ABSTRACT

Title of Dissertation:	DYNAMIC CONTROL OF DEXTEROUS SOFT ROBOTIC SYSTEMS
	Weerakoon Mudiyanselage Lasitha Tharinda Weerakoon Doctor of Philosophy, 2023
Dissertation Directed by:	Professor Nikhil Chopra Department of Mechanical Engineering

Soft robotics has grown exponentially during the past two decades due to the possibility of expanded manipulation capabilities over existing rigid robots in complex, unstructured environments. Additionally, soft robots can mitigate current safety risks associated with rigid robots due to their softness. The inspiration for soft robotics has been mainly due to the many examples from nature, such as the agile environmental interactions of the elephant trunk and octopus tentacles. Over the past two decades, several applications ranging from underwater operations to minimally invasive surgeries to space operations have been identified for soft robots. Motivated by these, the overall objective of this dissertation is to study and develop control frameworks for high-fidelity motion control of soft robotic systems. This entails exploiting generalized dynamics models for robust/adaptive control strategies for achieving various operational tasks involved in non-ideal environments, utilizing integrated sensing technologies, and investigating control of underactuated soft robotic systems.

This dissertation delve into passivity-based adaptive task space control for soft robots,

mitigating uncertainty in the parameters as accurate parameter estimation is particularly hard in soft robotic systems. Further, this approach is extended to task space bilateral teleoperation of a soft follower-rigid leader system exploiting null space velocity tracking to achieve sub-task goals such as conforming to the degree of curvature limits in the soft robot. An enhanced dynamics model is also introduced tailored for planar soft robots and elaborate on passivity-based robust control methods for task space trajectory tracking within this context. This enhanced dynamics model is subsequently extended to encompass 3D spatial soft robots and a comprehensive framework for passivity-based robust task space bilateral teleoperation is discussed. Extensive numerical simulations and experiments are conducted to illustrate the efficacy of these proposed control frameworks. Moreover, to deploy soft robots in the real world, this dissertation study integrated sensing and control of soft robots and a stretchable soft-sensing skin for proprioception s introduced. The mapping from the strain signal to the curvature degree is estimated using a recurrent neural network. Further, an adaptive control framework for curvature tracking is proposed, leveraging the soft stretchable sensing skins and providing experimental evidence of its successful application.

This dissertation also introduces a novel robotic system known as the hybrid rigid-soft robot, composed of serially attached rigid and soft links, offering a fusion of the dexterity inherent to soft robots with the precision and payload capacity associated with rigid counterparts. Notably, the study demonstrates that well-established passivity-based adaptive and robust control techniques can effectively apply to this unique class of robots. A soft inverted pendulum with a revolute base is also introduced, establishing a scientific foundation and a methodological approach for introducing innovative soft robots in various practical applications. An energy-based controller is discussed for the swing-up and stabilization of the soft inverted pendulum system, highlighting the efficacy of the controller through simulations. Further, a comprehensive control architecture is developed for the swing-up and stabilization of a class of underactuated mechanical systems, including the soft inverted pendulum, by applying output partial feedback linearization and linear control techniques that avoid switching between controllers. The utility of this control architecture is illustrated using numerical simulations on the soft inverted pendulum.

These research endeavors collectively contribute to advancing the understanding of soft robotics and developing effective control strategies for various dexterous soft robotic systems.

DYNAMIC CONTROL OF DEXTEROUS SOFT ROBOTIC SYSTEMS

by

Weerakoon Mudiyanselage Lasitha Tharinda Weerakoon

Dissertation submitted to the Faculty of the Graduate School of the
University of Maryland, College Park in partial fulfillment
of the requirements for the degree of
Doctor of Philosophy
2023

Advisory Committee:
Professor Nikhil Chopra, Chair/Advisor
Professor Balakumar Balachandran
Professor Miao Yu
Assistant Professor Yancy Diaz-Mercado
Associate Professor Pratap Tokekar, Dean's Representative

Dedication

To my parents, brother and Anu.

Acknowledgments

I express my deepest gratitude to my advisor, Professor Nikhil Chopra, whose unwavering guidance, support, and invaluable counsel have been instrumental throughout my Ph.D. journey. Our interactive research discussions and conversations have not only enriched my understanding but also taught me how to formulate meaningful and technically sound engineering problems, exposing me to numerous intriguing research avenues. His encouragement, confidence, and kindness have consistently boosted my morale and provided motivation during the most challenging times. Beyond his academic guidance, Professor Chopra's caring nature made my transition into graduate student life seamless. Overall, his mentorship has been a cornerstone of my personal and academic development, and my Ph.D. journey under the guidance of Professor Chopra has been an incredible learning experience. I am grateful for being one of his students and a member of his research group.

I would also like to thank Professors Balakumar Balachandran, Miao Yu, Yancy Diaz-Mercado and Pratap Tokekar for taking time from their busy schedules to serve in my dissertation committee. I am also thankful to Professor Derek Paley who served in my dissertation proposal committee. Their valuable feedback on my research have assisted me in shaping this dissertation. I also want to thank Professor Elisabeth Smela for her support in one of the collaborative projects and her continuous encouragement. I am also grateful to all the professors that I took courses with and interacted with in any capacity at the University of Maryland. These interactions have undoubtedly taught me so many things. Specially, I want to thank Professor Abhijit Dasgupta, for introducing me to Professor Chopra. I am also grateful for the staff at the ME department at UMD for being friendly and helpful to me over the past years.

I wish to acknowledge the National Science Foundation (NSF Grant No. ECCS1711554) and the Army Research Laboratory (ARL Grant No. 310251-00001) for partially supporting my research endeavors. I am also grateful for the 2022 Maryland Robotics Center Graduate Research Assistantship (MRC-GRA) award and the Dean's fellowship (2023 summer) for supporting me.

I am grateful for all the lab mates of our research group which I had the pleasure of associating with: Nirupam, Gurtaj, Kushal, Jeff, Tianchen, Kaustubh, Alex and Shraddha. I have always enjoyed the intriguing research discussions and chit-chats with them. I also thank some of the undergraduate students I was fortunate to work with: Tomas, Julian, Joseph and Daniel, for helping me in some experiments. I am also thankful for Gizem, Celeste, Vipin, Abdul-Rahuman and Lautaro from the lab next door for their kindness, especially when I first came to UMD. I would also like to thank all the other friends I made through various interactions who helped me in some way or the other to get through the graduate student life. Special thanks goes to Dilhara, Sajani, Umesha, Yashish and Dushyanthi for being so close and helpful in every aspect of my life at UMD.

I also thank the MRC for the amazing facilities which helped me to conduct my experiments. Specifically, I would like to thank the lab manager, Dr.Ivan Penskiy, for his continuous support in helping me set up some experiments. I am also grateful for the numerous events organized by the MRC which gave me great opportunities to interact with various people that helped me get some feedback on my research from a wider community.

I wish to extend my gratitude to all of my past teachers back in Sri Lanka. I would not have been here if not for all their support. Specially, I would like to thank Dr.Sanjeeva Maithripala, my undergraduate research advisor at the University of Peradeniya, for motivating me and guiding me through the initial steps in controls research. His continuous support has been a blessing in my academic career. I would also like to thank Professor Jordan Berg from Texas Tech./NSF, with whom I collaborated with prior to joining University of Maryland, for truly inspiring me to pursue a Ph.D. in the United States. I am also thankful to Dr.Ravindra Herath for his support and guidance in seeking graduate studies.

Most importantly, I am extremely thankful for my father and mother, SB Weerakoon and Prabha, for their unconditional love, encouragement and believing in me. It was my parents who paved the way for my success with their motivation and words of wisdom through out my entire life. I also thank my brother Rajitha for being there for me always and supporting me. I can not thank enough my loving wife, Anuththara for tolerating me and my unusual study/work hours when things got pretty tight, especially when deadlines were approaching. As a Ph.D. graduate herself, she possessed an innate understanding of my aspirations, dreams, and goals, consistently providing encouragement and unwavering support to help me unleash my full potential. At times, her constant cheers were the thing got me through days when everything else research-wise was not going so well. I am also incredibly grateful for my in-laws, who were constantly encouraging and supporting me, from far away.

Lastly, I would like to thank a very special person, my beloved son, Ayod Sansith, for not fussing and cooperating with me to finish up my dissertation, who was still figuring out the dynamics and control of his dexterous body, merely 1 month of age at the time!

-Lasitha Weerakoon (November, 2023)

Table of Contents

List of Tables

List of Figures

List of Abbreviations

ANN	Artificial Neural Networks
ATSBT	Adaptive Task Space Bilateral Teleoperation
CC	Constant Curvature
COM	Center of Mass
cPDMS	Carbon nanotube infused Polydimethylsiloxane
DOF	Degree of Freedom
EG	Exfoliated Graphite
FEM	Finite Element Method
GF	Gauge Factor
HyRiSo	Hybrid Rigid-Soft
LQR	Linear Quadratic Regulator
LSTM	Long Short Term Memory
MoCap	Motion Capture
ML	Machine Learning
MLP	Multilayer Perceptron
OPFL	Output Partial Feedback Linearization
PCC	Piecewise Constant Curvature
PDMS	Polydimethylsiloxane
PWM	Pulse Width Modulation
RMSE	Root Mean Square Error
RNN	Recurrent Neural Network
ROA	Region of Attraction
RTSBT	Rdaptive Task Space Bilateral Teleoperation
RWP	Reaction Wheel Pendulum
SEA	Serial Elastic Actuators
SIPR	Soft Inverted pendulum with Revolute base
TCP/IP	Transmission Control Protocol/Internet Protocol

Chapter 1: Introduction

1.1 What are soft robots?

Conventional rigid robots are typically programmed for efficient execution of specific tasks in controlled environments [1]. However, their adaptability in response to unexpected changes is often limited due to the rigidity of their structural materials. Soft robots, on the other hand, are systems capable of autonomous behavior fabricated from materials that have low elastic moduli, in the range of those of biological materials (10^4–10^9Pa) [2]. Moreover, unlike traditional rigid robots, soft robots typically do not have actuated joints in their structure, and the motions are achieved by continuous deformation of the soft body [3]. Consequently, they exhibit a high degree of freedom in engaging with their surroundings flexibly and safely and swiftly adapting to abrupt alterations in their operating context. This is precisely what is observed in biological systems. Thus, the inspiration for the development of soft robots has been the robust and agile environmental manipulation by animals, such as the octopus's varied use of tentacles and elephant's dexterous trunk [4, 5, 6].

From a robotics application standpoint, the compliance and dexterity of soft robots can be utilized for effective manipulation in unstructured environments. Moreover, these robotic systems could reduce the harm that could be done in the case of contact, thanks to the softness of the body, thus potentially mitigating safety risks associated with the classical rigid robots [7]. The potential applications of soft robots are numerous and they include minimal invasive surgery [8], endoscopy [9], inspection tasks [10], search and rescue [11, 12], locomotion [13] and agriculture harvesting [14] among others.

1.2 Soft robotics: the research field

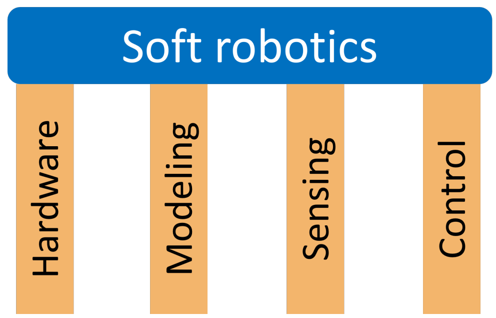


Figure 1.1: Research directions in the field of soft robotics.

Soft robotics is an interdisciplinary field that amalgamates various branches of knowledge, including biology, material science, continuum mechanics, and robotics, all with the overarching goal of addressing fundamental questions while yielding practical applications [3]. In the pursuit of advancing soft robotics, numerous hardware platforms have been put forth, each demonstrating improved reliability and increased functionalities [15]. This evolution has predominantly concentrated on the technological aspects of the field, resulting in a diverse array of hardware solutions. However, this profusion has introduced a new challenge – the need to formulate efficient control strategies capable of leveraging the unique properties of soft robots. To that end, the researchers have investigated different modeling and control frameworks for soft robots. Moreover, to realize the complete autonomy of soft robots, it is also essential to equip them with advanced proprioception (sensing capabilities) that do not hinder their inherent properties. In view of these aspects, four main research directions can be identified in the soft robotics field - hardware design, modeling, sensing and control, as illustrated in Figure 1.1. This dissertation primarily explores the controls development research avenue, which in fact is closely intertwined with modeling as well as sensing. A brief discussion on the recent advances in these related aspects are given below.

1.2.1 Analytical modeling

A soft body is an infinite-dimensional dynamic system, making it a challenge to articulate the kinematics and dynamics of the soft robot in a way that is amenable to control theoretical formulations. Over the past two decades, several reduced-order mathematical models have been proposed [3, 5], as summarized below.

The piecewise constant curvature (PCC) formulation is the most widely used modeling approach for soft robots [3]. It reduces the dimensionality of the soft robot by assuming it to be composed of a finite number of segments, each with a time-dependent constant curvature (CC) [16]. The CC segments are attached so that the resulting curve is differentiable everywhere. To describe each segment, only three parameters are required: the degree of curvature, the arc angle or the length of the segment, and the bending plane. The PCC formulation was initially used to analyze the kinematics of soft robots and has been studied extensively [16, 17, 18]. A few PCC dynamics models have also been proposed [19, 20, 21, 22]. Recently, a modeling approach proposed by Della Santina et. al. [22], which models a PCC soft robot as if it were a rigid robot using an augmented formulation, has opened avenues to implement and test traditional control algorithms that were developed for rigid robots.

Another popular modeling approach for soft robots is based on Cosserat rod theory. This represents a body as a stack of infinitesimal solid bodies, making it infinite dimensional [23]. Reduced order models have been developed by using constraints on the deformations, such as assuming PCC of segments or by including only a finite number of solid bodies in the model [24]. Both kinematics [25, 26] and dynamics models [27, 28] have been shown leveraging Cosserat rod theory, and they have proven to be more accurate than simplified PCC models. However, solving the resulting partial differential equations of these models for control applications remains cumbersome.

The authors in [29, 30] proposed Finite Element Method (FEM) based approaches for modeling soft robots under quasi-static conditions, and ref. [31] proposed a dynamic model using a linearized FEM. However, the drawback of FEM is the high dimensionality due to the fine mesh required for an accurate approximation. Among other models, there are also dynamics models using Ritz-Galerkin methods [32] and models employing discretizations of a continuum rod with lumped parameters [33]. Recently, a soft robot dynamics model based on polynomial curvature was introduced [34], which enables additional degrees of freedom in modeling. It was used to model a soft inverted pendulum with affine curvature [35].

1.2.2 Control

Soft robot control can be categorized into two main groups: kinematic controllers and dynamic controllers. Kinematic control exclusively focuses on the robot's posture, disregarding any consideration of forces. Consequently, this approach relies on a steady-state model, allowing it to execute control based solely on position or velocity/acceleration parameters. On the other hand, dynamic control considers the system's dynamics, incorporating physical forces like inertia and gravity. As a result, dynamic control can attain a higher level of precision than kinematic controllers and offers greater versatility in enabling force-based interactions with the robot's environment.

Among kinematic controllers, the authors in [36] introduced an adaptive visual servo controller based on a PCC formulation without requiring knowledge of the actual values of the manipulator's length. A closed-loop position controller using a FEM-based kinematic model was demonstrated in [37]. The authors in [38] introduced a real-time closed-loop kinematic controller for configuration space trajectory tracking using a PCC formulation.

Considering dynamic controllers, a majority of the state-of-the-art dynamic controllers are open-loop due to the complex nature of the models, which have a high computational cost for closed-loop control. Feedforward position control of soft robots using PCC-based dynamic models has been proposed [20, 34, 39]. Using the augmented formulation introduced in [21], dynamic feedback controllers for soft robots have been designed for curvature and surface tracking using an impedance controller [21, 22, 40]. A posture controller using a PCC-based dynamics model has been proposed [41] and a sliding mode controller for tracking [42]. All of these controllers assumed perfect knowledge of the soft robot models. To handle uncertainty in dynamic parameters, a task space adaptive control for soft robots has been proposed in [43]. Using an augmented model, the curvature space adaptive control of soft robots has been proposed in [44]. Configuration space trajectory tracking using an interval-arithmetic-based robust controller was proposed [45]. Recently, in [46], a nonlinear adaptive position and stiffness controller for pneumatic soft robots was proposed.

The interested reader is referred to the recent survey papers [3, 5] for a comprehensive review on soft robot modeling and control.

1.2.3 Proprioception

Much work on developing embedded sensing for soft robots has emerged in recent years [47]. Some studies demonstrated the use of commercially available flex sensors embedded in soft robots to measure the bending of the body [48, 49, 50, 51]. However, a drawback of these commercial flex sensors is their potential to stiffen the soft bodies since they lack the same level of softness as the robot itself [52, 53]. As a result, various research groups have turned their focus to creating soft embedded sensors that preserve the mechanical compliance of soft robots, thus avoiding this issue [53, 54, 55]. The review paper [56] discusses soft pneumatic actuators fabricated entirely with additive manufacturing methods with self-sensing capability. Recent efforts in embedded sensing technology have used polydimethylsiloxane (PDMS) filled with carbon nanotubes (cPDMS) [52, 57]. Another approach presented in [58] employs off-the-shelf conductive silicone elastomer sheets that are laser-cut into Kirigami patterns and subsequently bonded to the soft robot's skin, resulting in soft piezoresistive silicone sensors. These sensors have been employed to predict the steady-state 3D configuration of the soft robot through the application of a trained Recurrent Neural Network (RNN). This strategy has been harnessed for developing data-driven disturbance observers to estimate external forces acting on soft robots [59].

1.2.4 Challenges and opportunities

Due to the complexity of the behavior of the soft body, it has been a challenge to articulate suitable models to describe the dynamics of the soft robots. A few exact models based on Cosserat rod theory and reduced order models based on certain assumptions on the curvature of the robot (constant or polynomial) have emerged in the literature for soft robots. However, the efficacy of these models in control aspects has not been thoroughly investigated, for example, the susceptibility of these dynamic models when presented with external forces. Additionally, as soft robots are inherently infinite dimensional, any reduced-order modeling will introduce some form of underactuation in the model. However, these reduced-order models are often assumed to be fully actuated in the considered order. While most of the proof of concept work in the literature has demonstrated the effectiveness of these fully actuated approximations, the downside is that relying on a coarse approximation during control design might result in performance decline and instability. Also note that the designing of control frameworks for soft robots has been under-developed due to the lack of adoptable dynamic models developed for soft robots. The exact models are computationally costly for implementing real-time controllers, while the reduced-order models might be unrealistic for practical implementations. Furthermore, the existing controllers developed for soft robots assume the perfect knowledge of the robot model. These might fail if the controllers are not tuned perfectly due to the uncertainties arising from the parameters, such as the stiffness and damping of the soft robot. Furthermore, the actuation limits (such as applicable maximum pressure for a pneumatically actuated soft robot) and the configuration space limits (such as the curvature limits) should also be considered in developing the control frameworks as these influence the controller's performance.

Given these challenges in soft robotics, a crucial focus lies in developing control strategies that empower soft robots to perform precise and controlled movements in the presence of continuous interactions with unstructured and dynamic environments. These control strategies must demonstrate robustness, allowing them to handle uncertainties introduced in the model.

Furthermore, effectively addressing underactuation in soft robots remains a significant aspect of research and development of this class of robots. This necessitates the incorporation of underactuation into control strategies by considering the distinctive dynamics of soft robotic systems, where there are fewer control inputs than configuration variables. Achieving control in such systems demands innovative approaches and methods that leverage these robots' inherent compliance and flexibility, allowing them to accomplish tasks in ways that traditional rigid robots cannot.

1.3 Scope and contributions of the dissertation

The primary scope of this dissertation revolves around the dynamic control of soft robotic systems. In pursuit of this, and motivated by the above challenges and opportunities, this dissertation contributes to the soft robotics field by investigating the following key challenges:

- *Uncertainty mitigation and robust stabilization of soft robots:* exploring methods to address uncertainties and ensure the robust stabilization of soft robotic systems.
- *Practical embodied sensing:* exploring the application of stretchable soft sensors for providing feedback in the motion control of soft robots.
- *Dexterous and precise manipulation of hybrid rigid-soft robots:* exploring the manipulation capabilities of hybrid rigid-soft robots, focusing on their dexterity and precision while studying the impact of uncertainties and use different actuation modalities to achieve task space tracking.

- *Swing-up and stabilization of underactuated soft robots:* exploring strategies for swinging up and stabilizing a novel underactuated soft robotic model, specifically the soft inverted pendulum with a revolute base.

1.3.1 Contributions

By addressing the challenges identified above, this dissertation aim to contribute to advancing knowledge and capabilities in soft robotics. The main contributions of this dissertation are listed in detail below.

Given uncertainty mitigation and robust stabilization of soft robotic systems, this dissertation investigates classical passivity-based adaptive control and passivity-based robust control approaches for task space trajectory tracking of soft robotic manipulators and show their utility for control of soft robots. This is the first time such passivity-based controllers were investigated for soft robot control. Moreover, this dissertation introduce an enhanced modeling framework for multi-link soft robots, which captures the inertia effects of the soft robots. Further, these passivity-based task space controllers are extended for bilateral teleoperation frameworks, which were studied for the first time.

Considering enabling the soft robots to be deployed in non-ideal scenarios in the real world, this dissertation investigates the utilization of proprioceptive sensing to obviate the need for external sensors such as motion capture cameras for curvature measurements. To that end, a stretchable soft-sensing skin retrofitted to a soft robotic segment is introduced for the degree of curvature estimation. The utility of this sensor is demonstrated by employing a passivity-based adaptive controller for curvature tracking. This demonstrates, for the first time, the utilization of soft sensing skins for precise control of soft robots.

Given dexterous and precise manipulation, a novel robotic system is introduced, a hybrid rigid-soft robot composed of serially attached rigid and soft links. This robot would provide both the dexterity of soft robots and the precision and payload capacity of a rigid robot. One challenge in controlling these hybrid robots is that there could be increased uncertainty in parameters, including the actuation parameters. This dissertation demonstrates that the well-known passivity-based adaptive control and robust control can be utilized to address this challenge. Further, if the hybrid robots are constructed to be redundant manipulators, it is shown that the null space velocity of this class of robots can be used for obstacle avoidance sub-task.

As a first step toward understanding how softness impacts the control performance of underactuated soft systems, this dissertation introduces a novel system coined as a soft inverted pendulum with a revolute base and study the swing-up and stabilization of this system. Motivated by swing-up controllers developed for classical underactuated systems, a two-stage control method is investigated: an energy-based swing-up controller and a linearized stabilizing controller. Using theoretical proofs and numerical simulations, this dissertation shows that this soft inverted pendulum with a revolute base can be stabilized upright, a feat never achieved in the literature before. Further, a single continuous control strategy is introduced to enlarge the region of attraction of a linear controller for the swing up and stabilization of a class of underactuated systems, which includes the SIPR system, employing output partial feedback linearization and a linearized controller. The motivation lies in introducing controllers for underactuated pendulum-like systems that avoid switching between controllers. Indeed this continuous controller is formulated in a general way which can be applied to *class-I underactuated systems*.

