurately, where the dynamics of the robot was either simplified or ignored. The simulation was also not tested with the presence of external disturbances and measurement noises. Hence, during the experiment, the initial control parameters that were chosen often failed as the actual system is subjected to more uncertainties and hardware limitations than expected. Even though the effects on the system of each control parameter was studied beforehand, tuning the control parameters during the experiment was difficult and time-consuming.

Moreover, when comparing the results of the rigid and flexible joint robots, the flexible joint robot was harder to control due to the flexibility nature. For example, in the PD controller, the high proportional gain will lead to large oscillations, and low proportional gain will lead to a poor steady-state response. In contrast, high proportional gain will not cause oscillations in the rigid joint robot. In terms of controller design, there are more variables to consider as well, which complicates the control problem. However, the performance of the flexible joint robot can be as good as the rigid joint robot. For example, in the experimental results, the Modified SMC for flexible joint robot produces the better tracking performance compared to the Conventional SMC in rigid joint robot. Hence, it is not necessary that the rigid joint robot will always be more superior than flexible joint robot, as a good controller guarantees the performance.

Chapter 5

Conclusion

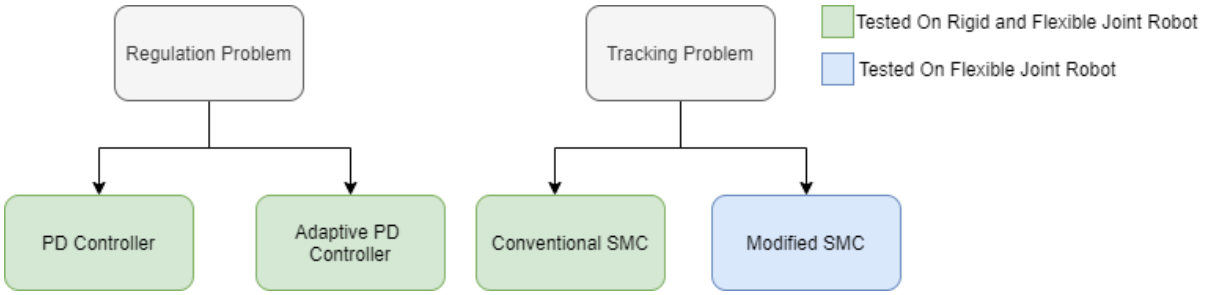

Figure 5.1: Proposed Controllers with Respective Control Problems

Joint flexibility is common in many robots which allows safe interactions with humans and the environment. However, systems with highly flexible joints are difficult to control. In this thesis, four robust controllers were designed for flexible and rigid joint robots and the respective control problem that is solved by each controller is shown in figure 5.1. The figure also shows which type of robot that the controller is tested on.

It was found that the actual robot was subjected to many uncertainties and hardware limitations, such as state-dependent dead zone in the motors, fixed sample time, deflection of motor and link position, measurement noise etc. These problems were not considered in the process of the controller design, which is a great test for robustness. This is because the controllers are designed to compensate these uncertainties and it is proved in the methodology. One of the aims of this thesis is to investigate how the proposed controllers can compensate these uncertainties.

All proposed controllers were also able to obtain experimental validation on a 2 DoF robot with highly flexible joints, especially on the Adaptive PD and Modified SMC.

Overall, for the regulation problem, the PD controller works well on the rigid joint robot by increasing the gain. While with high flexibility in the joints, the task failed due to large oscillations in the transient response, even with low gains. By incorporating a dead zone inverse and an adaptive mechanism with the Modified MIT rule into the PD controller, the rigid joint robot is able to achieve the regulation task well with lower PD gains. While in the flexible joint robot, it is able to complete the task with very little oscillation and a very small steady-state error but suffers from a very long settling time. As for the tracking problem, the Conventional SMC did not produce expected experimental results on the rigid joints, which might be due to the subjected uncertainties or unsuitable control parameters. For flexible joints, as expected, the Conventional SMC did poorly; however, the Modified SMC complete the task successfully.