1.4 Outline of the dissertation

Chapter 2 discusses passivity-based adaptive task space control of soft robots and its extension for bilateral teleoperation of soft robots. This chapter also introduces the simulation environment used for most of the numerical simulations presented in this dissertation. In Chapter 3, an enhanced dynamics model for planar soft robots is introduced, and passivity-based robust control for task space trajectory tracking of planar soft robots is discussed. Chapter 4 extends the enhanced dynamics model to 3D spatial soft robots, and a passivity-based robust task space bilateral teleoperation framework is discussed. An adaptive control framework for curvature tracking is introduced in Chapter 5 utilizing soft stretchable sensing skins and experimentally demonstrating successful curvature tracking control. In Chapter 6, a novel robotic system is introduced, coined as a rigid-soft hybrid robot composed of serially attached rigid and soft links. This robot would provide both the dexterity of soft robots and the precision and payload capacity of a rigid robot. It is demonstrated that the well-known passivity-based adaptive and robust control can be utilized for this class of robots. Chapter 7, introduces a soft inverted pendulum with a revolute base that could provide both scientific evidence and a prescribed methodology to introduce novel soft robots in several compelling applications in robotics and control education, agriculture, and health. An energy-based controller for swing-up and stabilization is discussed for the soft inverted pendulum system, illustrating the designed controller's efficacy using simulations. Moreover, in Chapter 8, a control architecture for the swing-up and stabilization of a class of underactuated mechanical systems is developed, which includes the soft inverted pendulum, by employing output partial feedback linearization and linear control. Chapter 9, adds the concluding remarks, providing a summary of the contributions, discussing the limitations, and noting possible future research directions. Additionally, some preliminaries and supplemental material are included in Appendix A.

Chapter 2: Adaptive Task Space Control of Soft Robots

In this chapter, a passivity-based adaptive task space control framework is developed for soft robots under the PCC hypothesis leveraging the *augmented formulation* [22]. Further, a bilateral teleoperation (leader-follower) framework for soft robots is also introduced. A simulation study and an experimental investigation of the proposed control algorithms on a planar soft robot are conducted and the results are discussed pointing out the important observations.

2.1 Overview

Dynamics models proposed for soft robots have been based on PCC formulation [19, 20, 21, 22], Cosserat rod theory [28] and Ritz-Galerkin models [32] among others [33, 60]. Using these available models, several control frameworks for soft robots have been developed [5, 22]. However, most of them require exact knowledge of the system parameters, which is hard to estimate accurately, especially in soft robots. One approach to mitigate the uncertainty in the parameters is to utilize adaptive control techniques [61]. In the literature, there have been several previous work on adaptive control of soft robots. A model reference adaptive control for soft robots [62] has been used in an open-loop inverse dynamic feedforward controller to follow reference trajectories in the degree of curvature space. A model-based sliding mode controller has been developed for curvature tracking [41, 63]. Recently, the authors in [44] proposed adaptive control for curvature tracking in 3D soft robots. However, these dynamic controllers are not designed to track task space trajectories. There has been a kinematic adaptive controller for soft robots proposed for task space trajectory tracking [64], but such kinematic controllers might fail under considerable dynamic effects.

This chapter proposes a passivity-based adaptive controller for task space trajectory tracking of soft robots considering a redundant inextensible soft robot under PCC hypothesis. The dynamic model of the soft robot is approximated to a rigid robot with elastic joints using the recently introduced *augmented formulation* [22]. Then the task space adaptive controller is developed using classical methods by following [61].

Further, as an extension, based on the adaptive task space controller, an adaptive task space bilateral teleoperation (ATSBT) framework for soft robots is proposed. Although the state of the art on control and motion planning of soft robots have grown recently, bilateral teleoperation of soft robot systems has not been addressed adequately [65]. On the other hand, for rigid link robotic systems bilateral teleoperation frameworks have been extensively studied, such as in systems with dissimilar kinematics [66], parameter uncertainty [67], constant / non-constant communication time delays [68, 69], etc. The redundancy of the follower robot has also been used to achieve secondary objectives such as singularity avoidance and collision avoidance while tracking the robot tip [67]. Among the very few work on teleoperation of soft robots, a telerobotic system for telesurgeries has been proposed in [70]. However, only a kinematic formulation was proposed. A task space teleoperation framework considering the dynamics for an extensible soft robot using feedback linearization was been proposed in [41, 71]. The authors also have utilized the null space velocity of the redundant follower robot to achieve singularity avoidance as a sub-task. A model based nonlinear control strategy was utilized to achieve asymptotic task space tracking between the leader device and the soft follower robots [72]. However, all these works did not include any feedback from the follower robot to the leader robot, and hence the robots were not controlled *bilaterally*. Furthermore, dynamic uncertainty was not considered in the literature on teleoperation of soft robots.

In the proposed ATSBT in this chapter, it is considered that the leader robot is a non-redundant rigid manipulator. The follower robot is considered to be a redundant inextensible soft manipulator under the PCC hypothesis. The end-effector is modeled as a lumped mass at the soft robot's tip. The ATSBT framework is introduced using classical control strategies developed for rigid robots. This dissertation demonstrates for the first time that passivity-based adaptive task space controllers [67] can be constructively utilized for *bilateral teleoperation* of a rigid leader robot and a soft follower robot. Consequently, the need for exact knowledge of the rigid leader or the soft follower robots' parameters is obviated. This is not only useful when the soft robot's model parameters cannot be exactly measured but also in the case when the inertial parameters of the end effector are not known *a priori*. Additionally, the null space velocity tracking is utilized to achieve sub-task goals such as conforming to degree of curvature limits in the soft robot. This work on bilateral teleoperation of soft robots was presented in [73].

The rest of the chapter is organized as follows. The *augmented formulation* of the soft robot is explained in Section 2.2. Then passivity-based adaptive control for task space trajectory tracking of soft robots is discussed in Section 2.3. The ATSBT framework for the soft robotic system is introduced in Section 2.4. In Section 2.5, the simulation environment is introduced and the simulation results are discussed. Finally, in Section 2.6, the experimental setup and the results for the proposed control frameworks for the soft robotic system are illustrated.

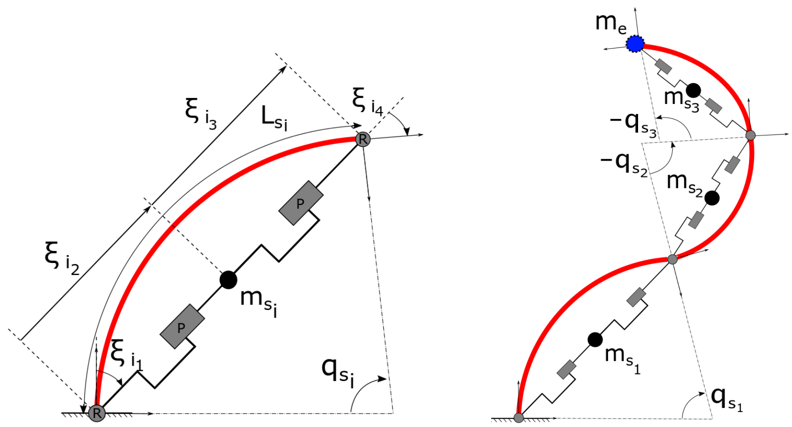
2.2 Augmented formulation of soft robots

This section briefly explains the mathematical model of the considered soft robot, which is a planar soft manipulator. The dynamics of the soft robot is formulated as a Lagrangian system through an approximated dynamically consistent *augmented formulation* using the methods introduced by Santina et al.,[21, 22]. The interested reader is referred to [22] for the complete derivation.

It is assumed that the soft robot is composed of n inextensible segments and each segment has a CC along the central axis of the segment which is time varying. To form the soft robot, these n CC segments are attached together in such a way that the resulting curve is differentiable everywhere. Under this hypothesis a single variable per segment, namely the degree of curvature, is sufficient to describe the segment's configuration in space. This model is known as the PCC model and is commonly used in soft robotics to reduce the infinite dimensionality of the soft robot [16].

The augmented robot model of this PCC robot is represented as a rigid link robot with a sequence of revolute and prismatic (R and P) joints with elasticity using the *augmented formulation* that connects the kinematic and approximated dynamic properties of the PCC soft robot to a conventional rigid robot system. Let the total number of joints per CC segment l and define the generalized coordinates for the PCC robot in this representation as $\xi \in \mathbb{R}^{nl}$ which is called as the *augmented configuration*. The continuously differentiable map, $\zeta : \mathbb{R}^n \rightarrow \mathbb{R}^{nl}$ connects the augmented configurations to the PCC robot model configurations (degree of curvature space).

Consider the i^{th} CC segment as shown in Figure 2.1a. Let us denote the degree of



(a) (b)

Figure 2.1: The PCC robot and the augmented robot as RPPR links are superimposed and shown here. The soft robot segments are shown in red. Panel (a) shows the i^{th} segment superimposed RPPR model and panel (b) illustrates the representation of a soft robot manipulator with three CC segments with an end effector of lumped mass m_e at the tip.

curvature as q_{si}, which would be the configuration variable of this segment. Under the hypothesis of inextensible segments, the length of the segment is assumed to be constant and is denoted by L_{si}. The mass, m_{si}, of the segment is assumed to be lumped at the center of the main chord connecting the ends of the CC segment.

In this work the augmented robot is represented as a sequence of $l = 4$ joints (RPPR) and is illustrated in Figure 2.1a. The augmented representation of this RPPR link representation is $\xi_i = [\xi_{i1},\ \xi_{i2},\ \xi_{i3},\ \xi_{i4}]^\top$ and the kinematically and dynamically consistent connection map is defined as,

$$\xi_i = \zeta(q_{si}) = [\frac{q_{si}}{2},\ L_{si}\frac{\sin(\frac{q_{si}}{2})}{q_{si}},\ L_{si}\frac{\sin(\frac{q_{si}}{2})}{q_{si}},\ \frac{q_{si}}{2}]^\top$$

As the PCC robot is a sequence of n such CC segments joined together, the complete configuration of the equivalent rigid robot model would be a sequence of RPPR links,

$$\xi = \zeta(q_s) = \left[\zeta(q_{s1})^\top,\ \zeta(q_{s2})^\top,\ \ldots,\ \zeta(q_{sn})^\top\right]^\top$$

Figure 2.1b illustrates a PCC robot comprised of three CC segments which also carries an end-effector approximated as a lumped mass at the end of the robot with mass m_e.

The dynamics of the augmented robot model including the end-effector is then found using Euler-Lagrangian methods for rigid robots [74] as,

$$M(\xi)\ddot{\xi} + C(\xi, \dot{\xi})\dot{\xi} + G(\xi) = \tau - J_\xi^\top F_{ext} \tag{2.1}$$

where $M(\xi) \in \mathbb{R}^{nl \times nl}$ is the inertia matrix, $C(\xi, \dot{\xi})\dot{\xi} \in \mathbb{R}^{nl}$ is the Coriolis and centrifugal terms matrix, $G(\xi) \in \mathbb{R}^{nl}$ is the gravity vector, $\tau \in \mathbb{R}^{nl}$ is the augmented control input vector. Here, the task space of the robot is considered to be $\mathbb{R}^p$ where $p \leq n$ and let the end-effector position of the robot to be $X_s(t, \xi)$. $F_{ext} \in \mathbb{R}^p$ is the external forces by the environment on the end-effector of the soft robot and are mapped through the Jacobian $J_\xi = \partial X_s(\xi)/\partial \xi$.

In order to propose control algorithms to the soft robot, one needs to derive the dynamics of the PCC robot evolving on $\xi = \zeta(q)$. This is done by calculating the augmented configuration derivatives $\xi, \dot{\xi}, \ddot{\xi}$ with regard to $q, \dot{q}, \ddot{q}$ as $\xi = \zeta(q)$, $\dot{\xi} = J_\zeta(q)\dot{q}$ and $\ddot{\xi} = \dot{J}_\zeta(q, \dot{q})\dot{q} + J_\zeta(q)\ddot{q}$ and substituting them in (2.1). Here $J_\zeta(q) \in \mathbb{R}^{mn \times n}$ represents the Jacobian $J_\zeta(q) = \partial\zeta(q)/\partial q$. Then the resulting equation is pre-multiplied with $J_\zeta^\top$ yielding the dynamic equation,

$$M_s(q)\ddot{q}_s + C_s(q_s, \dot{q}_s)\dot{q}_s + G_s(q_s) = \tau_s - J_s^T(q_s)F_{ext} \tag{2.2}$$

where,

$$
\begin{aligned}
M_s(q_s) &= J_\zeta^\top(q_s)\, M(\zeta(q_s))\, J_\zeta(q_s) \\
C_s(q_s, \dot{q}_s) &= J_\zeta^\top(q_s)\, M(\zeta(q_s))\, \dot{J}_\zeta(q_s, \dot{q}_s) + J_\zeta^\top(q_s)\, C(\zeta(q_s), J_\zeta(q_s)\dot{q}_s)\, J_\zeta(q_s) \\
G_s(q_s) &= J_\zeta^\top(q_s)\, G(\zeta(q_s)) \\
\tau_s &= J_\zeta^\top(q_s)\, \tau \\
J_s(q_s) &= J_\xi(\zeta(q_s)) J_\zeta(q_s)
\end{aligned}
$$

Finally, in order to incorporate the compliance of the soft robot, linear elastic and dissipative terms are introduced to the dynamic equation (2.2). Then, the complete dynamics of the soft robot in the Lagrangian form is derived as,

$$
M_s(q_s)\ddot{q}_s + C_s(q_s, \dot{q}_s)\dot{q}_s + D_s\dot{q}_s + K_s q_s + G_s(q_s) = \tau_s - J_s^T F_{ext} \tag{2.3}
$$

where, damping $D_s = \text{diag}(d_i) \in \mathbb{R}^{n\times n}$ and stiffness $K_s = \text{diag}(k_i) \in \mathbb{R}^{n\times n}$ are usually determined through system identification. Here diag$(\cdot)$ denotes the diagonal matrix. Specific properties of Lagrangian systems which are utilized in this dissertation is included in Appendix A.2 for completeness.

2.3 Passivity-based adaptive control for task space trajectory tracking

Consider the direct forward kinematics $h_s(\cdot) : \mathbb{R}^n \to \mathbb{R}^p$ which maps the configuration space to the task space $\mathbb{R}^p$. Thus, the end effector position and the velocity are defined as,

$$X_s(t) = h(q_s), \quad \dot{X}_s(t) = J_s(q_s)\dot{q}_s$$

where, $J_s(q_s) = \frac{\partial h_s(q)}{\partial q_s} \in \mathbb{R}^{p\times n}$ is the Jacobian matrix. Let $X_d(t)$ be the desired reference task space trajectory. Then, the tracking error is defined as,

$$e_s(t) = X_s(t) - X_d(t).$$

In order to drive the tracking error to approach origin, the trajectories of the system is restricted to the sliding surface,

$$s_s(t) = J_s^+(q_s)\dot{e}_s(t) + J_s^+(q_s)\Lambda e_s(t) \tag{2.4}$$

where, $\Lambda \in \mathbb{R}^{p\times p}$ is a positive definite gain matrix. Here, $J_s^+ \triangleq J_s^\top (J_s J_s^\top)^{-1} \in \mathbb{R}^{n\times p}$ is the pseudo inverse of J_s and satisfies the property $J_s J_s^+ = \mathbb{I}_n$ where $\mathbb{I}_n$ is the $n \times n$ identity matrix.

Note that when $s_s = 0$,

$$\dot{e}_s(t) = -J_s J_s^+ \Lambda e_s(t) + J(\mathbb{I}_n - J_s^+ J_s)\psi_s = -\Lambda e_s(t) \tag{2.5}$$

and hence, the errors will reach the origin when $s_s = 0$. Let us define signals,

$$v_s = \dot{q}_s - s_s$$

$$a_s = \ddot{q}_s - \dot{s}_s.$$

Assuming uncertainty in the parameters, let us use the notation $(\hat{\cdot})$ to denote the estimated values and $(\tilde{\cdot})$ to denote the estimation error. Using the linearity in parameters property (*Property A.3*) of Lagrangian systems for the system (2.3), one can define the regressor $(Y_s\,(q_s, \dot{q}_s, v_s, a_s))$ and parameter (Θ_s) vector pair for the estimated systems,

$$Y_s(q_s, \dot{q}_s, v_s, a_s)\hat{\Theta}_s = \hat{M}_s(q_s)a_s + \hat{C}_s(q_s, \dot{q}_s)v_s + \hat{D}_s v_s + \hat{K}_s q_s + \hat{G}_s(q_s). \tag{2.6}$$

Here time invariant uncertainties in the dynamic terms, stiffness, damping and actuator parameters are assumed. Thus, the parameter vector Θ_s is a constant.

Following the adaptive control approach [61], the control input for the soft robot is defined as,

$$\tau_s = Y_s(q_s, \dot{q}_s, v_s, a_s)\hat{\Theta}_s - K_0 s_s \tag{2.7}$$

where K_0 is a positive definite diagonal gain matrix. The closed loop system can be found by substituting the proposed control (2.7) in the soft robot dynamics (2.3), and using (2.6),

$$M_s(q_s)\dot{s}_s + C_s(q_s, \dot{q}_s)s_s + D_s s_s + K_0 s_s = Y_s(q_s, \dot{q}_s, v_s, a_s)\tilde{\Theta}_s \tag{2.8}$$

where $\tilde{\Theta}_s = \hat{\Theta}_s - \Theta_s$. Let the adaptation law for the parameter estimation defined as,

$$\dot{\hat{\Theta}}_s = -\Gamma_s Y_s^\top s_s \tag{2.9}$$

where Γ_s is a positive definite symmetric gain matrix that needs to be tuned.

Theorem 2.1 *Consider the closed loop system (2.8) with the parameter adaptation law (2.9) and sliding surface (2.4). In the absence of any external wrenches, the task space position error (e_s) and velocity error $(\dot{e}_s)$ asymptotically reach the origin while the parameter estimation error $(\tilde{\Theta}_s)$ remains bounded.*

Proof of Theorem 2.1 *Consider a Lyapunov like function for the system defined as,*

$$V = \frac{1}{2}\left(s_s^\top M_s s_s + \tilde{\Theta}_s^\top \Gamma_s^{-1} \tilde{\Theta}_s\right) \geq 0$$

Differentiating V with respect to time yields,

$$\begin{aligned}
\dot{V} =& \frac{1}{2} s_s^\top \dot{M}_s s_s + s_s^\top M_s \dot{s}_s + \tilde{\Theta}_s^\top \Gamma_s^{-1} \dot{\tilde{\Theta}}_s \\
=& \frac{1}{2} s_s^\top \dot{M}_s s_s + s_s^\top \left(-C_s s_s - D_s s_s - K_0 s_s + Y_s \tilde{\Theta}_s\right) + \dot{\tilde{\Theta}}_s^\top \Gamma_s^{-1} \tilde{\Theta}_s
\end{aligned}$$

Using the skew symmetry property (Property A.2) and using the chosen adaptation law (2.9) one can obtain,

$$\dot{V} = -s_s^\top D_s s_s - s_s^\top K_0 s_s \leq 0.$$

As $V \geq 0$ and $\dot{V} \leq 0$, $\lim_{t\to\infty} V$ is finite. Therefore, $s \in \mathcal{L}_2$ and $s_s, \tilde{\Theta}_s \in \mathcal{L}_\infty$. From (2.8), noting the properties of Lagrangian systems, it is observed that $\dot{s}_s \in \mathcal{L}_\infty$. Therefore, since $s_s \in \mathcal{L}_2$ and $\dot{s}_s \in \mathcal{L}_\infty$, one can show that $s_s \to 0$ as $t \to \infty$. Now from (2.5), $e_s, \dot{e}_s \to 0$ once $s = 0$.

2.4 Adaptive task space bilateral teleoperation framework

The proposed ATSBT framework is developed based on the Lagrangian formulation of the soft follower robot and the rigid leader robot as given by (2.3) and (2.10) assuming the leader is non-redundant and the follower is redundant. Similar methods as in [67] is followed in this dissertation to derive the framework.

2.4.1 The rigid manipulator : leader robot

This section introduces the mathematical model of the leader robot which is modeled as a rigid serial link manipulator with p links. In the absence of friction, the dynamics of the leader robot can be written in the usual Euler-Lagrangian form as [74],

$$M_m(q_m)\ddot{q}_m + C_m(q_m, \dot{q}_m)\dot{q}_m + G_m(q_m) = \tau_m + J_m^\top(q_m)F_h \tag{2.10}$$

where $M_m(q_m) \in \mathbb{R}^{p\times p}$ is the inertial matrix, $C_m(q_m, \dot{q}_m)\dot{q}_m \in \mathbb{R}^p$ collects the centrifugal and Coriolis terms with $C_m(q_m, \dot{q}_m) \in \mathbb{R}^{p\times p}$ expressed in Christoffel symbols and $G_m(q_m) \in \mathbb{R}^p$ is the gravitational force vector. $q_m \in \mathbb{R}^p$ is the generalized configuration variables which in this case are the relative joint angles and $\tau_m \in \mathbb{R}^p$ is the generalized force vector.

2.4.2 Control design

The ATSBT framework is designed in this section. Subscripts $j = m, s$ are used for concise representation where, subscript m refers to the rigid leader robot and subscript s refers to the soft follower robot. Similar to Section 2.3, consider the maps $h_m(\cdot) : \mathbb{R}^p \to \mathbb{R}^p$ and $h_s(\cdot) : \mathbb{R}^n \to \mathbb{R}^p$ which map the configuration spaces to the task space $\mathbb{R}^p$. In this work the direct forward kinematics of the robots are utilized as these maps and the robot tip positions and their velocities are defined as, $X_j = h_j(q_j)$ and $\dot{X}_j = J_j(q_j)\dot{q}_j$ respectively. Here, $J_j(q_j) = \frac{\partial h_j(q_j)}{\partial q_j}$ are the Jacobian matrices.

Following the passivity-based adaptive control approach [61], assuming time invariant uncertainties in the inertia and Coriolis/centrifugal terms, the control inputs for the rigid leader and soft follower robots are given as,

$$
\begin{aligned}
\tau_m =& \hat{M}_m(q_m)a_m + \hat{C}_m(q_m, \dot{q_m})v_m + \hat{G}_m(q_m) - K_m s_m - J_m^T \bar{\tau}_m \\
\tau_s =& \hat{M}_s(q_s)a_s + (\hat{C}_s(q_s, \dot{q}_s) + \hat{D}_s)v_s + \hat{K}_s q_s + \hat{G}_s(q_s) - K_0 s_s - J_s^T \bar{\tau}_s
\end{aligned}
\tag{2.11}
$$

where $\hat{(\cdot)}$ indicates the estimates for the corresponding terms. K_m and K_0 are positive definite diagonal matrices that needs to be tuned and $\bar{\tau}_j$ are the coordinating control. The signals a_j, v_j and s_j, and coordinating controls $\bar{\tau}_j$ are defined subsequently.

Using the properties of Lagrangian systems one can define the regressor-parameter vector pair (*Property A.3*) for the estimated systems,

$$\begin{aligned}\hat{M}_m(q_m)a_m + \hat{C}_m(q_m, \dot{q_m})v_m + \hat{G}_m(q_m) &= \quad Y_m(q_m, \dot{q}_m, v_m, a_m)\hat{\Theta}_m \\ \hat{M}_s(q_s)a_s + (\hat{C}_s(q_s, \dot{q}_s) + \hat{D}_s)v_s + \hat{K}_0 q_s + \hat{G}_s(q_s) &= \quad Y_s(q_s, \dot{q}_s, v_s, a_s)\hat{\Theta}_s\end{aligned}$$

The tracking errors for the leader system and soft follower system are defined as

$$e_m(t) = X_s(t - T_s) - X_m(t),$$

$$e_s(t) = X_m(t - T_m) - X_s(t).$$

Constant communication time delays of T_s and T_m are assumed here. The signals $a_m(t), v_m(t), s_m(t)$ for the leader and $a_s(t), v_s(t), s_s(t)$ for the follower are as follows (omitting the dependencies due to brevity):

$$\begin{aligned}s_m &= -J_m^{-1}\Lambda_m e_m + \dot{q}_m \\ v_m &= \dot{q}_m - s_m = J_m^{-1}\Lambda_m e_m \\ a_m &= \ddot{q}_m - \dot{s}_m = \dot{J}_m^{-1}\Lambda_m e_m + J_m^{-1}\Lambda_m \dot{e}_m \\ s_s &= -J_s^{+}\Lambda_s e_s + \dot{q}_s - (I_n - J_s^{+}J_s)\psi_s \\ v_s &= \dot{q}_s - s_s = J_s^{+}\Lambda_s e_s + (I_n - J_s^{+}J_s)\psi_s \\ a_s &= \ddot{q}_s - \dot{s}_s = \dot{J}_s^{+}\Lambda_s e_s + J_s^{+}\Lambda_s \dot{e}_s + \frac{d}{dt}[(I_n - J_s^{+}J_s)\psi_s]\end{aligned} \tag{2.12}$$

where Λ_j are properly chosen positive definite matrices and $\psi_s \in \mathbb{R}^p$ is the negative gradient of an appropriately defined convex function which is utilized for the sub-task control (see Appendix A.4). $J_s^+ \triangleq J_s^T(J_sJ_s^T)^{-1} \in \mathbb{R}^{n\times p}$ is the pseudo inverse of J_s and satisfies the property $J_sJ_s^+ = \mathbb{I}_p$. Here $\mathbb{I}_i$, for $i = n, p$ is the $i \times i$ identity matrix. The coordinating controls $\bar{\tau}_j$ are defined as,

$$\bar{\tau}_j = k_{r_j}(-\Lambda_j e_j + \dot{X}_j) - k_{J_j}\dot{e}_j \tag{2.13}$$

where k_{r_j} and k_{J_j} are positive gains that need to be chosen properly.

The closed loop dynamics of the system is found by substituting the proposed controls (2.11) to (2.3) and (2.10). Then, using the linear parameterizability of the Lagrangian dynamics (*Property A.3*),

$$\begin{aligned} M_m(q_m)\dot{s}_m + C_m(q_m,\dot{q}_m)s_m + K_m s_m &= Y_m(q_m,\dot{q}_m,v_m,a_m)\tilde{\Theta}_m - J_m^T\bar{\tau}_m + J_m^T F_h \\ M_s(q_s)\dot{s}_s + C_s(q_s,\dot{q}_s)s_s + D_s s_s + K_0 s_s &= Y_s(q_s,\dot{q}_s,v_s,a_s)\tilde{\Theta}_s - J_s^T\bar{\tau}_s - J_s^T F_{ext} \end{aligned} \tag{2.14}$$

where $\tilde{\Theta}_j = \hat{\Theta}_j - \Theta_j$. The adaptation law for the parameter estimation is defined as,

$$\begin{aligned} \dot{\hat{\Theta}}_m &= -\Gamma_m Y_m^T s_m, \\ \dot{\hat{\Theta}}_s &= -\Gamma_s Y_s^T s_s \end{aligned} \tag{2.15}$$

where Γ_m and Γ_s are positive definite symmetric gain matrices.

First, the free motion case is considered when there is no human operator force on the leader and there is no environmental force on the follower (i.e: $F_h = F_e = 0$). Following the proof of Theorem 3.1 of [67], assuming that J_m is full rank and operated under free motion of the closed loop teleoperation system (2.13) – (2.15), the tip of the soft follower robot asymptotically tracks the tip position and velocity of the rigid leader robot. i.e: $e_i, e_s \to 0$ and $\dot{e}_m, \dot{e}_s \to 0$ as $t \to \infty$.

Next, the case when the human operator exerts a force on the leader and/or the follower robot experiences environmental forces is considered. Here it is assumed that both the human operator and the remote (follower) environment are passive with respect to the inputs $F_h(t)$, $F_e(t)$ and outputs $r_m = J_m s_m$, $r_s = J_s s_s$. Hence, there exist constants $k_h,\ k_e \in R^+$ such that $-\int_0^t F_h^T(\sigma) r_m(\sigma) d\sigma \leqslant -k_h$ and $\int_0^t F_e^T(\sigma) r_e(\sigma) d\sigma \leqslant -k_e$.

Similarly to the free motion, assuming that J_m is full rank and the human operator and the remote environment are passive with respect to the inputs $F_h(t)$, $F_e(t)$ and outputs r_m, r_s it can be shown that the closed loop teleoperation system described by (2.13) – (2.15) will drive $e_i, e_s \to 0$ and $\dot{e}_m, \dot{e}_s \to 0$ as $t \to \infty$.

2.5 Simulation study

This section presents the simulation results for the proposed adaptive controller for soft robot task space trajectory tracking and the proposed bilateral teleoperation framework. First, the general simulation environment for the soft robot will be introduced and subsequently the simulated scenarios and the results are illustrated.

2.5.1 Simulation environment: simulating the soft robot

Since exact models of soft robots are hard to simulate, a FEM approach is utilized to simulate a planar soft manipulator. While, this model is a hyper-redundant system which lies outside the PCC hypothesis, it validates the control performance even outside the assumed PCC hypothesis.

A planar soft robot comprised of three actuated CC segments operating on the horizontal plane with a fixed base at the origin is considered in the simulations. All the three actuated segments of the soft robot are considered to be identical with a mass of $1kg$ and a length of $1m$. Each segment is discretized into six equal rigid links (each with mass $0.083kg$ and length $0.167m$) connected through linear torsional spring-dampers with parallel axes. The states of the FEM, $q_{sim}, \dot{q}_{sim} \in \mathbb{R}^{18}$ are mapped to the states of the soft robot model $q_s, \dot{q}_s \in \mathbb{R}^3$ by summing the corresponding states of the finite elements in one segment as,

$$q_s = \left[\sum_{j=1}^{6} q_{sim_j} \ , \ \sum_{j=7}^{12} q_{sim_j} \ , \ \sum_{j=13}^{18} q_{sim_j} \right]^\top$$
$$\dot{q}_s = \left[\sum_{j=1}^{6} \dot{q}_{sim_j} \ , \ \sum_{j=7}^{12} \dot{q}_{sim_j} \ , \ \sum_{j=13}^{18} \dot{q}_{sim_j} \right]^\top .$$

For a particular CC segment, the FEM is actuated providing the same torque input for all the joints in the FEM of that segment. This results in an actuation mapping between the FEM actuation $\tau_{sim} \in \mathbb{R}^{18}$ and the soft robot actuation $\tau_s \in \mathbb{R}^3$ defined as $\tau_{sim} = A\tau_s$. Here the transmission matrix $A \in \mathbb{R}^{18\times3}$ defined as,

$$A = \begin{bmatrix} \mathbb{I}_{6,1} & \mathbb{O}_{6,1} & \mathbb{O}_{6,1} \\ \mathbb{O}_{6,1} & \mathbb{I}_{6,1} & \mathbb{O}_{6,1} \\ \mathbb{O}_{6,1} & \mathbb{O}_{6,1} & \mathbb{I}_{6,1} \end{bmatrix}$$

where $\mathbb{I}_{6,1}$ is the 6×1 unity vector ($[1, 1, 1, 1, 1, 1]^\top$) and $\mathbb{O}_{6,1}$ is the 6×1 null vector ($[0, 0, 0, 0, 0, 0]^\top$). Each torsional spring-damper in the FEM is considered to have a stiffness of 3 Nm/rad and a damping of 1.2 Nms/rad. Thus, assuming identical stiffness and damping in the CC segments, the torsional damping of each CC segment is found out to be $d_i = d = 0.2$ Nm/rad resulting in the systems damping to be $D_s = 0.2\ I_{3\times3}$ Nms/rad and torsional stiffness is $k_i = k = 0.5$ Nm/rad resulting in the systems stiffness to be $K_s = 0.5\ I_{3\times3}$ Nm/rad. The Robotics System Toolbox in Simulink is used to simulate the FEM of the soft robot.

2.5.2 Simulation results for adaptive task space trajectory tracking

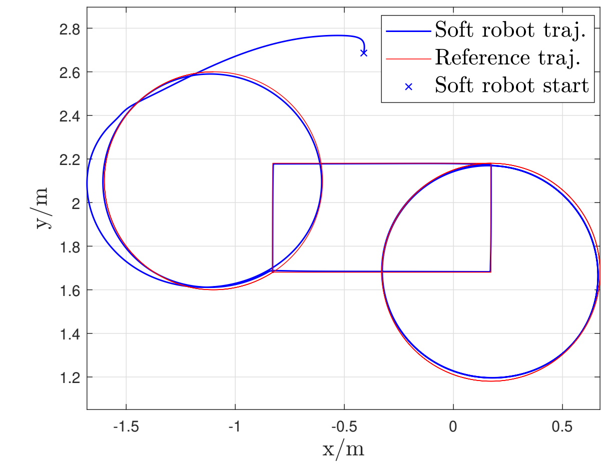
In this simulation study, the uncertainty was assumed in the soft robot's segment masses (m_i), torsional stiffness (k_i) and torsional damping (d_i). Note that, the end effector is omitted in the dynamics used for this simulation. Thus the parameter vector was chosen as $\Theta_s = [m_1, m_2, m_3, k, d]^\top$ and the initial estimates for the parameters were set to $\hat{\Theta}_s(0) = [1.2, 0.8, 1.3, 0.1, 0.1]^\top$ which are different from the nominal values. The control gains were set to $\Gamma = 0.75$, $\Lambda = 7$ and $K_0 = 2$. The soft robot was initialized at $q_s(0) = [-0.15,\ 0.1,\ 1.5]^\top$rad and $\dot{q}_s(0) = [0,\ 0,\ 0]^\top$rad/s. A predetermined reference trajectory $(X_d(t))$ as shown in Table 2.1 was used over a period of $100s$ for task space trajectory tracking.

Table 2.1: Reference trajectory for task space trajectory tracking used in the simulation study

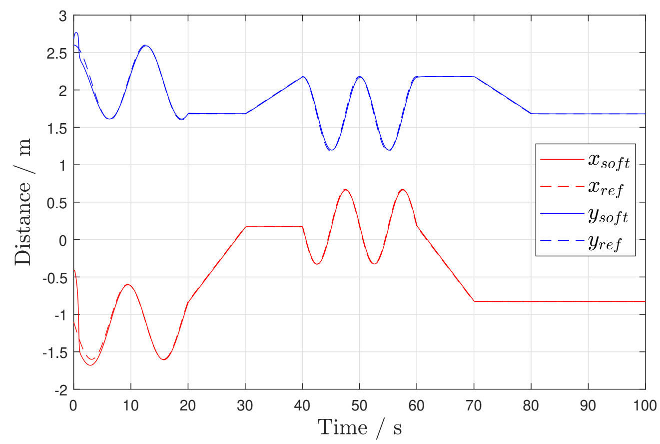
Time / s	Reference trajectory ($X_d(t)$) / m
$0 \leq t < 20$	$[-1.1 - 0.5\sin(t/2)\ ,\ 2.1 + 0.5\cos(t/2)]^\top$
$20 \leq t < 30$	$[0.1t - 3.1 - 0.5\sin(10)\ ,\ 2.1 + 0.5\cos(10)]^\top$
$30 \leq t < 40$	$[-0.1 - 0.5\sin(10)\ ,\ 0.05t + 0.6 + 0.5\cos(10)]^\top$
$40 \leq t < 60$	$[-0.1 - 0.5\left(\sin(10) - \sin(\frac{(t-40)\pi}{5})\right)\ ,\ 2.1 + 0.5\left(\cos(10) + \cos(\frac{(t-40)\pi}{5})\right)]^\top$
$60 \leq t < 70$	$[-0.1t + 5.9 - 0.5\sin(10)\ ,\ 2.6 + 0.5\cos(10)]^\top$
$70 \leq t < 80$	$[-1.1 - 0.5\sin(10)\ ,\ -0.05t + 6.1 + 0.5\cos(10)]^\top$
$80 \leq t \leq 100$	$[-1.1 - 0.5\sin(10)\ ,\ 2.1 + 0.5\cos(10)]^\top$

The trace of the soft robot's tip trajectory along with the reference trajectory in the task space is illustrated in Figure 2.2a. Considerably high tracking errors over the initial $10s$ as seen in Figure 2.2b might be due to the fact that the parameters are being updated and has not converged. However, as the time progresses the tracking error approaches zero. Figure 2.2c illustrates the control torques under the proposed adaptive controller. Sudden spikes are observed initially as well as in instances when the reference trajectory suddenly moves direction. Although such control signals are possible in a simulation environment, in actual physical soft robot models there might be actuation limits, thus inhibiting the performance of the controller. To investigate such behavior, the proposed controller is tested in an experimental study (Section 2.6.2).

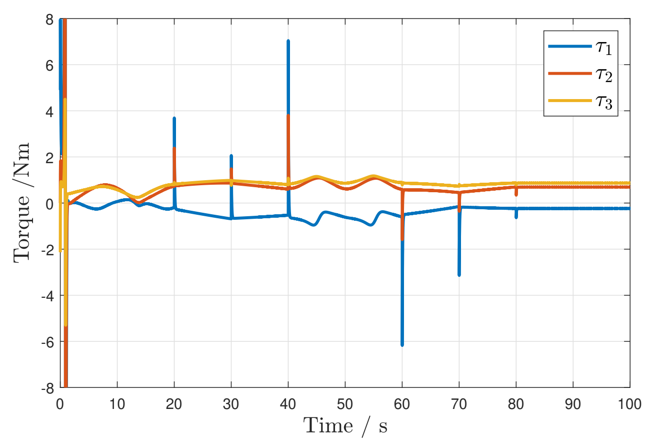
The evolution of parameter estimation is illustrated in Figure 2.3. Here, it is observed that the parameters do not converge to their actual values, except for the stiffness value. Intuitively, the convergence of the stiffness value to the actual value is required for steady state stability. However, it can be seen that the other parameter values remain bounded, as expected.



(a) Trace of the task space trajectory



(b) Tracking performance



(c) Control inputs

Figure 2.2: Simulation results for the trajectory tracking performance of the proposed adaptive controller.

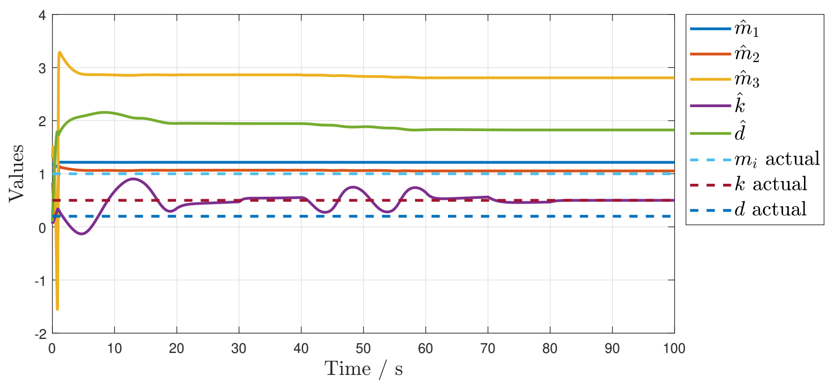


Figure 2.3: Simulation results for parameter adaptation of the system parameters.

2.5.3 Simulation results for ATSBT

This section presents the simulation results demonstrating the performance of the proposed bilateral teleoperation framework. The rigid leader robot was composed of two links with $[1.55, 1.45]^{\top}$m length and each weighing 1kg. The follower robot was a three segment soft robot with the same parameters as described above in Section 2.5.1. The control gains were constant throughout the simulations and were set to $\Gamma_m = \Gamma_s = 0.75$ $\Lambda_m = \Lambda_s = 10$, $K_{rm} = K_{rm} = 1$, $K_{jm} = K_{js} = 1$ and $K_m = K_0 = 1$. The human force and environmental force were modeled as spring-damper forces. The spring and damper gains for the human force was set to 50 N/m and 50 Ns/m. Thus, the exerted human force is calculated as $F_h(t) = 50(X_{desired} - X_m(t)) - 50\dot{X}_m(t)$, where $X_{desired}$ is the desired location of the leader robot to be moved to. Those gains for the environmental force is 1050 N/m and 10 Ns/m resulting in exerted environmental force $F_h(t) = 1050(X_{wall} - X_m(t)) - 10\dot{X}_m(t)$ with X_{wall} is the location of the environment obstacle. Considering the parameterization on the dynamics, the rigid leader was parameterized using the minimum parameters as shown in [74] (see Appendix A.3) and the soft follower was parameterized with respect to the masses of the segments and the end-effector, the stiffness and damping as $\Theta_s = [m_1, m_2, m_3, m_e, k, d]^{\top}$. In the simulations, the initial parameter estimates were selected to be $\Theta_m(0) = [1, 1, 1]^T$ and $\Theta_s(0) = [1, 1, 1, 1, 1, 1]^T$.

2.5.3.1 Scenario with no environmental force

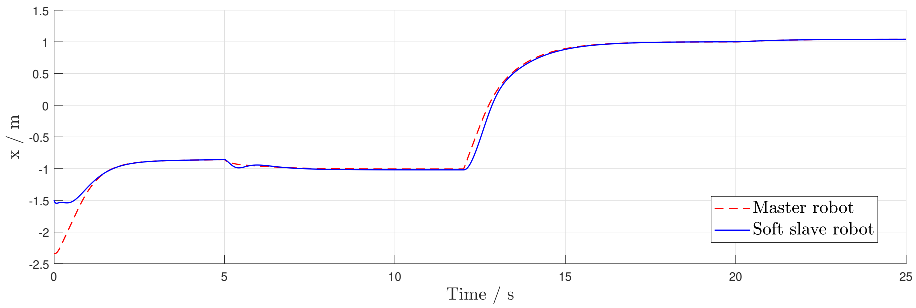
In this simulation study, the remote environment was free from any obstacles, thus the follower robot was allowed to move freely. The initial conditions for the rigid leader robot was $q_m(0) = [\frac{\pi}{4}, \frac{\pi}{4}]^T$, $\dot{q}_m = [0, 0]^T$ and that for the soft follower robot was $q_s(0) = [\frac{\pi}{4}, -\frac{\pi}{4}, \frac{\pi}{6}]^T$,

$\dot{q}_s = [0, 0, 0]^T$. The task space synchronization of the ATSBT with sub-task control for degree of curvature limits is illustrated in Figure 2.4. The end-effector mass is 0.05kg for this simulation. Initially no human force is exerted before 5s. At $t = 5$s the human operator moves the leader to $X_m = [-1, 1]^T$ exerting a force as described above. At $t = 12$s the human operator moves the leader to $X_m = [1, 2]^T$. At $t = 20$s the system is set to operate in free motion. Figure 2.5 illustrates the degree of curvature evolution for this case. Figure 2.6 and Figure 2.7 illustrates the same experiment with no sub task control.

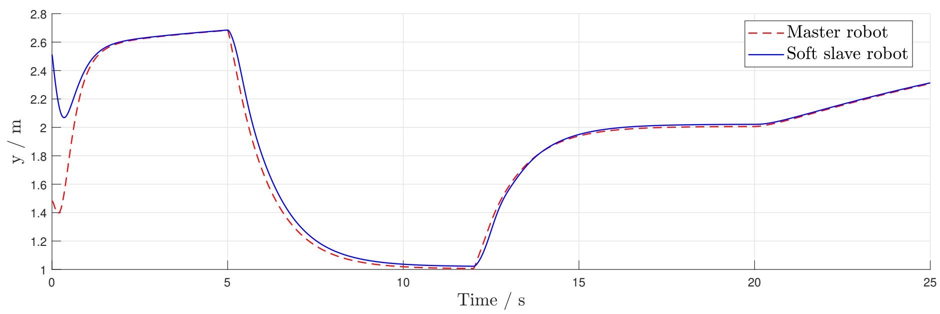
2.5.3.2 Scenario with environmental interaction

In this simulation, a wall was assumed to be present at $x = 0$ in the remote environment. Figure 2.8 depicts the task space teleoperation with sub-task control of curvature limits of $q_{s2} = [-\pi, \pi]^\top$ and $q_{s3} = [-\frac{\pi}{2}, \frac{\pi}{2}]^\top$. The initial conditions for the leader and follower were $q_m(0) = [1.8, 1.0]^\top$, $\dot{q}_m(0) = [0, 0]^T$, $q_s(0) = [2.5, -2.0, -1.65]^\top$ and $\dot{q}_s(0) = [0, 0, 0]^\top$. Initially, 0-5s is in free motion. Then, at $t = 5$s the human operator moves the leader to $X_m = [1.5, 1.5]^\top$ exerting a force as described above. At $t = 12$s the human operator tries to move the leader to $X_m = [-1, 1]^\top$ and at $t = 16$s to $X_m = [1, 1]^\top$. Finally, at $t = 24$s the system is set to operate again in free motion.

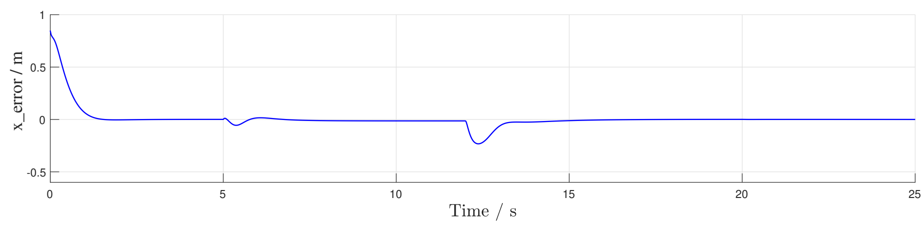
At around $t = 13$s the soft follower robot contacts the wall and the consequences of error not converging and the force feedback at the leader robot side is clearly seen in Figure 2.8c illustrating the performance of the proposed bilateral teleoperation framework.