5.1 Future Work

Further work can be done in improving the dynamical model by considering nonlinearities in the system. This would benefit in the controller design and also in the simulation process. Besides, to investigate the performance of the proposed controllers with a more challenging and fast trajectory.

Bibliography

[1] A. Albu-Schäffer, C. Ott, and G. Hirzinger, "A unified passivity-based control framework for position, torque and impedance control of flexible joint robots," *The international journal of robotics research*, vol. 26, no. 1, pp. 23–39, 2007.

[2] P. Corke, *Robotics, vision and control: fundamental algorithms in MATLAB® second, completely revised*. Springer, 2017, vol. 118.

[3] S. K. Dwivedy and P. Eberhard, "Dynamic analysis of flexible manipulators, a literature review," *Mechanism and machine theory*, vol. 41, no. 7, pp. 749–777, 2006.

[4] A. De Luca and W. J. Book, *Robots with flexible elements*. Springer, 2016, pp. 243–282.

[5] A. Ibrahim, R. Alexander, M. Shahid, U. Sanghar, R. Dsouza, and D. Souza, "Control systems in robotics: A review," *International Journal of Engineering Inventions*, vol. 5, pp. 2278–7461, 2016.

[6] D.-g. Zhang and S.-f. Zhou, "Dynamic analysis of flexible-link and flexible-joint robots," *Applied Mathematics and Mechanics*, vol. 27, no. 5, pp. 695–704, 2006.

[7] M. W. Spong, "Control of flexible joint robots: a survey," *Coordinated Science Laboratory Report no. UILU-ENG-90-2203, DC-116*, 1990.

[8] ——, "Modeling and control of elastic joint robots," 1987.

[9] T. D. Tuttle and W. P. Seering, "A nonlinear model of a harmonic drive gear transmission," *IEEE Transactions on Robotics and Automation*, vol. 12, no. 3, pp. 368–374, 1996.

[10] S. Ulrich and J. Sasiadek, "Trajectory tracking control of flexible-joint space manipulators," *Canadian Aeronautics and Space Journal*, vol. 58, no. 1, pp. 47–59, 2012.

[11] P. Tomei, "A simple pd controller for robots with elastic joints," *IEEE Transactions on automatic control*, vol. 36, no. 10, pp. 1208–1213, 1991.

[12] R. Kelly, R. Ortega, A. Ailon, and A. Loria, "Global regulation of flexible joint robots using approximate differentiation," *IEEE Transactions on Automatic Control*, vol. 39, no. 6, pp. 1222–1224, 1994.

[13] Y.-I. Son, H. Shim, and J.-H. Seo, "Set-point control of elastic joint robots using only position measurements," *KSME international journal*, vol. 16, no. 8, p. 1079, 2002.

[14] A. De Luca, B. Siciliano, and L. Zollo, "Pd control with on-line gravity compensation for robots with elastic joints: Theory and experiments," *automatica*, vol. 41, no. 10, pp. 1809–1819, 2005.

[15] L. Zollo, B. Siciliano, A. De Luca, E. Guglielmelli, and P. Dario, "Compliance control for an anthropomorphic robot with elastic joints: Theory and experiments," 2005.

[16] L. Sun, W. Yin, M. Wang, and J. Liu, "Position control for flexible joint robot based on online gravity compensation with vibration suppression," *IEEE Transactions on Industrial Electronics*, vol. 65, no. 6, pp. 4840–4848, 2017.

[17] S. Kumar, K. Jayaswal, and D. Kothari, "Investigation of feasible controller for position control of flexible joint manipulator using multiple control techniques," *SN Applied Sciences*, vol. 1, no. 12, p. 1634, 2019.

[18] J. L. Meza, V. Santibáñez, R. Soto, and M. A. Llama, "Fuzzy self-tuning pid semiglobal regulator for robot manipulators," *IEEE Transactions on industrial electronics*, vol. 59, no. 6, pp. 2709–2717, 2011.