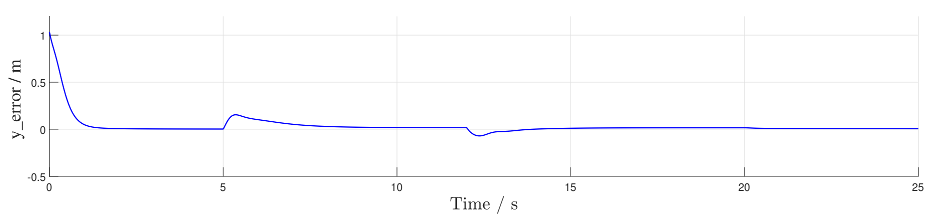
a) Synchronization in x direction.



b) Synchronization in y direction.



c) Synchronization error in x direction.



d) Synchronization error in y direction.

Figure 2.4: Simulation results for the ATSBT framework without environmental force with subtask control for degree of curvature limits.

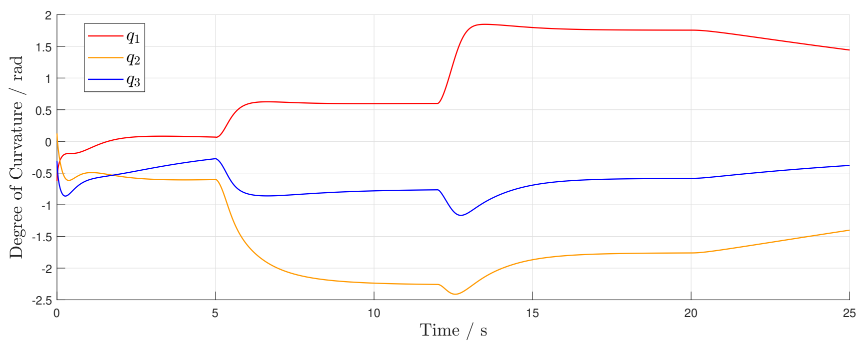
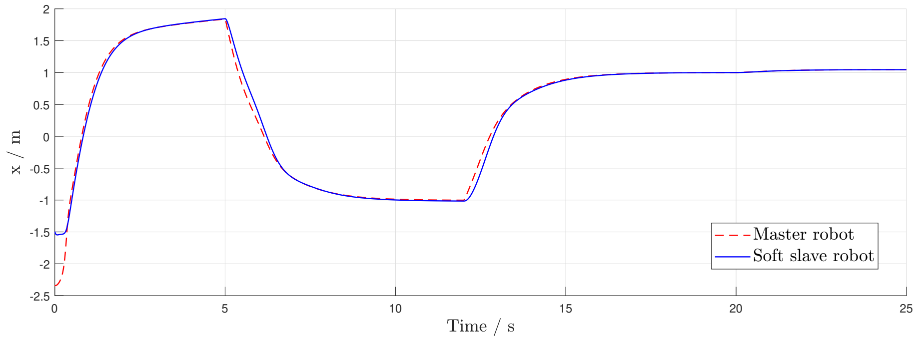
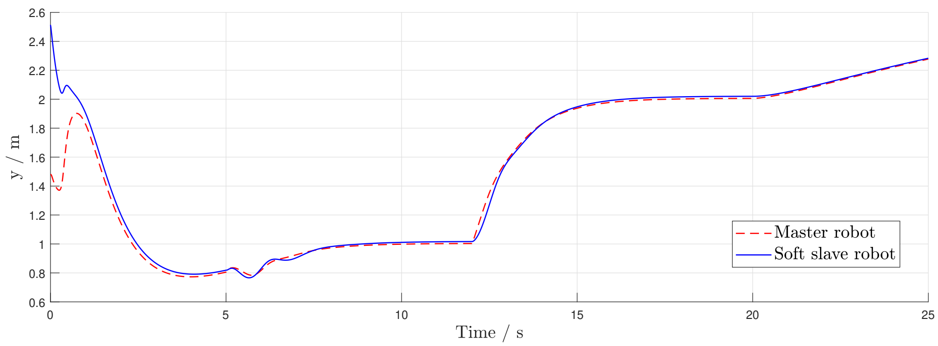


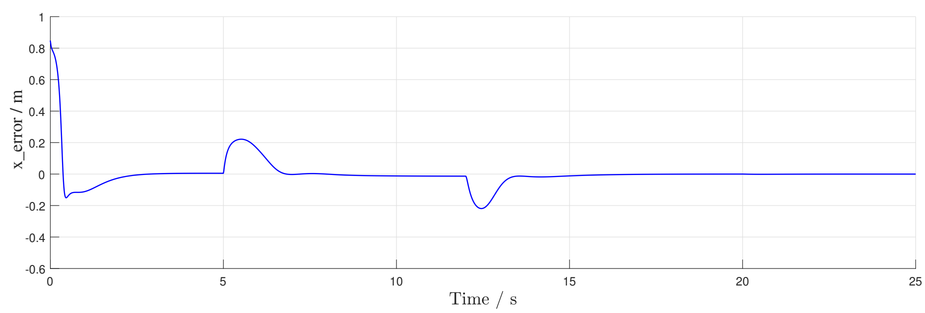
Figure 2.5: Simulation results for the ATSBT framework without environmental force illustrating the degree of curvature evolution with sub-task control for degree of curvature limits.



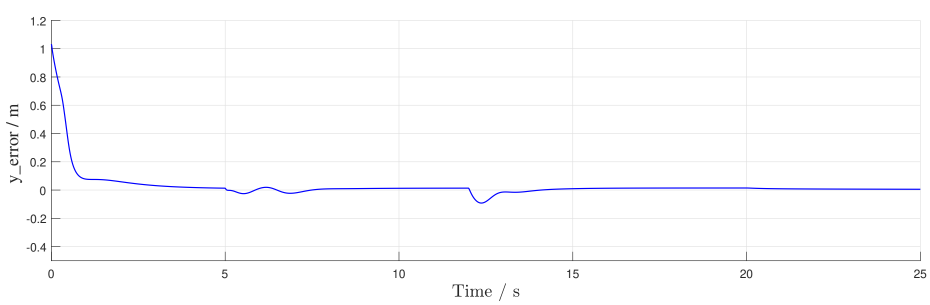
a) Synchronization in x direction.



b) Synchronization in y direction.



c) Synchronization error in x direction.



d) Synchronization error in y direction.

Figure 2.6: Simulation results for the ATSBT framework without environmental force and without sub-task control.

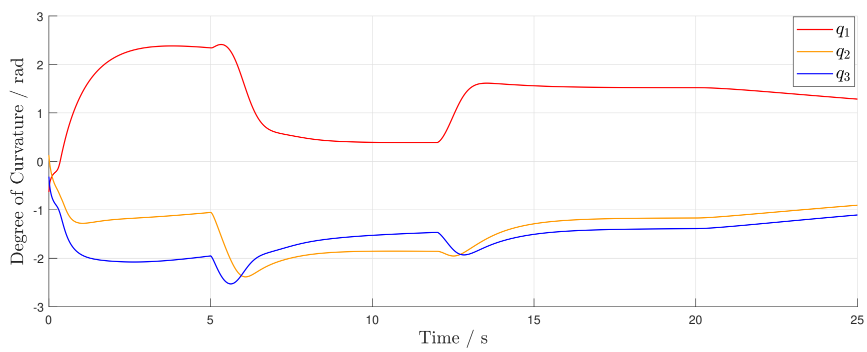
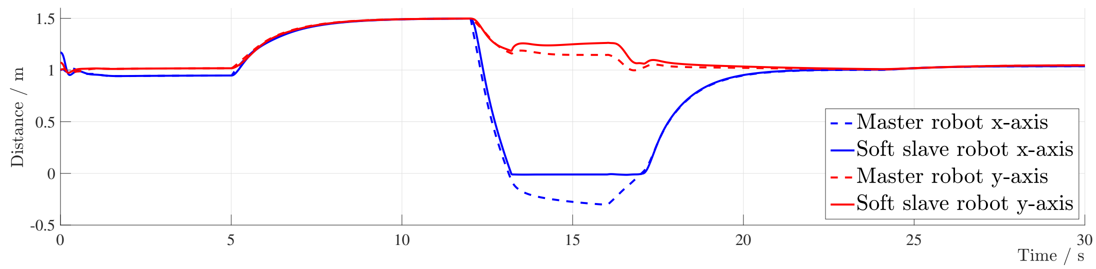
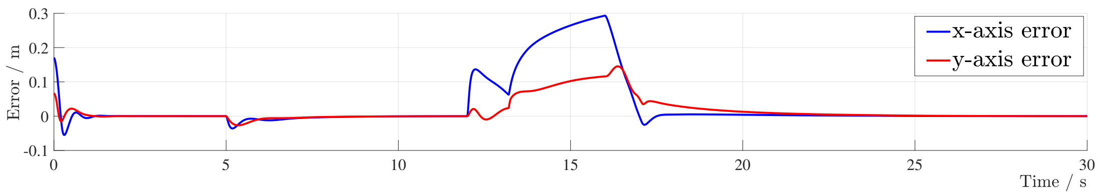


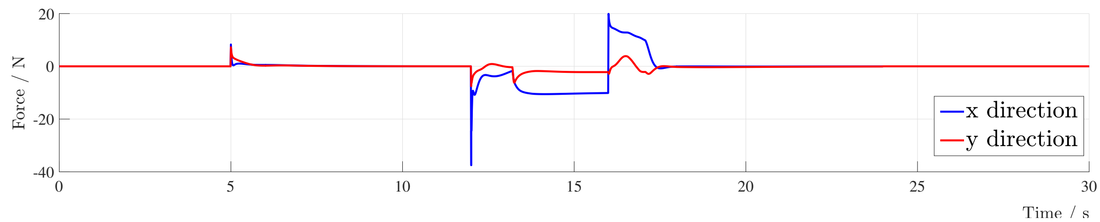
Figure 2.7: Simulation results for the ATSBT framework without environmental force illustrating the degree of curvature evolution without sub-task control.



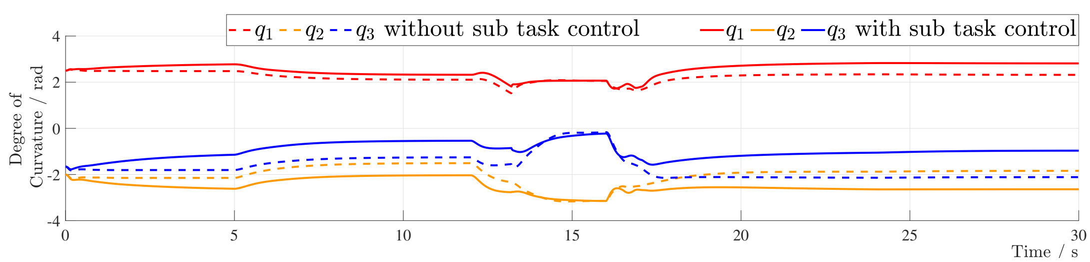
(a) Robot tip positions



(b) Follower robot tip errors



(c) Forces on leader side



(d) Evolution of degree of curvature of segments with and without sub-task control

Figure 2.8: Illustration of the simulation results for ATSBT with environmental force and with sub-task control for degree of curvature limits.

2.6 Experimental investigation

This section presents the experimental results demonstrating the performance of the proposed bilateral teleoperation framework. First the experimental setup is described. Subsequently, the experimental results of the bilateral teleoperation scenarios are presented and discussed.

2.6.1 Experimental setup

The experimental setup shown in Figure 2.9 consists of a pneumatically actuated pleated type soft robot [75], OptiTrack motion capture system and two i7 16GM RAM Windows 10 laptops: one running the control algorithms on Matlab 2015 and the other simulating the rigid leader robot for the bilateral teleoperation framework only.

2.6.1.1 The Soft Robot

The soft robot used in the experiments, as shown in Figure 2.10, comprises of three bi-directional pleated type segments ($n = 3$) which were fabricated following methods outlined in [75]. The robot was constrained to move on a horizontal table and ball transfers were used underneath near the segment joints to reduce friction.

Each segment has two *compartments* that are individually actuated pneumatically. The segments are assumed to deform with a constant curvature under the applied pressure. The *middle layer* of each segment (the joint between the two compartments) is inextensible due to the restrained material layer. The segment lengths along the inextensible middle layer were measured to be $L_{s_i} = 0.125$ m. The segment masses $m_{s_i} = 0.110$ kg were measured prior to

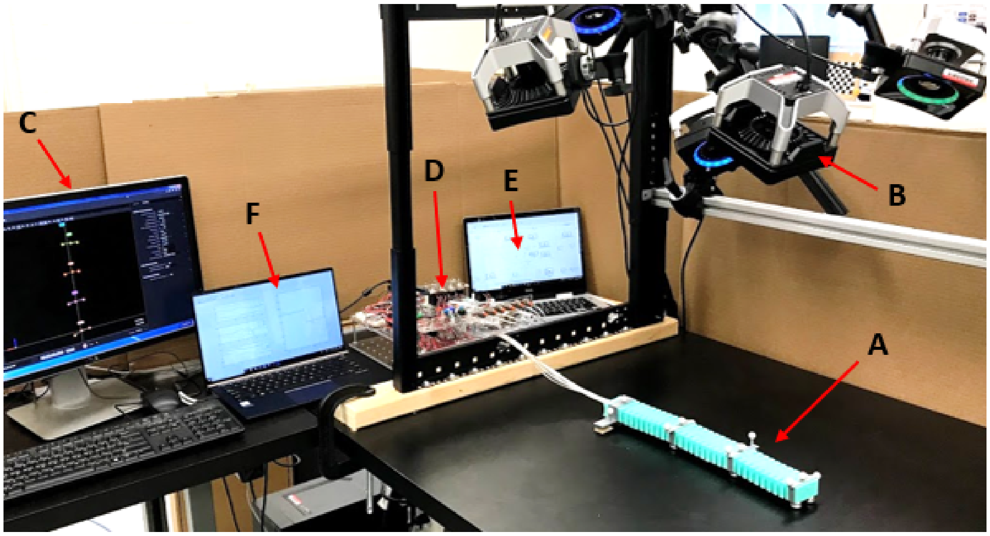


Figure 2.9: The experimental setup consisting of A-soft robot, B-OptiTrack cameras for motion capturing, C-PC for streaming motion capture data, D-pneumatic control unit, and E-laptop simulating the leader robot and F-laptop running the main control algorithms.

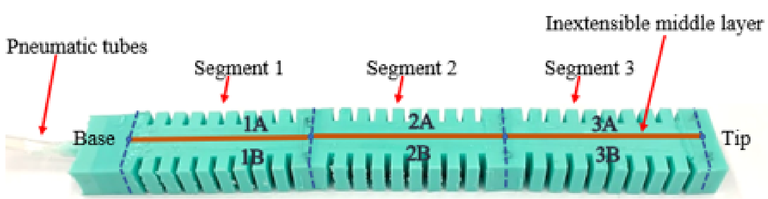
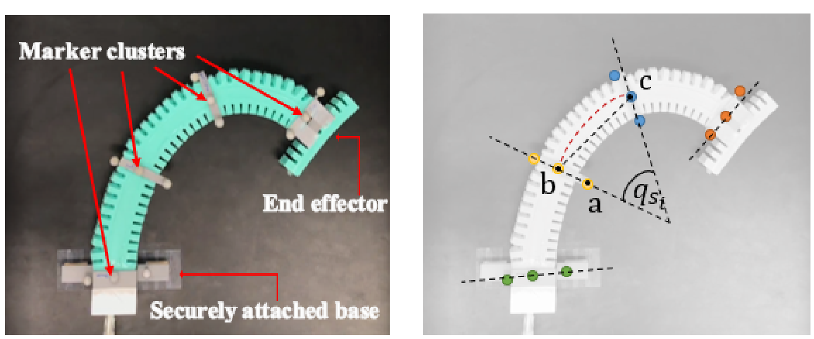


Figure 2.10: The pleated type soft robot.



(a) (b)

Figure 2.11: Positioning of the markers is shown in panel (a). Panel (b) illustrates the degree of curvature of the mid segment as seen using the motion capture system.

joining the segments together. The material properties of each segment were assumed to be identical. Therefore, for all the segments identical torsional stiffness of $k_i = k$ and damping of $d_i = d$ were assumed. The identification of the nominal values of k and d will be discussed subsequently. As seen in Figure 2.11a, an unactuated end effector resembling a soft gripper of mass $m_e = 0.050$ kg was attached to the tip of the soft robot (gripper was not actuated).

2.6.1.2 Degree of Curvature Estimation using Motion Capture

The positions of the end points of each segment was extracted using a Motion Capture (MoCap) system (OptiTrack) by attaching clusters of markers on the segment ends as shown in Figure 2.11a. The base of the soft robot was securely attached to the table so that experiments could be conducted without re-calibration. The degree of curvature of each segment was calculated using the properties of the dot product of the labeled marker positions as illustrated in Figure 2.11b:

$$q_{s_i} = 2\left(\frac{\pi}{2} - \cos^{-1}\left(\frac{(a-b)\cdot(c-b)}{||a-b||\;||c-b||}\right)\right).$$

These labeled markers were assumed to be lying on the horizontal plane throughout the trial.

2.6.1.3 The Actuation Unit

The soft robot was actuated using a pneumatic controller unit based on the open source hardware platform [76] with command inputs serially transmitted to the control board (Arduino Mega). The compressed air to the unit was supplied by an external compressor at a constant pressure of 20psi. The air pressure in the segments was regulated by Pulse-Width Modulation

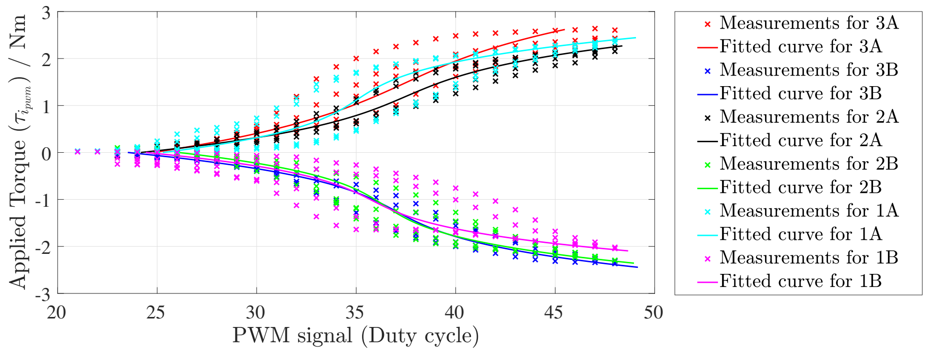


Figure 2.12: Relationship between the PWM signal and the applied torque

(PWM) using a frequency of 100Hz. The control input calculated by the controller, in terms of a *torque*, was converted to a PWM signal for each segment using corresponding mappings. At a given time instance, only one compartment out of the two in each segment is actuated. A positive torque commanded one compartment while a negative torque commanded the opposite compartment of the same segment.

The torque-to-PWM signals mapping was identified by a curve fitting process for each compartment of the three segments. Considering the dynamic model of the soft robot (2.3), for the i^{th} segment, a step input of $\tau_{s_i} = \tau_{\text{pwm}}$ resulted in a steady state represented by $\tau_{i_{\text{pwm}}} = k\,\theta_{i_{\text{pwm}}}$. Here $\theta_{i_{\text{pwm}}}$ is the steady state degree of curvature of the considered i^{th} segment and k is the torsional stiffness.

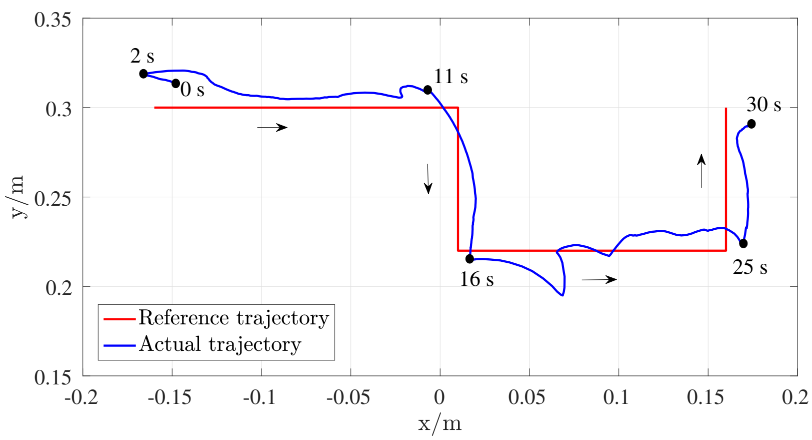
Sending commands to only one compartment of a segment, the steady state degree of curvatures ($\theta_{i_{\text{pwm}}}$) were recorded for different PWM signal values. Then the equivalent torque being applied, $\tau_{i_{\text{pwm}}}$, was calculated hypothesizing the torsional stiffness of each segment to be $k = 1$ Nm/rad. Finally, a third order polynomial curve fit was performed using this data for each of the segments to obtain the mapping from torque to PWM signal. Figure 2.12 illustrates the relationship between the PWM signals and the equivalent torque being applied for the individual compartments of the three segments. The nominal value of the torsional damping of a segment $d = 0.2$ Nms/rad was calculated using a system identification process by applying a unit torque to each of the segment.

2.6.2 Experimental results for adaptive task space trajectory tracking

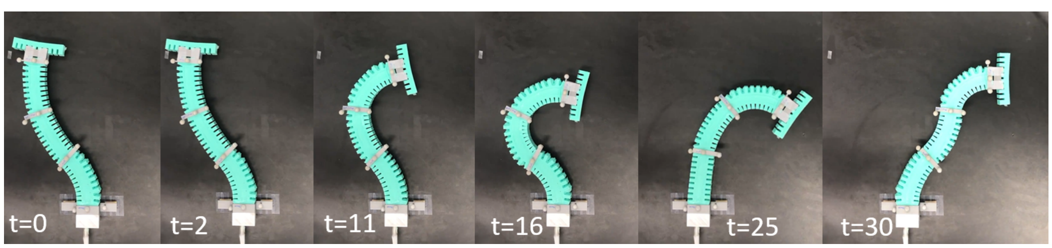
In this experimental study, the uncertainty was assumed in the soft robot's segment masses (m_i), end- effector mass (m_e) torsional stiffness (k_i) and torsional damping (d_i). Thus the parameter vector was chosen as $\Theta_s = [m_1, m_2, m_3, m_e, k, d]^\top$ and the initial estimates for the parameters were set to $\hat{\Theta}_s(0) = [0.6, 0.6, 0.6, 0.05, 1, 0.2]^\top$. The control gains were set to $\Gamma = 0.075$, $\Lambda = 3.5$ and $K_0 = 0.3$. A predetermined reference trajectory as shown in Table 2.2 was used over a period of 30s for task space trajectory tracking. Figure 2.13a shows the task space trajectories of the reference and the actual paths. The configurations of the soft robot during the experiment is shown in Figure 2.13b.

2.6.3 Experimental results for ATSBT

The performance of the ATSBT framework is evaluated experimentally in this section. In these experiments, the fabricated soft robot served as the follower device and a rigid manipulator which was simulated due to the unavailability of such a robot served as the leader device. The simulation was done using Matlab/Simulink 2019a in real time using the Simulink Desktop Real-Time library block on a separate laptop (see Figure 2.9). The communication between the two laptops, i.e. between the leader and follower robots was achieved through the TCP/IP protocol over WiFi. The network time delays were measured to be $T_m = T_s = 0.004s$.



(a) Task space trajectories



(b) Sequenced photos of the task space trajectory tracking

Figure 2.13: Experimental results for task space trajectory tracking

Table 2.2: Reference trajectory for task space trajectory tracking used in the experimental study

Time / s	Reference trajectory ($X_r(t)$) / m
$0 \le t < 2$	$[-0.16\ ,\ 0.3]^\top$
$2 \le t < 10.5$	$[-0.16 + 0.02t, 0.3]^\top$
$10.5 \le t < 11$	$[0.01, 0.3]^\top$
$11 \le t < 15$	$[0.01, 0.3 - 0.02t]^\top$
$15 \le t < 16$	$[0.01, 0.22]^\top$
$16 \le t < 23.5$	$[0.01 + 0.02t, 0.22]^\top$
$23.5 \le t < 25$	$[0.15, 0.22]^\top$
$25 \le t < 29$	$[0.15, 0.22 + 0.02t]^\top$
$29 \le t \le 30$	$[0.15, 0.3]^\top$

The simulated leader robot comprised of a 2-DoF rigid planar elbow robot and was modeled according to [74] (see Appendix A.3). The link lengths were chosen as $L_m = [0.1895, 0.1895]^\top$m so that the leader robot and the soft follower robot has a similar task space. The masses and inertia were chosen as $m_m = [0.2, 0.2]^\top$kg and $I_m = [0.003, 0.003]^\top$kgm^2 respectively. The human force was modeled as a spring-damper forces with spring and damper gains set to 150 N/m and 150 Ns/m respectively.

For the soft follower robot, the uncertainty was assumed in the segment masses, end effector mass, torsional stiffness and torsional damping. Thus the parameter vector was chosen as $\Theta_s = [m_{s_1}, m_{s_2}, m_{s_3}, m_e, k, d]^\top$. The initial parameter estimates were set as $\hat{\Theta}_s(0) = [0.5, 0.5, 0.5, 0.01, 2, 1]^\top$ which were different from the measured nominal values. The control gains were constant throughout the experiments and were set to $\Gamma_s = 0.075$, $\Lambda_s = 3.5$, $K_{rs} = 0.3$, $K_{js} = 10$ and $K_0 = 0.3$.

The uncertainty in the rigid leader robot was assumed to be in the rigid robot's link masses and in the link inertia. Thus the parameter vector was chosen as $\Theta_m = [m_{m_1}, m_{m_2}, I_{m_1}, I_{m_2}]^\top$. The initial parameter estimates were $\hat{\Theta}_m(0) = [0.05, 0.05, 0.001, 0.001]^\top$. The control gains were constant throughout the experiments and were set to $\Gamma_m = 0.75$, $\Lambda_m = 5$, $K_{rm} = 1$, $K_{jm} = 12$ and $K_m = 1$.

2.6.3.1 Scenario with no environmental force

Here the soft follower robot was allowed to move freely with no environmental forces being applied. The sub-task control of conforming to degree of curvature limits (see Appendix A.4) of $q_{s_i} = [\frac{-5\pi}{9}, \frac{5\pi}{9}]^\top$ is considered. In the experiment, $t = 0$-5.5s is in free motion for both the robots to synchronize. Then, from $t = 5.5$-16.5s (phase 1.1) the human operator exerts a force to move the leader toward $X_m = [0.001, 0.245]^\top$. From $t = 16.5$-27s (phase 1.2) the human operator tries to move the leader to $X_m = [0.2, 0.2]^\top$ and from $t = 27$-35s (phase 1.3) to $X_m = [0.25, 0.001]^\top$. At $t = 35$s the system is set to operate again in free motion.

Figure 2.14 illustrates the results of this scenario without and with the sub-task control. Figure 2.15 depicts the configurations of the soft robot during the experiment. It is clearly seen from the sequenced photographs that the sub-task control forces the degree of curvatures of the segments to be within the desired limits by altering the configuration of the soft robot.

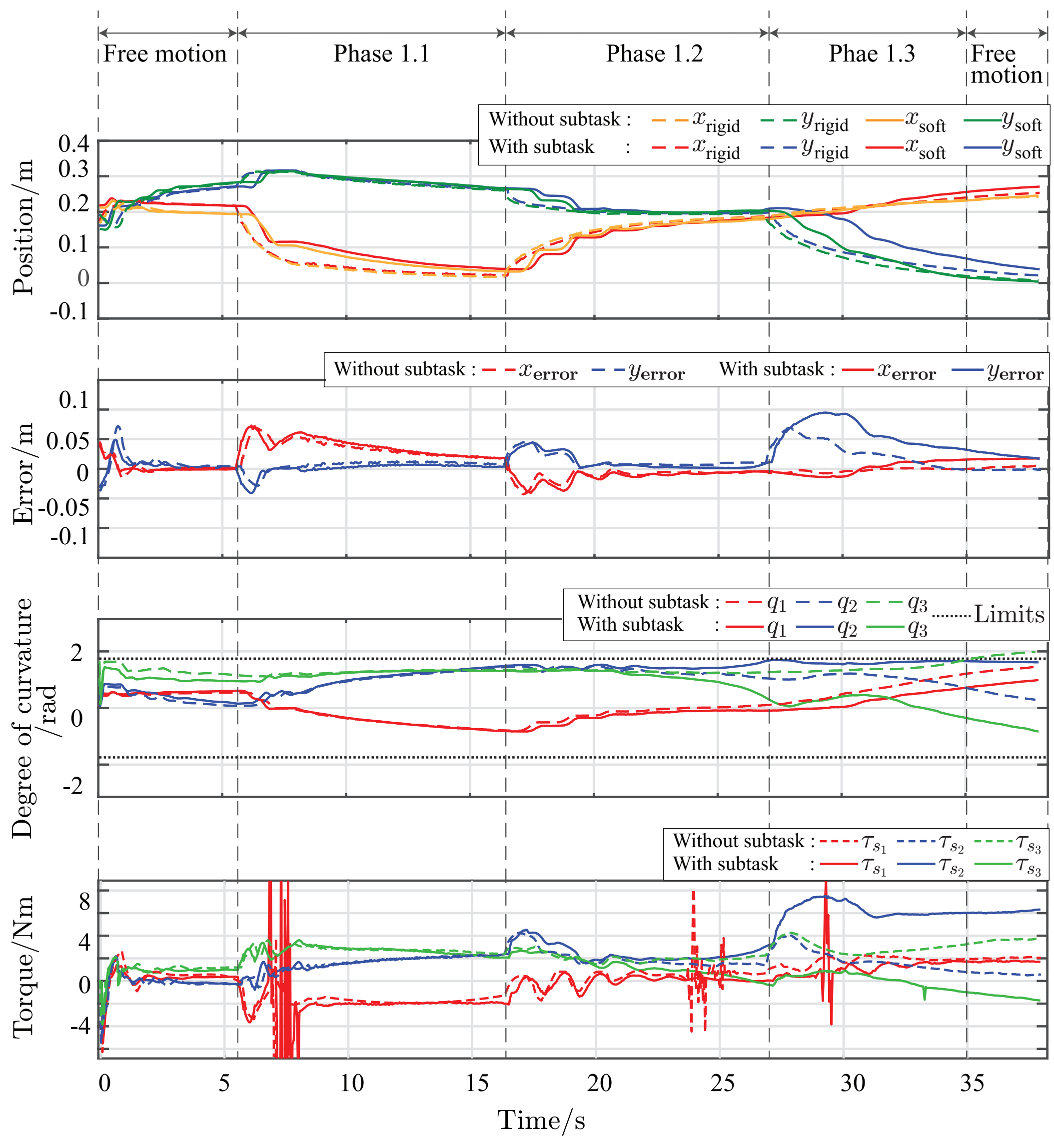
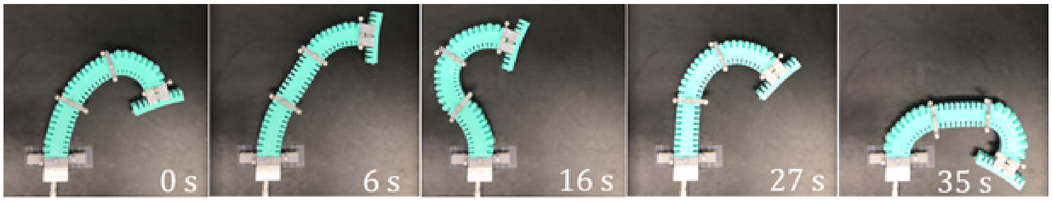
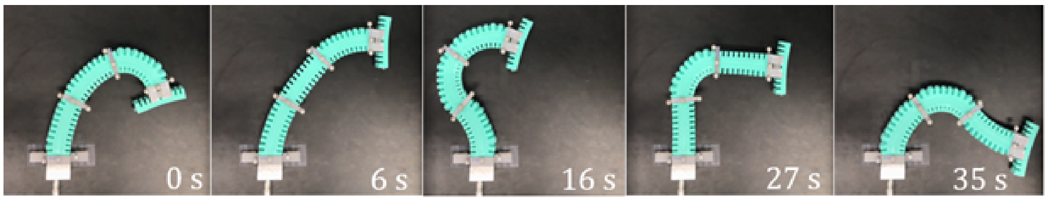


Figure 2.14: Experimental results for the ATSBT without and with subtask control of conforming to degree of curvature limits for the scenario with no environmental force.



(a) without the subtask control



(b) with the subtask control

Figure 2.15: Sequenced photographs of the bilateral teleoperation scenario with no environmental force.

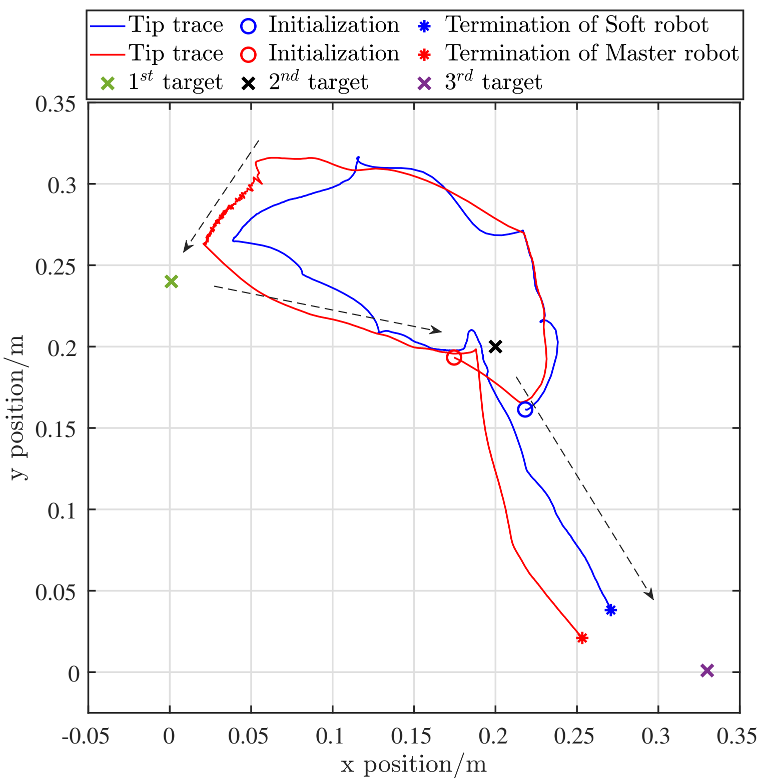


Figure 2.16: Trace of the robots' tips for the ATSBT experiment with sub-task control of conforming to degree of curvature limits for the scenario with no environmental force.

2.6.3.2 Scenario with the soft follower interacting with an obstacle

In this scenario an obstacle of the form of a wall at $x = 0.05$ m appears during the trial at $t = 15$s and the trial was conducted without attaching an end effector. In the initial 0 - 5.5s the robots operate in free motion. From $t = 5.5$-15.5s (phase 2.1) the human operator moves the leader toward $X_m = [0.001, 0.245]^\top$. From $t = 15.5$-21.5s (phase 2.2) the human operator tries to move the leader to $X_m = [0.2, 0.2]^\top$. The soft follower robot contacts the obstacle at around $t = 16.5$s. Since the soft robot cannot move past the obstacle, the position errors between the two robot tips do not approach zero. Therefore, the human operator continuously tries to push toward the desired point despite the inability to move it any further. This is clearly illustrated in Figure 2.17 bottom figure showing that the environmental forces experienced by the soft robot at the follower environment are reflected onto the leader side. From $t = 21.5$-29.5s (phase 2.3) the human operator moves the leader robot tip to the opposite direction of the obstacle, toward $X_m = [0.01, 0.33]^\top$. Now, it is observed that the tracking errors approach zero and the exerted forces diminish. At $t = 29.5$s the system is set in free motion. Figure 2.17 presents the results of this trial and Figure 2.18 illustrates the configurations of the soft robot and the obstacle. The force feedback at the leader robot side is clearly seen in this experiment illustrating the performance of the proposed bilateral teleoperation framework.

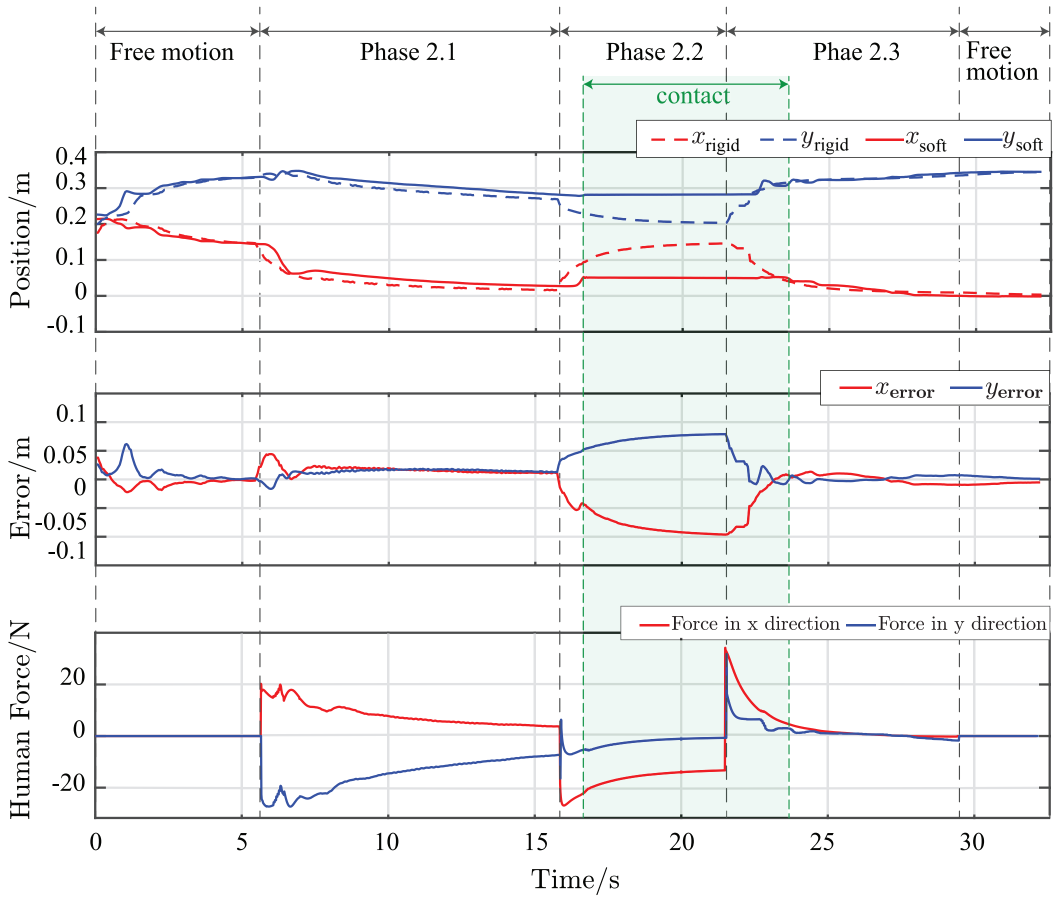


Figure 2.17: The results of the ATSBT involving obstacle contact in the remote(follower) environment. The forces experienced by the human is shown in the bottom plot.

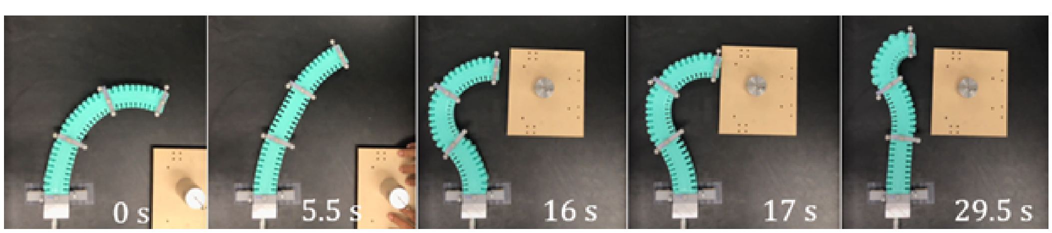


Figure 2.18: Sequenced photographs of the ATSBT scenario with the soft follower interacting with an obstacle.

2.6.4 Discussion

The goal of the experimental investigation was to study the performance of the proposed passivity-based task space controller and the ATSBT framework for a pneumatically actuated soft robot. It should be noted that the reference trajectories and the target positions for all the trials were carefully considered in order for the soft robot to be able to operate in the task space spanned by $q_{s_i} = [\frac{-5\pi}{9}, \frac{5\pi}{9}]^\top$ for all the segments so that the sub task control of conforming to degree of curvature limits of $q_{s_i} = [\frac{-5\pi}{9}, \frac{5\pi}{9}]^\top$ can be utilized.

In the experiments, although the results were satisfactory, the slow movement of the soft follower robot resulted in the follower robot not reaching the target reference positions in the allotted time interval. However, this observed result illustrates the critical need for bilateral teleoperation as the human operator is *immersed* in the remote follower environment and is able to respond to changes in the operational scenario.

The slow movement of the follower robot may be due to the *slow response* of the pneumatic actuation and also due to the *hysteresis effect* in the segments. Also spikes can be observed in the input where the large torques that are generated due to the *local singularities* of the segments.

Furthermore, although the stiffness of the segments were assumed to be constant, they may be changing with deformation. This time varying nature of the parameters are not handled by the passivity-based adaptive control and it can impact the performance as well as the stability of the bilateral teleoperation system. The frictional effects were neglected in the soft robot model as well as in the control algorithms. However, there might have been a significant friction between the soft robot and the surface despite the addition of ball transfers.

2.7 Summary

In this chapter, a task space adaptive control framework developed for PCC soft robots leveraging the *augmented formulation* was presented. The adaptive controller guaranteed task space trajectory tracking under time invariant parameter uncertainties. Further, an adaptive bilateral teleoperation framework for task space synchronization for a system consisting of a non-redundant rigid leader manipulator and a redundant PCC soft follower manipulator was developed. The redundancy in the soft follower manipulator was exploited to achieve sub-task objectives such as conforming to curvature limits while tracking the position of the leader robot. The simulations and the experimental study demonstrated good task space tracking performance and immersion in the remote environment.

Chapter 3: Robust Task Space Control of Soft Robots

This chapter investigates the classical passivity-based robust control approach for task space trajectory tracking of planar PCC soft robots. The developed controller leverages an enhanced dynamics model for planar multi-link soft robots that considers the mass distribution of the robot. The efficacy of the modeling and control framework is demonstrated using numerical simulations and physical experiments.

3.1 Overview

As discussed in Chapter 2, most of the controllers developed for soft robots assume perfect knowledge of the parameters. However, accurate parameter estimation in soft robots is hard. To overcome this the researchers have recently developed adaptive and robust controllers for soft robot curvature tracking [44, 45] and task space tracking [77, 78], getting inspiration from the classical rigid robotics community [61, 79, 80, 81]. In this chapter passivity-based robust control ideas [80] have been applied for soft robots for task space trajectory tracking.

The recent modeling approach proposed by the authors in [22], called the *augmented formulation*, which models a soft robot as if it was a rigid robot using an augmented rigid formulation, has opened avenues to implement and test traditional control algorithms that were developed for rigid robots. Indeed, this model was utilized in developing controllers in Chapter

2. However, this model considers lumped masses; hence, it is not capturing the total effects due to the soft body's inertia. In [82], the authors proposed a dynamics model for variable length multisection continuum arms assuming a uniform linear density for the segments. Although this method captures the inertia effects under the linear density assumption, it does not allow the mass distribution across the width of the links. Recently, a modeling approach that considers the mass distribution along the length and width of a soft link under affine curvature was introduced in [35]. However, this approach was only limited to a single link soft robot. This chapter proposes an enhanced dynamics model for PCC soft robots that considers the soft robot's mass distribution along its body by considering the length and width of the soft robot links. This modeling approach circumvents the inaccuracies caused by the lumped mass models that do not consider the total inertial effects of the links.

The main contribution of this chapter is to demonstrate the theoretical and experimental realization of robust trajectory tracking for planar PCC multi-link soft robots. The proposed framework encapsulates the body's total inertial effects by considering the soft robot's mass distribution along its length and width. Using numerical simulations and experiments on a multi-link soft robot, the efficacy of the proposed control framework is demonstrated.

The rest of the chapter is organized as follows. The enhanced dynamics model is introduced for a planar inextensible PCC soft robot in Section 3.2. The passivity-based robust controller for task space tracking is developed in Section 3.3 leveraging the proposed enhanced dynamics model. The simulation results are presented in Section 3.4 and the experimental results in Section 3.5.

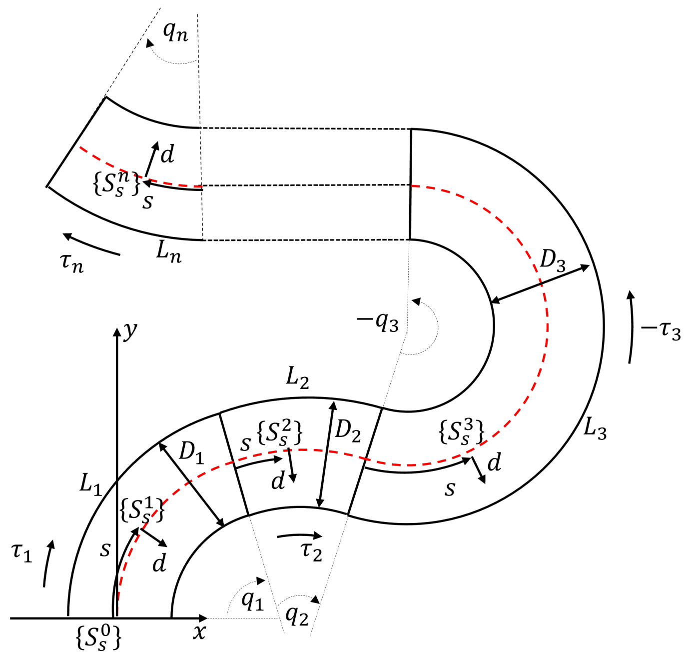


Figure 3.1: A planar PCC soft robot with n links is illustrated. The inextensible middle layer is colored red.