[19] C. C. de Wit, B. Siciliano, and G. Bastin, *Theory of robot control*. Springer Science Business Media, 2012.

[20] K. P. Jankowski and H. Van Brussel, "An approach to discrete inverse dynamics control of flexible-joint robots," *IEEE transactions on robotics and automation*, vol. 8, no. 5, pp. 651–658, 1992.

[21] Y.-C. Chang, B.-S. Chen, and T.-C. Lee, "Tracking control of flexible joint manipulators using only position measurements," *International Journal of Control*, vol. 64, no. 4, pp. 567–593, 1996.

[22] S. E. Talole, J. P. Kolhe, and S. B. Phadke, "Extended-state-observer-based control of flexible-joint system with experimental validation," *IEEE Transactions on Industrial Electronics*, vol. 57, no. 4, pp. 1411–1419, 2009.

[23] H. Sage, M. De Mathelin, and E. Ostertag, "Robust control of robot manipulators: a survey," *International Journal of control*, vol. 72, no. 16, pp. 1498–1522, 1999.

[24] J. S. Yeon and J. H. Park, "Practical robust control for flexible joint robot manipulators," in *2008 IEEE international conference on robotics and automation*. IEEE, Conference Proceedings, pp. 3377–3382.

[25] W. Alam, A. Mehmood, K. Ali, U. Javaid, S. Alharbi, and J. Iqbal, "Nonlinear control of a flexible joint robotic manipulator with experimental validation," *Strojniški vestnik-Journal of Mechanical Engineering*, vol. 64, no. 1, pp. 47–55, 2018.

[26] V. Utkin and H. Lee, "Chattering problem in sliding mode control systems," in *International Workshop on Variable Structure Systems, 2006. VSS'06.* IEEE, Conference Proceedings, pp. 346–350.

[27] J. Liu, *Sliding mode control using MATLAB*. Academic Press, 2017.

[28] F. Moldoveanu, "Sliding mode controller design for robot manipulators," *Bulletin of the Transilvania University of Brasov. Engineering Sciences. Series I*, vol. 7, no. 2, p. 97, 2014.

[29] M. R. Soltanpour, M. Moattari *et al.*, "Voltage based sliding mode control of flexible joint robot manipulators in presence of uncertainties," *Robotics and Autonomous Systems*, vol. 118, pp. 204–219, 2019.

[30] W. Alam, S. Ahmad, A. Mehmood, and J. Iqbal, "Robust sliding mode control for flexible joint robotic manipulator via disturbance observer," *Interdisciplinary Description of Complex Systems: INDECS*, vol. 17, no. 1-B, pp. 85–97, 2019.

[31] C.-C. Fuh, "Variable-thickness boundary layers for sliding mode control," *Journal of Marine Science and Technology*, vol. 16, no. 4, pp. 288–294, 2008.

[32] X. Liu, C. Yang, Z. Chen, M. Wang, and C.-Y. Su, "Neuro-adaptive observer based control of flexible joint robot," *Neurocomputing*, vol. 275, pp. 73–82, 2018.

[33] *2-DOF Serial Flexible Joint Robot*, Quanser Inc., Markham, Ontario, 2013.

[34] P. Jain and M. Nigam, "Design of a model reference adaptive controller using modified mit rule for a second order system," *Advance in Electronic and Electric Engineering*, vol. 3, no. 4, pp. 477–484, 2013.

[35] C. Adrian, A. Corneliu, and B. Mircea, "The simulation of the adaptive systems using the mit rule," in *International Conference on Mathematical Methods and Computational Techniques in Electrical Engineering*, 2008, pp. 301–305.

Appendix A

The information listed below can be found in: `https://github.com/meiyan1712/Thesis`

- Rigid and flexible joint dynamical model derivation
- Modified SMC controller derivation
- Simulink block diagram for PD, Adaptive PD, Conventional SMC and Modified SMC