3.2 Planar distributed mass soft robot model

This section introduces the kinematics and the dynamics of a planar inextensible soft robot and considers the case wherein the base of the soft robot is fixed. First, the kinematics of the soft robot is discussed and subsequently the dynamics model is introduced, which explicitly takes into account the mass distribution of the planar soft robot. Note that since this section is considering a planar soft robot, the task space has a dimension of two.

3.2.1 Kinematics

In this work, the soft robot is consideed to be PCC [16]. Thus, the soft robot is assumed to be composed of n inextensible segments and each segment is assumed to have a time-varying CC along the length of the segment. To constitute the soft robot, these n CC segments are attached so that the resulting curve is differentiable everywhere and is on a plane. Under this hypothesis, a single variable per segment, namely the degree of curvature, is sufficient to describe the segment's configuration in space. Let $q(t) = [q_1(t), q_2(t), ..., q_n(t)]^\top$ be the curvatures of the soft segments.

The soft robot is oriented so that when all the links are straight, the robot aligns with the y-axis. As shown in Figure 3.1, let the length of the i^{th} segment along its central axis be L_i and its thickness D_i. The length of each segment is parameterized by $s \in [0, 1]$, such that sL_i is the arc length along the segment's central axis to the point s from the i^{th} segment's base. Moreover, the width of the segments is parameterized by $d \in [-0.5, 0.5]$ such that dD_i is the lateral distance to point d on the i^{th} segment from the central axis. At each point s for any segment i along the soft robot's body, reference frames $\{S_s^i\}$, with the superscript i denoting the considered segment are attached. The base frame, $\{S_0\}$, is fixed in space.

The orientation $\alpha_s^i(t)$ of the reference frame $\{S_s^i\}$ at a point s along the central axis on the i^{th} segment with reference to the base frame $\{S_0\}$, can be written as the sum of integral of the curvature,

$$\alpha_s^i(t) = \sum_{k=1}^{i-1} q_k(t) + \int_0^s q_i(t)\mathrm{d}l = \sum_{k=1}^{i-1} q_k(t) + q_i(t)s.$$

Thus the Cartesian coordinates $\left(x_{s,d}^i(t), y_{s,d}^i(t)\right)$ at a general point on the i^{th} segment parameterized by (s, d) on the soft robot is given by,

$$\begin{aligned}
x_{s,d}^i(t) &= \sum_{k=1}^{i-1} \left(L_k \int_0^1 \sin \alpha_s^k(t) \mathrm{d}s \right) - dD_i \cos \alpha_s^i(t) + L_i \int_0^s \sin \alpha_s^i(t) \mathrm{d}s \\
y_{s,d}^i(t) &= \sum_{k=1}^{i-1} \left(L_k \int_0^1 \cos \alpha_s^k(t) \mathrm{d}s \right) - dD_i \sin \alpha_s^i(t) + L_i \int_0^s \cos \alpha_s^i(t) \mathrm{d}s.
\end{aligned}$$

Define the direct forward kinematics $h(q) : \mathbb{R}^n \to \mathbb{R}^2$ which maps the configuration space to the task space,

$$h(q) = \left[x^n_{1,0}\ ,\ y^n_{1,0}\right]^\top$$

Thus, the end effector position and the velocity are defined as, $X(t) = h(q)$ and $\dot{X}(t) = J(q)\dot{q}$, where, $J(q) = \frac{\partial h(q)}{\partial q} \in \mathbb{R}^{2\times n}$ is the Jacobian matrix.

3.2.2 Dynamics

This section derives the dynamics of the soft robot based on the Euler-Lagrange (E-L) formalism. It should be noted that in this dynamics derivation, I do not assume any simplifying assumptions on the mass distribution, such as uniform linear mass density [82] or concentrated masses at discrete points [22, 43]. Instead, a normalized mass distribution $\rho_i(s, d)$ for each segment i is considered. In the following, where obvious, the time arguments are suppressed in the expressions due to brevity and clear representation.

Following the E-L method, the inertia matrix $M(q) \in \mathbb{R}^{n\times n}$ is evaluated as,

$$M(q) = \sum_{i=1}^{n} \int_0^1 \int_{-0.5}^{0.5} \rho_i(s, d)\nabla_q(x^i_{s,d}, y^i_{s,d})^\top \nabla_q(x^i_{s,d}, y^i_{s,d})\mathrm{d}d\mathrm{d}s$$

where ∇_q is the gradient operator $\nabla_q(\cdot) = \frac{\partial(\cdot)}{\partial q}$. Then the centrifugal and Coriolis terms matrix $C(q, \dot{q}) \in \mathbb{R}^{n\times n}$ is evaluated using the standard Christoffel symbols [74]. The gravitational potential energy $P_g \in \mathbb{R}$ of the soft robot is calculated by,

$$P_g = \sum_{i=1}^{n} \left(\int_0^1 \int_{-0.5}^{0.5} m_i g(x_{s,d}^i \sin(\phi) + y_{s,d}^i \cos(\phi)) \mathrm{d}d\mathrm{d}s \right)$$

where ϕ defines the direction of the gravitational field. Therefore the gravity terms vector $\mathbf{G}(q_0, \theta) \in \mathbb{R}^n$ is evaluated as,

$$G(q) = \nabla_q (P_g) .$$

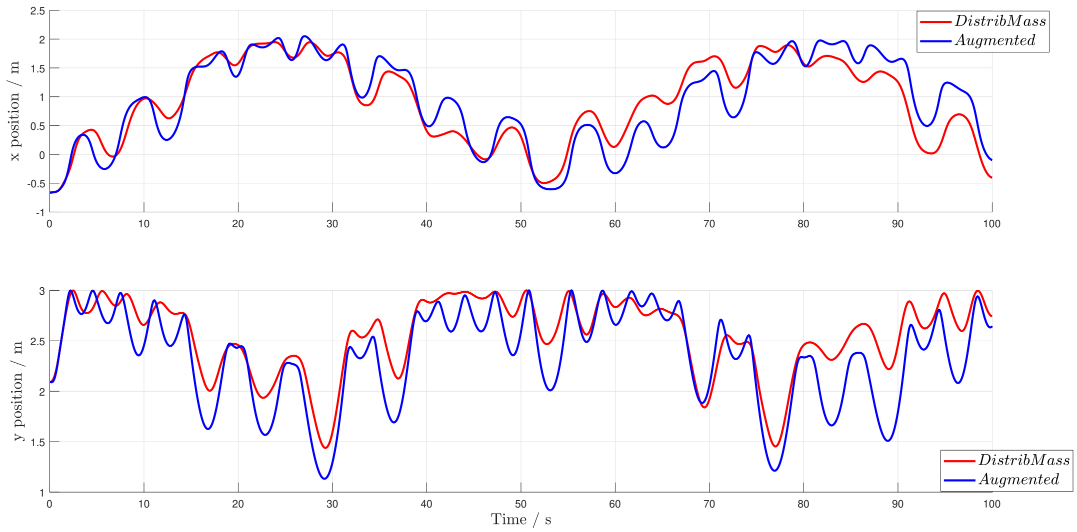
In this work, the elasticity of the soft robot is modeled through a continuous and homogeneous distribution of infinitesimal springs and dampers along the length of the soft segments. Assuming a linear relationship for elastic parameters, the stiffness and damping terms are Kq and $B\dot{q}$, respectively. Here the coefficient of stiffness $K \in \mathbb{R}^{n \times n}$ and damping $B \in \mathbb{R}^{n \times n}$ are diagonal matrices with the stiffness k_i and damping β_i of the i^{th} segment as the diagonal elements respectively.

Finally, adding the stiffness and damping terms the complete dynamics are,

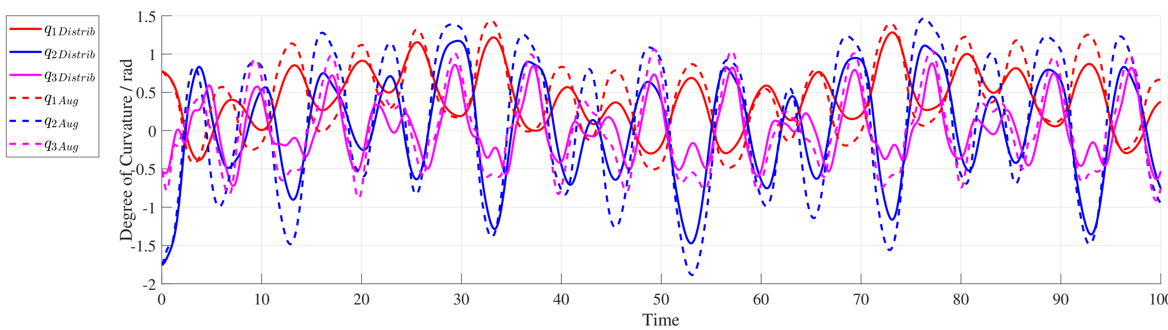
$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + B\dot{q} + Kq + G(q) = \tau \tag{3.1}$$

Here $\tau \in \mathbb{R}^n$ is the control input and physically it is the effective torque applied on the soft links causing them to bend with a CC.

Note that the soft robot dynamics in (3.1) is in the Lagrangian form, and hence it encapsulates the properties of Lagrangian systems such as the skew symmetric property and the linearly parameterizable property [74] as noted in Appendix A.2.



a) Evolution of the soft robot tip positions



b) Evolution of the soft robot's segment curvatures

Figure 3.2: Illustrating the difference in the forward dynamics of the proposed distributed mass dynamics model when compared with the *augmented formulation*. The input was $\tau = [0.2 + 0.2\sin(0.2\pi t),\ 0.2\cos(0.3\pi t),\ 0.15\sin(0.5\pi t)]^\top$.

Figure 3.2 illustrates the open loop behavior of a three-link planar PCC soft robot modeled using the proposed distributed mass approach and the *augmented formulation* [22] for a sinusoidal input torque. It is clearly seen from the evolution of the soft robot tip and the individual curvatures that there are considerable effects due to the soft robot's mass distribution, which the *augmented formulation* does not capture.

3.3 Passivity-based robust control for task space trajectory tracking

This section presents the passivity-based robust controller for task space trajectory tracking of planer PCC soft robots. Leveraging the soft robot dynamics as given in (3.1) and following the well-known passivity-based robust control approach in [80], this chapter shapes it to task space trajectory tracking inspired by the works in [67, 73]. Note that, for conciseness, a soft robot operating on the horizontal plane is considered, thus ignoring gravity effects. However, the ideas can be readily extended to soft robots under gravity, as long as the PCC assumption is valid.

Let $X_d(t) \in \mathbb{R}^2$ be the desired task space trajectory. Then, the tracking error is defined as,

$$e(t) = X(t) - X_d(t).$$

The objective of task space trajectory tracking is to drive $e(t), \dot{e}(t) \to 0$. To that end, the tracking error is restricted to thr sliding surface [67],

$$s(t) = J^{+}(q)\dot{e}(t) + J^{+}(q)\Lambda e(t) \tag{3.2}$$

where, $\Lambda \in \mathbb{R}^{2\times 2}$ is a positive definite gain matrix and $\mathbb{I}_n$ is the $n \times n$ identity matrix. Here, $J^{+} \triangleq J^{\top}(JJ^{\top})^{-1} \in \mathbb{R}^{n\times 2}$ is the pseudo inverse of J and satisfies the property $JJ^{+} = \mathbb{I}_2$. It is seen that once the trajectories reach $s \equiv 0$,

$$\dot{e}(t) = -JJ^{+}\Lambda e(t) = -\Lambda e(t)$$

and hence, the errors will reach the origin exponentially when $s \equiv 0$.

Let us define signals, $v = \dot{q} - s$ and $a = \ddot{q} - \dot{s}$. The notation $\hat{(\cdot)}$ is used to denote the estimated values. Using the linearity in parameters property (*Property A.3*) of Lagrangian systems for the system (3.1), the regressor $(Y\,(q, \dot{q}, v, a))$ and the parameter (Θ) vector pair for the estimated systems can be defined,

$$Y(q, \dot{q}, v, a)\hat{\Theta} = \hat{M}(q)a + \hat{C}(q, \dot{q})v + \hat{B}v + \hat{K}q + \hat{G}(q).$$

Here, time invariant uncertainties in the mass, stiffness and damping are assumed. The estimated parameter vector $\hat{\Theta}$ can be written as,

$$\hat{\Theta} = \Theta_0 + u \tag{3.3}$$

where Θ_0 is a fixed nominal parameter vector and u is an additional control term which will be designed for achieving robustness for uncertainty in the model parameters.

The control input τ is defined as,

$$\tau = Y(q, \dot{q}, v, a)\hat{\Theta} - K_s s \tag{3.4}$$

where K_s is a positive definite diagonal gain matrix. Using (3.3) in the control input (3.4) and substituting it in (3.1) yields,

$$M(q)\dot{s} + C(q, \dot{q})s + Ds + K_s s = Y(q, \dot{q}, v, a)(\tilde{\Theta}_0 + u) \tag{3.5}$$

where $\tilde{\Theta}_0 = \Theta_0 - \Theta$ is the parameter uncertainty which is constant. Suppose the uncertainty is bounded such that a constant bound $\gamma \geq 0$ can be found satisfying,

$$||\tilde{\Theta}_0|| = ||\Theta_0 - \Theta|| \leq \gamma. \tag{3.6}$$

Then, letting $\epsilon > 0$, the control term u is designed as,

$$u = \begin{cases} -\gamma \frac{Y^\top s}{||Y^\top s||} & \text{if} \quad ||Y^\top s|| > \epsilon \\ -\frac{\gamma}{\epsilon} Y^\top s & \text{if} \quad ||Y^\top s|| \leq \epsilon \end{cases} \tag{3.7}$$

Considering the closed loop system (3.5) with bounded parameter uncertainty as (3.6), the additional control u defined as (3.7) and the sliding surface (3.2), it can be shown that the tracking error is uniformly ultimately bounded (u.u.b). This can be done by considering a Lyapunov like function for the system defined as,

$$V = \frac{1}{2} s^\top M_0 s$$

and following the proof given in [74] which is sketched below.

Differentiating V with respect to time and simplifying yields,

$$\dot{V} = - s^\top Q s + s^\top Y_2 (\tilde{\Theta}_0 + u)$$

where the skew symmetric property (*Property A.2*) is utilized and defined $Q := D_0 + K_s$.

Considering the term $s^\top Y_2(\tilde{\Theta}_0 + u)$, observe that if $||Y_2^\top s|| > \epsilon$ then,

$$s^\top Y_2(\tilde{\Theta}_0 + u) =(Y_2^\top s)^\top \left(\tilde{\Theta}_0 - \gamma \frac{Y_2^\top s}{||Y_2^\top s||}\right) \leq ||Y_2^\top s|| \left(||\tilde{\Theta}_0|| - \gamma\right) < 0.$$

This implies that $\dot{V} < 0$ with respect to s. Note that $||\tilde{\Theta}_0|| \leq \gamma$ and $\gamma \geq 0$. Hence, $\tilde{\Theta}_0 \leq \gamma \frac{Y_2^\top s}{||Y_2^\top s||}$. Now, if $||Y_2^\top s|| \leq \epsilon$ then,

$$\begin{aligned} s^\top Y_2(\tilde{\Theta}_0 + u) =&(Y_2^\top s)^\top (\tilde{\Theta}_0 + u) \\ \leq&(Y_2^\top s)^\top \left(\gamma \frac{Y_2^\top s}{||Y_2^\top s||} + u\right) \\ =&\gamma ||Y_2^\top s|| - \frac{\gamma}{\epsilon} ||Y_2^\top s||^2. \end{aligned}$$

The maximum of the R.H.S in the above expression is $\epsilon\gamma/4$ which is achieved when $||Y_2^\top s|| = \epsilon/2$. Therefore,

$$\dot{V} \leq -s^\top Q s + \epsilon\gamma/4$$

and see that $\dot{V} < 0$ if $s^\top Q s > \epsilon\gamma/4$. Using the bounds on the quadratic form, $\lambda_{min}(Q)||s||^2 \leq s^\top Q s \leq \lambda_{max}(Q)||s||^2$ where $\lambda_{min}(Q)$ and $\lambda_{max}(Q)$ are, respectively, the minimum and maximum Eigenvlaues of the matrix Q, we have that $\dot{V} < 0$ if $\lambda_{min}(Q)||s||^2 > \epsilon\gamma/4$ or, equivalently

$$||s|| > \left(\frac{\epsilon\gamma}{4\lambda_{min}(Q)}\right) =: \delta.$$

The u.u.b follows from this result using δ to define the radius of the ultimate boundedness set.

3.4 Numerical simulations

A three-link planar soft robot with uniform mass distribution is used in this simulation study. Each link was considered to be identical with a length of $L_i = 1$m, width of $D_i = 0.05$m, normalized mass of $\rho_i = 5$kg, stiffness of $k_i = 0.5$ and damping of $\beta_i = 0.05$. Matlab/Simulink was used for the simulations, and for all the simulations, the simulation environment was as in Section 2.5.1.

For the simulation study, two scenarios were considered. In *scenario 1*, the desired task space trajectory was defined as,

$$x_d(t) = -0.9 + 0.8\sin(0.5\pi t), \qquad y_d(t) = 1.6 + 0.6\cos(0.5\pi t)$$

For the *scenario 2*, the frequency and the offset of the x-direction is changed at the middle of the simulation. For the initial 11s, the desired trajectory is the same as in *scenario 1*. At 11s, the desired trajectory is changed to

$$x_d(t) = 0.5 + 0.8\sin(0.8\pi t), \qquad y_d(t) = 1.6 + 0.6\cos(0.5\pi t)$$

until 30s at which point the desired trajectory is changed back to the same as in *scenario 1*.

The performance of the proposed robust controller for task space trajectory tracking was tested with a passivity-based adaptive control approach (as in Chapter 2) for both the scenarios. Moreover, the robust controller and the adaptive controller designed on the basis of the *augmented formulation* [22] was tested as well.

3.4.1 Simulation results

The uncertainty in the segment masses, damping, and stiffness parameters were considered in the simulations. Therefore, the parameter vector was chosen as $\Theta = [m_1, m_2, m_3, k, d]^\top$ and the nominal parameter vector for the robust controller was set to $\Theta_0 = [4, 4, 4, 0.5, 1]^\top$. The bound on uncertainty was found to be $\gamma \geq 1.858$, and it was set to $\gamma = 2.5$. The value $\epsilon = 0.1$ and the control gains $K_s = 1$, $\Lambda = 12$ were chosen. For the adaptive controller, the initial parameter estimation was set to $\hat{\Theta}(0) = [4, 4, 4, 0.5, 1]^\top$. The adaptation gain was $\Gamma = 2$ and gain $K_s = 1$.

3.4.1.1 Scenario 1

For this scenario, the performance comparison for task space trajectory tracking for the proposed robust and adaptive controller is illustrated in Figure 3.3. Observe that the adaptive controller initially has higher errors, but the robust and adaptive controllers eventually have similar performance. Considering the robust controller, the dynamics model used for the control derivation has shown negligible effect for the studied case, as seen in Figure 3.4. However, the controller developed based on the proposed distributed mass model achieved less tracking error for adaptive control than the *augmented formulation*-based adaptive controller. From Figure 3.5, observe that the *augmented formulation*-based controller exhibits unsteady motions at certain instances.

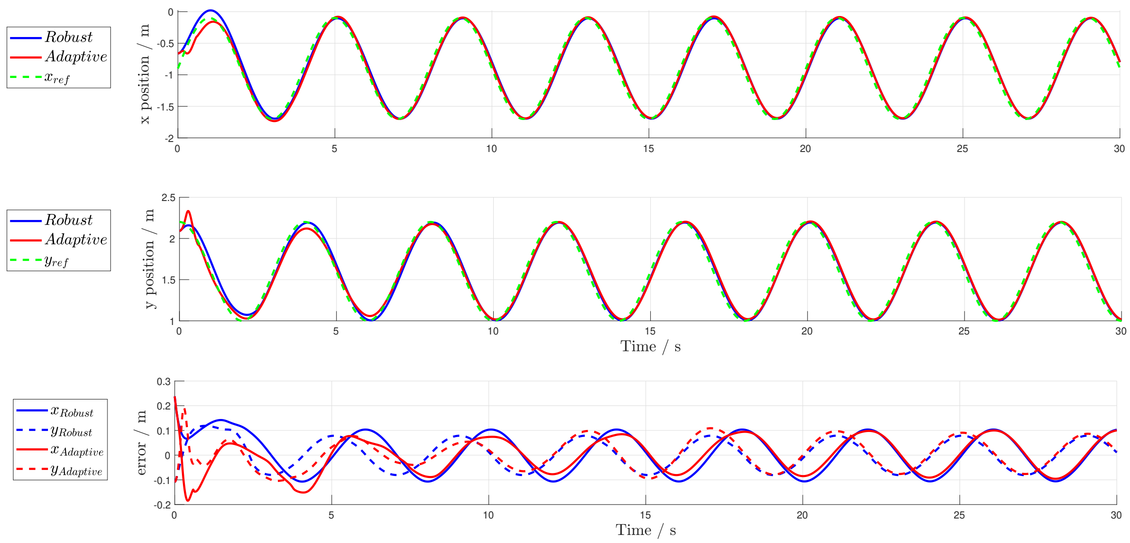


Figure 3.3: Simulation results for *scenario 1* illustrating the performance comparison between robust control and adaptive control.

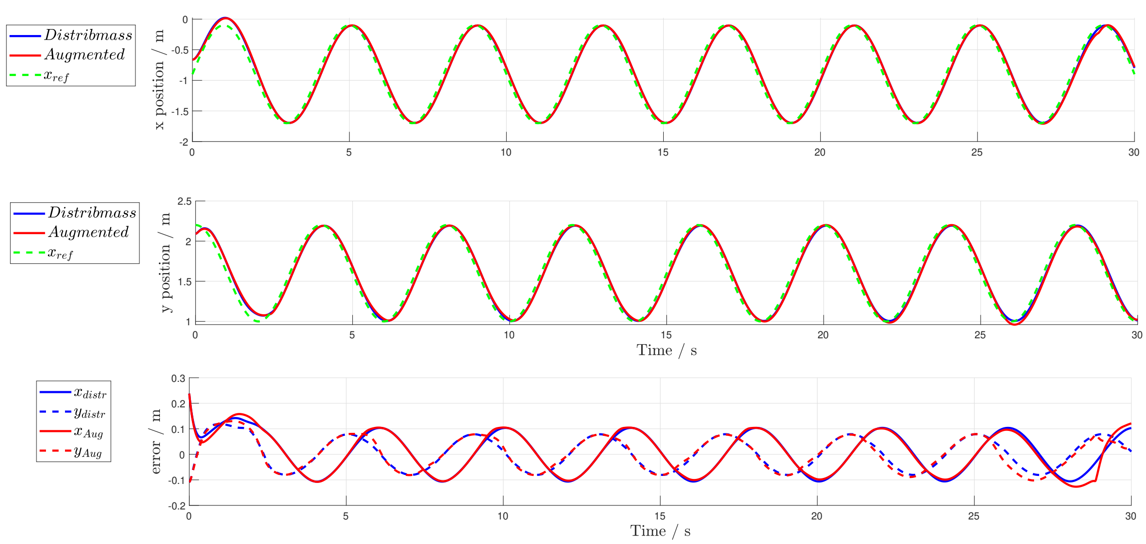


Figure 3.4: Simulation results for *scenario 1* illustrating the performance comparison of the robust controllers designed on the basis of the distributed mass model and the *augmented formulation*

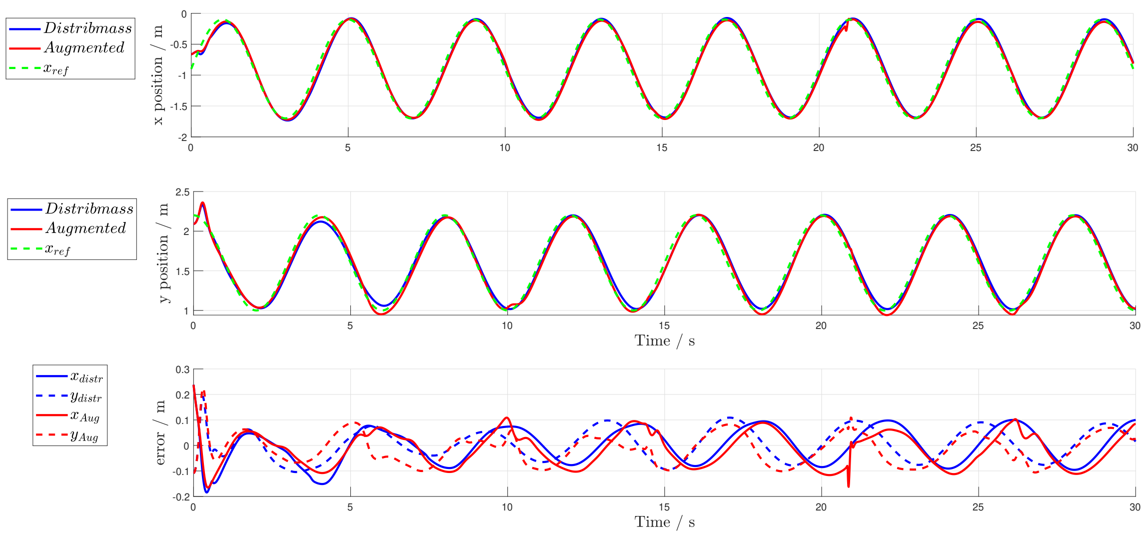


Figure 3.5: Simulation results for *scenario 1* illustrating the performance comparison of the adaptive controllers designed on the basis of the distributed mass model and the *augmented formulation*

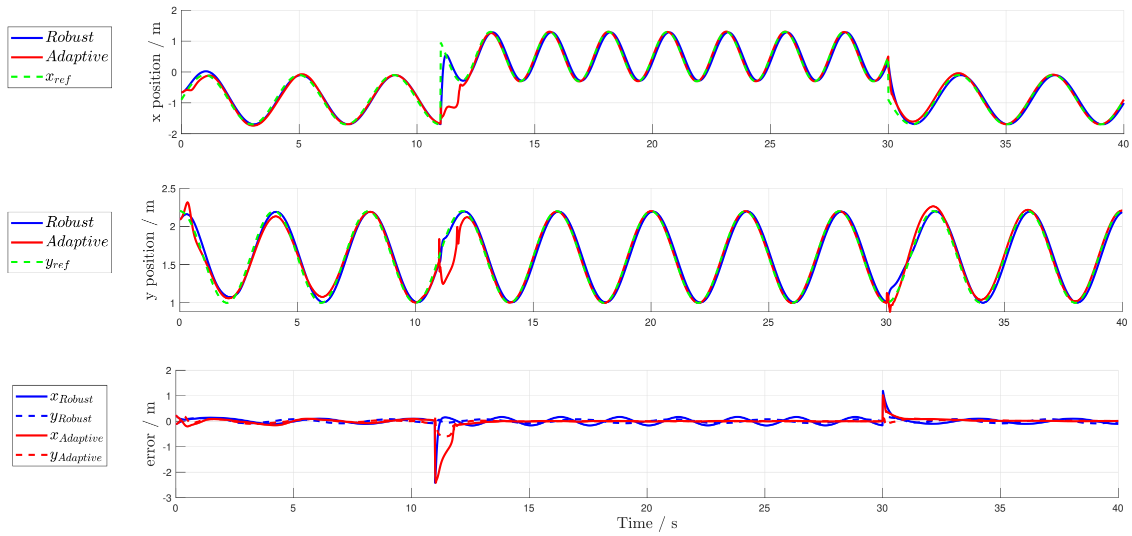


Figure 3.6: Simulation results for *scenario 2* illustrating the performance comparison between robust control and adaptive control.

3.4.1.2 Scenario 2

For this scenario, observe from Figures 3.6-3.7 that the robust controller designed on the basis of the distributed mass model performs well with small transients errors when the reference trajectory is changed. On the other hand, the robust controller designed on the basis of the *augmented formulation* becomes unstable due to the sudden change in the reference trajectory as seen in Figure 3.7. The performance of the adaptive controller is illustrated in Figure 3.8 where satisfactory tracking performance is observed despite higher transient times than the robust controller.

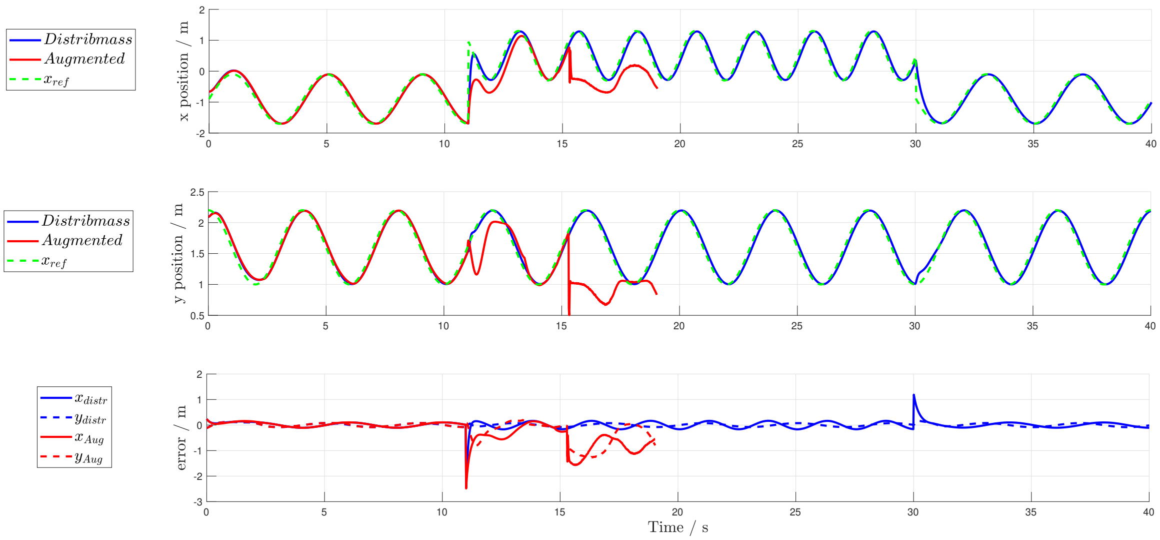


Figure 3.7: Simulation results for *scenario 2* illustrating the performance comparison of the robust controllers designed on the basis of the distributed mass model and the *augmented formulation*

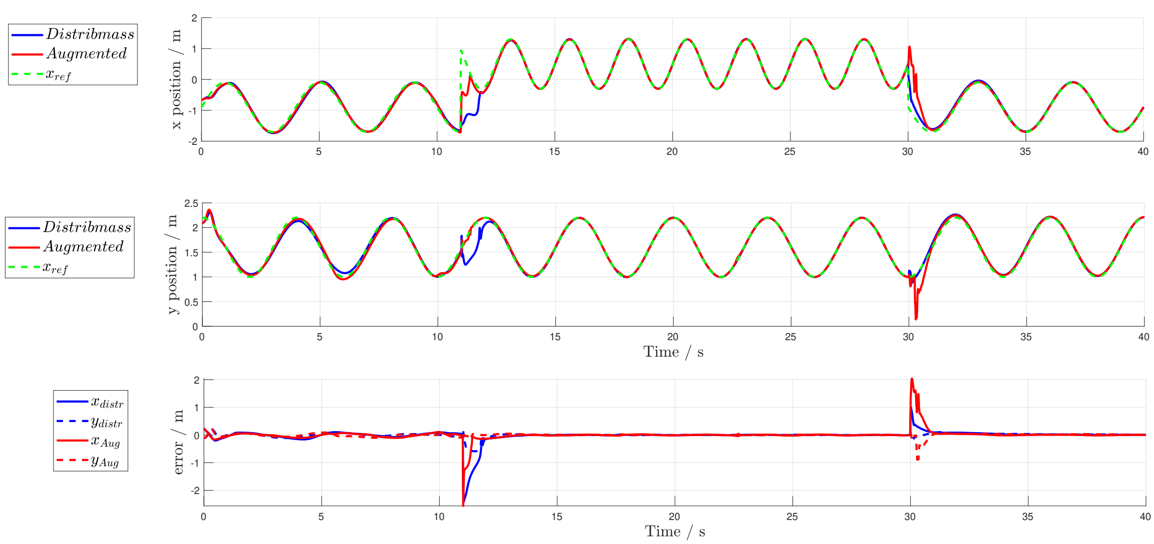


Figure 3.8: Simulation results for *scenario 2* illustrating the performance comparison of the adaptive controllers designed on the basis of the distributed mass model and the *augmented formulation*

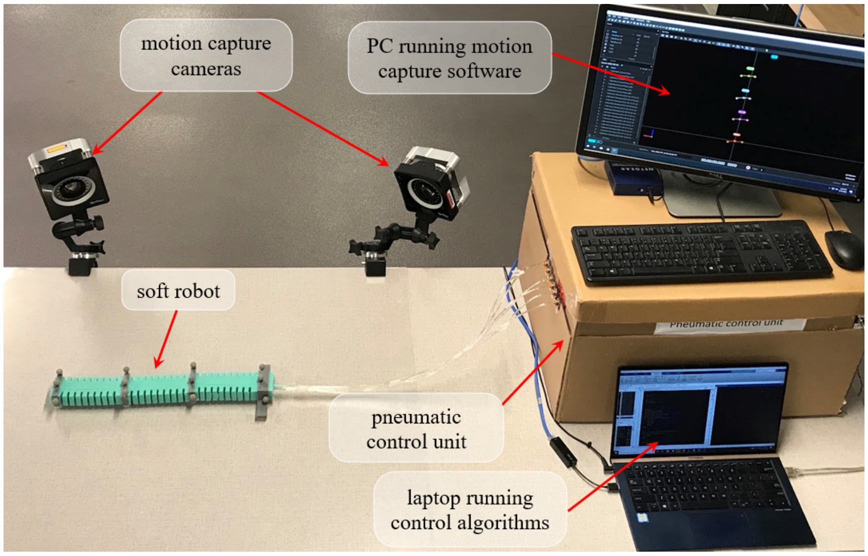


Figure 3.9: The experimental setup

3.5 Experimental evaluation

This section demonstrates the proposed passivity-based robust controller for task space trajectory tracking. First, the experimental setup is briefly described and then the results are presented discussing the insights.

3.5.1 Experimental setup

The experimental setup, as shown in Figure 3.9 is similar to the one that was used in Chapter 2. The soft robot is a pneumatically actuated three-link pleated type soft robot fabricated following methods outlined in [75]. To reduce friction, ball transfers were used underneath near the segment joints. Each segment of the soft robot was measured to be $L_i = 0.125$ m and the segment masses $m_i = 0.110$ kg were measured prior to joining the segments together. The material properties of each segment were assumed to be identical. Therefore, for all the segments identical torsional stiffness of $k_i = k$ and damping of $d_i = d$ were assumed and were identified to be $k_i = 1$Nm/rad and $d = 0.2$Nms/rad.

Six motion capture cameras were used to measure the markers attached at the end of each segment and used them to estimate the degree of curvatures of the links. The soft robot was actuated using a pneumatic control unit based on the open source hardware platform [76] with command inputs (PWM signals) serially transmitted to the control board (Arduino Mega). An external compressor supplied the compressed air to the unit at a constant pressure of 20psi. The torque-to-PWM signals mapping was identified by a curve fitting process for each compartment of the three segments as described in Chapter 2.

3.5.2 Experimental results

The proposed robust controller for tracking a predefined task space reference trajectory was tested and compared the performance with passivity-based adaptive control. Similarly to the simulation study, the robust and adaptive controllers designed on the basis of the *augmented formulation* were tested.

For the experimental investigation, the parameter vector was chosen as $\Theta = [m_1, m_2, m_3, k, d]^\top$ and the nominal parameter vector for the robust controller was set to $\Theta_0 = [0.075, 0.075, 0.075, 0.75, 1.2]^\top$. The bound on uncertainty was found to be $\gamma \geq 0.588$, and it was set to $\gamma = 1.2$ in the controller. The value $\epsilon = 0.05$ and the control gains $K_s = 0.3$, $\Lambda = 3.5$ were chosen. For the adaptive controller, the initial parameter estimation was set to $\hat{\Theta}(0) = [0.075, 0.075, 0.075, 0.75, 1.2]^\top$. The adaptation gain was $\Gamma = 0.1$ and gains $K_s = 0.3$,$\Lambda = 3.5$.

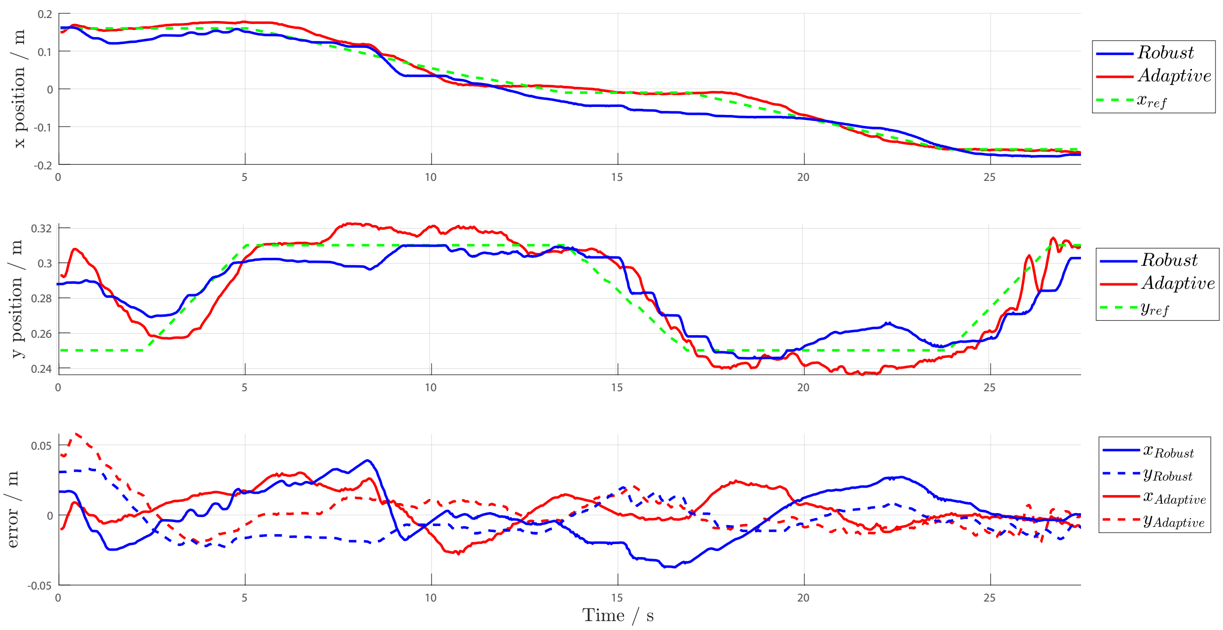


Figure 3.10: Experimental results for task space trajectory tracking illustrating the performance of the robust vs adaptive control.

The performance comparison between the robust and adaptive controller is shown in Figure 3.10, and observe that the performance of both controllers is comparable, with the robust controller having a positional RMSE of 0.028m and the adaptive controller 0.03m. The performance of the proposed robust controller designed upon the proposed distributed mass dynamics model is shown in Figure 3.11 along with that of the robust controller designed using the *augmented formulation*. Observe that the controller designed utilizing the proposed distributed mass dynamics model achieves better tracking performance. The performance of the adaptive control frameworks are illustrated in Figure 3.12, and observe that utilizing the proposed distributed dynamics model for control has yielded marginally better tracking.

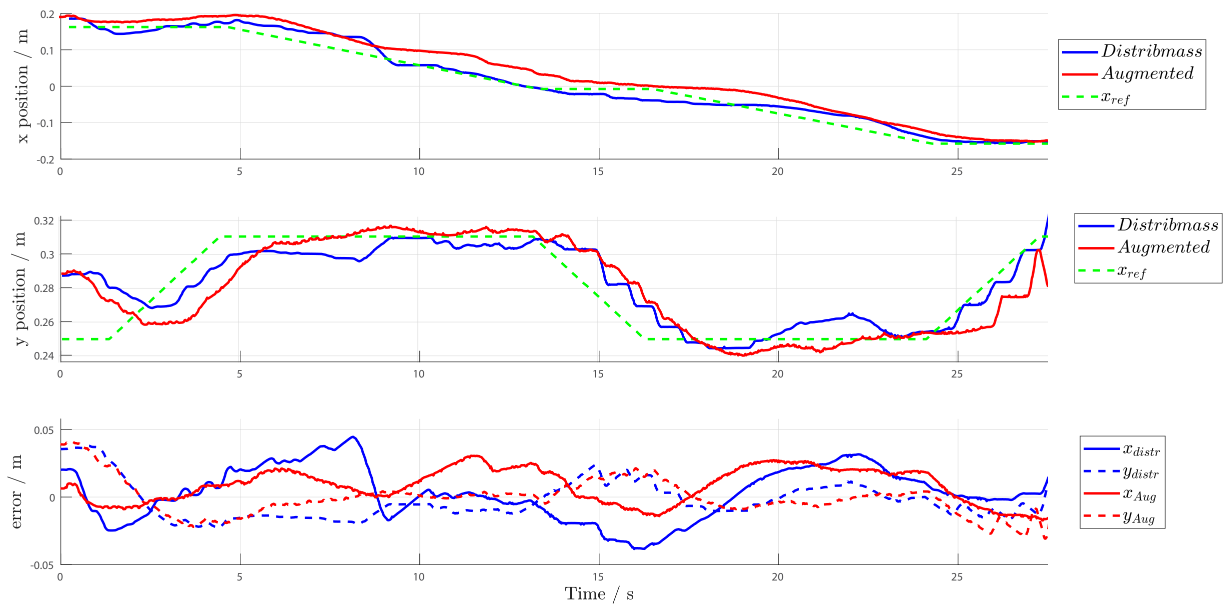


Figure 3.11: Experimental results for task space trajectory tracking illustrating the performance comparison of the robust controllers designed on the basis of the distributed mass model and the *augmented formulation*

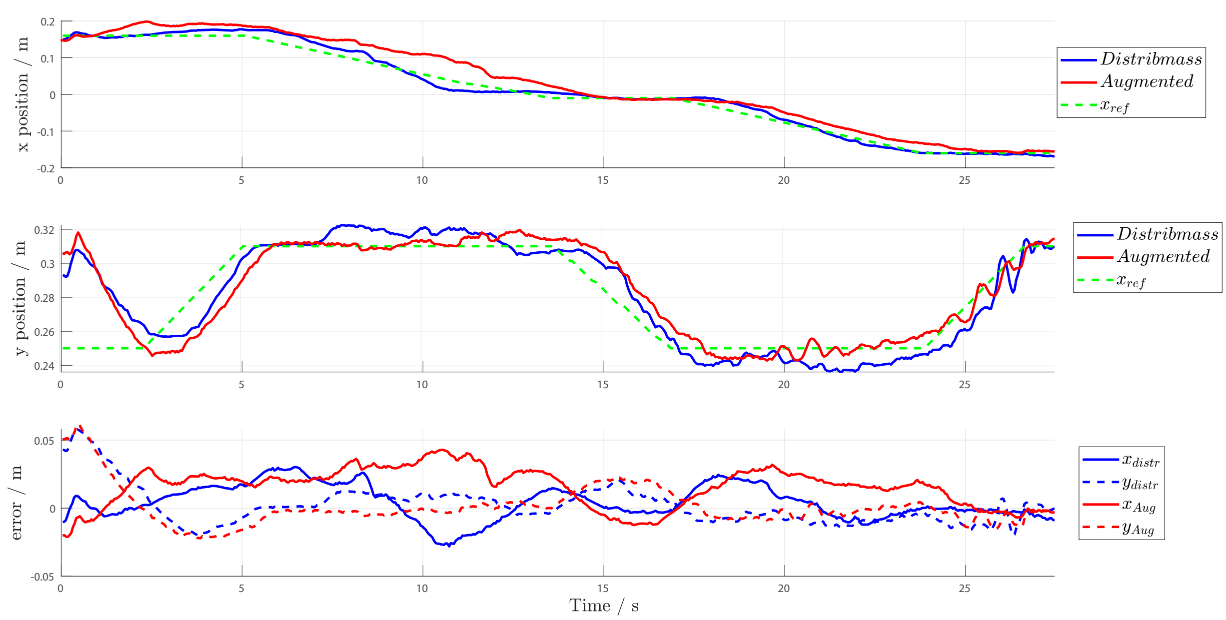


Figure 3.12: Experimental results for task space trajectory tracking illustrating the performance comparison of the adaptive controllers designed on the basis of the distributed mass model and the *augmented formulation*

3.6 Summary

In this chapter, an enhanced dynamics model for planar PCC soft robots based on the Euler-Lagrange formulation was introduced by explicitly considering the mass distribution of the soft robot. This model captures the inertia effects of the soft robots, which is neglected in simplified models considering a lumped mass system. Further, a passivity-based robust controller was designed leveraging this dynamics model and its efficacy was illustrated via simulations and physical experiments. However, note that due to the limitations set by the physical soft robot used for the experiments, fast-moving trajectories could not be tested for the proposed controller.

Chapter 4: Robust Task Space Bilateral Teleoperation of Soft Robots

This chapter introduces a robust task space bilateral teleoperation (RTSBT) framework for soft robots with dynamic uncertainties in the presence of time delays and external disturbances. The leader robot is assumed to be a non-redundant rigid manipulator and the follower robot is assumed to be a redundant spatial soft manipulator under the PCC hypothesis. Passivity-based robust control is employed to formulate the bilateral teleoperation framework proving the ultimate boundedness of the bilateral teleoperation system trajectories in the presence of dynamic uncertainties and constant time delays and further enhancing the control framework by adding disturbance rejection. The null space velocity of the soft robot is also exploited to achieve collision avoidance sub-task in the follower environment while achieving task space synchronization. The proposed control algorithms were implemented in simulations on a spatial soft robot as well as in physical experiments on a planar soft robot illustrating the efficacy of the proposed robust teleoperation framework.

4.1 Overview

Bilateral teleoperation of soft robots can be utilized to realize tasks in remote/hazardous environments by deploying a soft robotic manipulator, as discussed in Chapter 2. Moreover, this capability of the soft robots alongside their agile and dexterous motions, can be beneficial especially for robotic tele-surgery, rehabilitation, inspections in cluttered environments and operations in extreme conditions such as space exploration among others which a human operator can guide the soft robot in the remote environment by manipulating a local rigid robotic manipulator.

This chapter extends the distributed mass dynamics model discussed in Chapter 3.2 to a 3D spatial soft robot. Leveraging that model, a passivity-based robust task space bilateral teleoperation framework is designed as illustrated in Figure 4.1. Robust control is used to mitigate any uncertainty in the dynamics as well as any unmodelled disturbances. Different from the robust joint space teleoperation in [83], this chapter develops robust teleoperators for task space with the goal of position synchronization. This is especially important in view of potential applications of soft robots. Moreover, this chapter considers dissimilar systems, the follower robot being a soft robot and the leader being a rigid robot, and show that task space position synchronization can be achieved. The ultimate boundedness of the trajectories of the teleoperation system is proven even under dynamic uncertainties, constant time delays and unmodelled external disturbances in the presence of passive or non-passive external forces. Further, the null space velocity tracking of the redundant soft follower robot is utilized to achieve collision avoidance subtask which is useful for semi-autonomous operations. Different from Chapter 2 and previous work [73], the passivity-based robust control for task space bilateral teleoperation framework presented here is able to handle unmodelled uncertainties. Further, an additional disturbance rejection control added in this chapter will enable robust teleoperation performance even in the presence of bounded disturbances. Extensive simulations were conducted for 3D operation of the teleoperation system showing the efficacy of the proposed RTSBT framework. While the analysis is for a 3D teleoperation system, physical

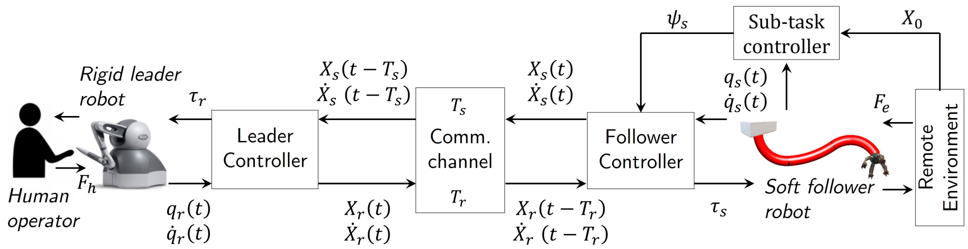


Figure 4.1: Control architecture of the bilateral teleoperation system with a rigid leader robot and a soft follower robot with constant communication delays.

experimentation of the proposed teleoperation framework was conducted on a planar 2D system, illustrating the utility of the developed RTSBT framework for the planar case as well.

The rest of the chapter is organized as follows. Section 4.2 introduces the enhanced soft robot model and the rigid robot model. The RTSBT framework is developed in Section 4.3. The sub-task control is discussed in Section 4.4. The simulation results are presented in Section 4.5 and the experimental results in Section 4.6.

4.2 Spatial distributed mass soft robot model

Consider a spatial in-extensible soft robot as the follower robot assuming PCC [16, 84] and consider the case wherein the base of the soft robot is fixed. In this section, the kinematics modelling follows [84], but this chapter considers the soft segments as continuous arc segments with a circular cross section and considers each and every point on that volume, unlike only considering specific point of each segments as done in [84]. In other words, this chapter is not assuming lumped mass, rather considers the soft segments to have a distributed mass. When compared with [85], the work here adheres to the physical limitation of non-physical torsion of the arm as detailed by [17, 84]. Moreover, in this section a PCC formulation is considered and it is assumed that the soft robot is constructed in such a way that it adheres to the have a PCC

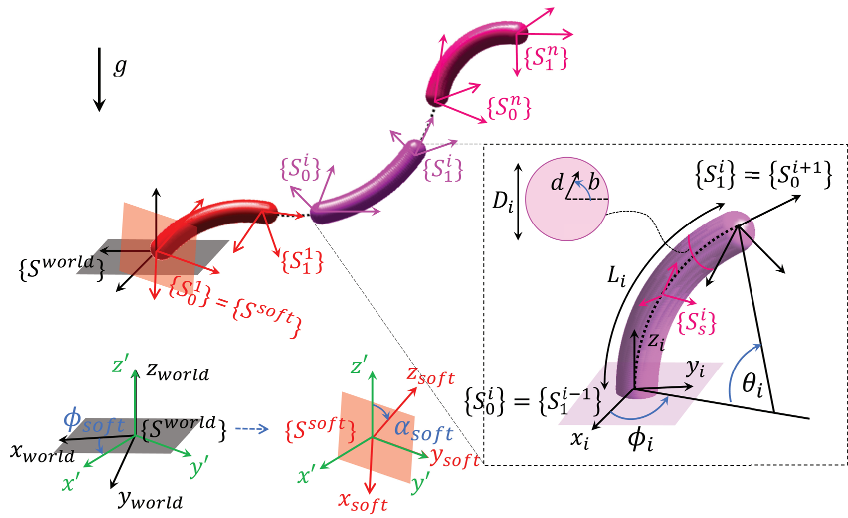


Figure 4.2: A spatial PCC soft robot with n links is illustrated with a closeup is of the i^{th} element. The bottom left figure illustrates the trasformation between $\{S^{world}\}$ ans $\{S^{soft}\}$.

such as the one in [86]. The work in this chapter will rely on the passivity-based robust control framework to handle any unmodelled effects.

4.2.1 Kinematics

The spatial soft robot is assumed to be composed of n in-extensible segments. Each segment is assumed to have a time-varying CC along the length of the segment. To constitute the PCC soft robot, these n CC segments are attached so that the resulting curve is differentiable everywhere. Under the PCC hypothesis, for a spatial segment, two variables, namely the degree of curvature (θ_i) and the bending plane angle (ϕ_i), as shown in Figure 4.2 is sufficient to describe the segment's configuration in space. The configuration variables of the soft segments are defines as $q_s(t) = [\theta_1(t), \phi_1(t), \theta_2(t), \phi_2(t), ..., \theta_n(t), \phi_n(t)]^\top$.

Let the global frame of reference $\{S^{global}\}$ fixed in space such that the gravity is along negative z-axis. Consider the base reference frame of the soft robot $\{S^{soft}\}$ which is rotated by

ϕ_{soft} around z axis of the $\{S^{global}\}$ and α_{soft} around the new y' axis. The soft robot is oriented so that when all the links are straight, the robot aligns with the z-axis of the $\{S^{soft}\}$. As shown in Figure 4.2, let the length of the i^{th} segment along its central axis be L_i and its cross sectional diameter D_i. The length of each segment is parameterized by $s \in [0, 1]$, such that sL_i is the arc length along the segment's central axis to the point s from the i^{th} segment's base. At each point s for any segment i along the soft robot's body, reference frames $\{S_s^i\}$ are attached, with the superscript i denoting the considered segment. Moreover, the cross section of the segments is parameterized by $d \in [0, 1]$ and $b \in [0, 2\pi]$ such that $dD_i/2$ is the lateral distance to point (d, b) on the i^{th} segment from the central axis located at an angle b from the local x axis of frame $\{S_s^i\}$.

The bending angle $\alpha_s^i(t)$ of the reference frame $\{S_s^i\}$ at a point s along the central axis on the i^{th} segment with reference to the frame $\{S_0^i\}$ can be written as the sum of integral of the curvature,

$$\alpha_s^i(t) = \int_0^s q_i(t)\mathrm{d}l = q_i(t)s.$$

Moreover, the bending plane of the i^{th} segment is ϕ_i. However, note that most soft robots are designed to achieve torsion-free backbones [17], hence requiring a reverse rotation of ϕ_i around the local y axis of each frame $\{S_s^i\}$. Thus the rotation from the base of the i^{th} segment, i.e, from frame $\{S_0^i\}$ to the reference frame $\{S_s^i\}$ is:

$$R_s^i = R_z(\phi_i)R_y(\alpha_s^i)R_z(-\phi_i)$$

and the translation is:

$$t_s^i = \begin{bmatrix} \cos(\phi_i) L_i \int_0^s \sin(\alpha_s^i(t)) \mathrm{d}s \\ \sin(\phi_i) L_i \int_0^s \sin(\alpha_s^i(t)) \mathrm{d}s \\ L_i \int_0^s \cos \alpha_s^i(t) \mathrm{d}s \end{bmatrix},$$

which defines the homogeneous transformation from $\{S_0^i\}$ to $\{S_s^i\}$,

$$H_s^i = \begin{bmatrix} R_s^i & t_s^i \\ 0_{1\times 3} & 1 \end{bmatrix}.$$

Thus the Cartesian coordinates

$$\left(x_{(s,d,b)}^i(t), y_{(s,d,b)}^i(t), z_{(s,d,b)}^i(t)\right)$$

at a general point $P_{s,d,b}^i$ on the i^{th} segment parameterized by (s, d, b) on the soft robot is given by,

$$P_{s,d,b}^i = H_s^i \begin{bmatrix} dD\cos(b) \\ dD\sin(b) \\ 0 \\ 1 \end{bmatrix}.$$

Define the direct forward kinematics $h_s(q_s) : \mathbb{R}^{2n} \to \mathbb{R}^3$ which maps the configuration space to the task space,

$$h_s(q_s) = \left[x^n_{(1,0,0)}\ ,\ y^n_{(1,0,0)}\ ,\ z^n_{(1,0,0)}\right]^\top .$$

Thus, the end effector position and the velocity are defined as, $X_s(t) = h_s(q_s)$ and $\dot{X}_s(t) = J_s(q_s)\dot{q}_s$, where, $J_s(q_s) = \frac{\partial h_s(q_s)}{\partial q_s} \in \mathbb{R}^{3\times 2n}$ is the Jacobian matrix.

4.2.2 Dynamics

This section derives the dynamics of the soft robot based on the Euler-Lagrange (E-L) formalism. It should be noted that in this dynamics derivation, no assumptions are mad on the mass distribution, such as uniform linear mass density [82] or concentrated masses at discrete points [22, 43]. Instead, a normalized mass distribution $\rho_i(s, d, b)$ is considered for each segment i. However, this formulation assumes that the change in the density due to shape changes is negligible. In the following, unless different from previously defined dependencies, the arguments will be suppressed in the expressions due to brevity and clear representation.

Following the E-L method, the inertia matrix $M_s(q_s) \in \mathbb{R}^{2n\times 2n}$ is evaluated as,

$$M_s(q_s) = \sum_{i=1}^{n} \int_0^1 \int_0^1 \int_0^{2\pi} \rho_i \nabla_{q_s}(P^i_{s,d,b})^\top \nabla_{q_s}(P^i_{s,d,b}) \mathrm{d}b\, \mathrm{d}d\, \mathrm{d}s$$

where ∇_{q_s} is the gradient operator $\nabla_{q_s}(\cdot) = \frac{\partial(\cdot)}{\partial q_s}$. Then the centrifugal and Coriolis terms matrix $C_s(q_s, \dot{q}_s) \in \mathbb{R}^{2n\times 2n}$ is evaluated using the standard Christoffel symbols [74].

The gravitational potential energy $\mathcal{P}_g \in \mathbb{R}$ of the soft robot can be calculated by,

$$\mathcal{P}_g = \sum_{i=1}^{n} \left(\int_0^1 \int_0^1 \int_0^{2\pi} g\rho_i z^i \, \mathrm{d}b \, \mathrm{d}d \, \mathrm{d}s \right)$$

Therefore the gravity terms vector $\mathbf{G}_s(q_s) \in \mathbb{R}^{2n}$ is evaluated as,

$$G_s(q_s) = \nabla_{q_s} (\mathcal{P}_g) \, .$$

The elasticity of the soft robot is modeled through a continuous and homogeneous distribution of infinitesimal springs and dampers along the length of the soft segments [84]. Assuming a linear relationship for elastic parameters, the stiffness and damping terms for the i^{th} segment are defined as $K_i q_{s_i}$ and $B_i \dot{q_{s_i}}$, respectively. Here $K_i \in \mathbb{R}^{2\times2}$ and damping $B_i \in \mathbb{R}^{2\times2}$ are defined as,

$$K_i = \begin{bmatrix} 0 & 0 \\ 0 & k_i \end{bmatrix}, \quad B_i = \beta_i \begin{bmatrix} \theta_i^2 & 0 \\ 0 & 1 \end{bmatrix}$$

where k_i and β_i are the stiffness the damping of the i^{th} segment. The total stiffness matrix and the damping matrix are block diagonal matrices consisted of K_i and B_i.

In the absence of unmodelled external disturbances, the complete dynamics are,

$$M_s(q_s)\ddot{q}_s + C_s(q_s, \dot{q}_s)\dot{q}_s + B_s\dot{q}_s + K_s q_s + G_s(q_s) = \tau_s - J_s^\top (q_s) \, F_e \tag{4.1}$$

Here $\tau_s \in \mathbb{R}^{2n}$ is the control input and physically it is the effective torque applied on the soft links causing them to bend with a CC. F_e is the external force applied at the soft robot's end effector.

4.3 Robust bilateral teleoperation framework

This section develops the RTSBT framework for soft robots considering a rigid leader robot and a soft follower robot. First the leader robot dynamics are introduced and subsequently the bilateral teleoperation framework is designed.

4.3.1 The rigid manipulator : leader robot

A classical 3-DoF elbow manipulator (rigid robot)[74] is considered as the leader robot for the bilateral teleoperation framework. In the absence of external disturbances, the dynamics of the rigid robot can be written in the Euler-Lagrangian formulation as [74],

$$M_r\left(q_r\right)\ddot{q}_r + C_r\left(q_r, \dot{q}_r\right)\dot{q}_r + G_r\left(q_r\right) = \tau_r + J_r^{\top}\left(q_r\right)F_h \tag{4.2}$$

with $q_r \in \mathbb{R}^3$. Here $M_r(q_r) \in \mathbb{R}^{3\times3}$ is the inertial matrix, $C_r(q_r, \dot{q}_r)\dot{q}_r \in \mathbb{R}^3$ is the matrix of the centrifugal and Coriolis terms and $G_r(q_r) \in \mathbb{R}^3$ is the gravitational torque. Here,$q_r \in \mathbb{R}^p$ is the vector of the relative joint angles, $\tau_r \in \mathbb{R}^3$ is the control torque, and $F_h \in \mathbb{R}^3$ is the operator force at the end effector which is mapped using the Jacobian $J_r(q_r) = \frac{\partial h_r(q_r)}{\partial q_r} \in \mathbb{R}^{3\times3}$. The end effector position and the velocity are defined as, forward kinematics $X_r(t) = h_s(q_r)$ and $\dot{X}_r(t) = J_r(q_s)\dot{q}_r$.

Remark 4.1 *Note that the soft robot dynamics in (4.1) and the rigid robot dynamics in (4.2) are in the Lagrangian form, and hence they both encapsulates the properties of Lagrangian* *systems such as the skew symmetric property and the linearly parameterizable property [74] (see Appendix A.2).*

The position synchronization is considered as the goal of the RTSBT framework. To that end, the tracking errors for the leader system and soft follower system are defined as $e_r(t) = X_s(t - T_s) - X_r(t)$ and $e_s(t) = X_r(t - T_r) - X_s(t)$ respectively. Here, communication time delays T_s and T_r are assumed to be constant. Moreover, it is assumed that the time delays are bounded such that $T_s \le \bar{T}_s$ and $T_r \le \bar{T}_r$ and the round trip delay bound is defined as $\bar{T} = \bar{T}_s + \bar{T}_r$.

Similar to Chapter 2, the signals $s_r(t)$ and $s_s(t)$ for the rigid leader and the soft follower are defined as,

$$
\begin{aligned}
s_r &= -J_r^{-1}\lambda e_r + J_r^{-1}\dot{e}_r, \\
s_s &= -J_s^{+}\lambda e_s + J_s^{+}\dot{e}_s - (I_n - J_s^{+}J_s)\psi_s
\end{aligned}
\tag{4.3}
$$

where $\lambda \in \mathbb{R}^{+}$ is positive gain. $\psi_s \in \mathbb{R}^3$ is the negative gradient of an appropriately defined convex function which is utilized for the sub-task control. $J_s^{+} \triangleq J_s^T(J_sJ_s^T)^{-1} \in \mathbb{R}^{2n\times 3}$ is the pseudo inverse of J_s and satisfies the property $J_sJ_s^{+} = \mathbb{I}_3$. The signals $v_i(t), a_i(t)$ are then defined as, $v_i = \dot{q}_i - s_i$ and $a_i = \ddot{q}_i - \dot{s}_i$, respectively, for $i \in \{r, s\}$.

Using the properties of Lagrangian systems the regressor $(Y_i\,(q_i, \dot{q}_i, v_i, a_i))$ and parameter (Θ_i) vector pair for the estimated systems [74] is defined as,

$$
\begin{aligned}
\hat{M}_r a_r + \hat{C}_r v_r + \hat{G}_r =& Y_r\hat{\Theta}_r \\
\hat{M}_s a_s + (\hat{C}_s + \hat{B})v_s + \hat{K}q_s + \hat{G}_s =& Y_s\hat{\Theta}_s
\end{aligned}
\tag{4.4}
$$

where $\hat{(\cdot)}$ indicates the estimates for the corresponding terms. Here, time invariant uncertainties in the mass of both robots, and stiffness and damping of the soft robot is assumed. The estimated parameter vectors are niw defined as,

$$\hat{\Theta}_s = \Theta_s^0 + u_s$$
$$\hat{\Theta}_r = \Theta_r^0 + u_r \tag{4.5}$$

where Θ_s^0 and Θ_r^0 are fixed nominal parameter vectors and u_s and u_r are additional control terms which will be designed subsequently for achieving robustness for uncertainty in the model parameters.

4.3.2 Bilateral teleoperation in the absence of disturbances

This section considers the case when there are no unmodelled disturbances in the systems. In this case, let the control inputs for the rigid leader and soft follower robots given as,

$$\begin{aligned}
\tau_r =& \hat{M}_r a_r + \hat{C}_r v_r + \hat{G}_r - K_r s_r - J_r^T \bar{\tau}_r \\
=& Y_r \hat{\Theta}_r - K_r s_r - J_r^T \bar{\tau}_r \\
\tau_s =& \hat{M}_s a_s + (\hat{C}_s + \hat{B}) v_s + \hat{K}_0 q_s + \hat{G}_s - K_s s_s - J_s^T \bar{\tau}_s \\
=& Y_s \hat{\Theta}_s - K_s s_s - J_s^T \bar{\tau}_s.
\end{aligned} \tag{4.6}$$

Here K_i for $i \in \{r, s\}$ are positive definite diagonal gain matrices. $\bar{\tau}_i$ are the coordinating control torques that are defined as,

$$\bar{\tau}_i = k_r(-\lambda e_i + \dot{X}_i) - k_J \dot{e}_i \tag{4.7}$$

where $k_r \in \mathbb{R}^+$ and $k_J \in \mathbb{R}^+$ are positive control gains.

The closed loop dynamics of the system is found by substituting the proposed controls (4.6) in (4.1)-(4.2) and using (4.4),

$$\begin{aligned} M_s \dot{s}_s + C_s s_s + B s_s + K_s s_s &= Y_s \left(\tilde{\Theta}_s + u_s \right) - J_s^T \bar{\tau}_s - J_s^T F_e \\ M_r \dot{s}_r + C_r s_r + K_r s_r &= Y_r \left(\tilde{\Theta}_r + u_r \right) - J_r^T \bar{\tau}_r + J_r^T F_h \end{aligned} \tag{4.8}$$

where $\tilde{\Theta}_i = \Theta_i^0 - \Theta_i$ is the parameter uncertainty. Suppose the uncertainty is bounded such that one can find a constant bound $\gamma_i \geq 0$,

$$||\tilde{\Theta}_i|| = ||\Theta_i^0 - \Theta_i|| \leq \gamma_i. \tag{4.9}$$

Following the well-known robust control approach in [80] and letting $\epsilon_i > 0$, the control terms u_i's are designed as,

$$u_i = \begin{cases} -\gamma_i \frac{Y_i^\top s_i}{||Y_i^\top s_i||} & \text{if} \quad ||Y_i^\top s_i|| > \epsilon_i \\ -\frac{\gamma_i}{\epsilon_i} Y_i^\top s_i & \text{if} \quad ||Y_i^\top s_i|| \leq \epsilon_i \end{cases} \tag{4.10}$$

Consider the states of the RTSBT system to be $z(t) = [s_s, s_r, e_s, e_r, \dot{e}_s, \dot{e}_r, \dot{X}_s, \dot{X}_r]^\top$. In the following result it is proven that in the absence of any external unmodelled disturbances, the proposed robust task space bilateral teleoperation framework is uniformly ultimately bounded (u.u.b) with respect to the states $z(t)$. First consider a Lemma that is useful in the proof.

Lemma 4.1 *[87] For a positive-definite matrix Υ, and signals $a, b \in \mathbb{R}^n$ the following inequality holds for $\forall T > 0$:*

$$-2a^\top(t)\int_{t-T}^{t} b(\sigma)d\sigma - \int_{t-T)}^{t} b^\top(\sigma)\Upsilon b(\sigma)d\sigma \leq Ta^\top(t)\Upsilon^{-1}a(t)$$

Theorem 4.1 *Consider the RTSBT framework (4.8) with additional control law (4.10). If there exist positive control parameters ϵ_r, ϵ_s, K_r, K_s, k_p, k_J, nominal parameter vectors Θ_r^0, Θ_s^0 constant bounds $\gamma_r, \gamma_s \geq 0$ for the uncertainties and $\lambda < 1/\bar{T}$, then in the absence of any external forces ($F_h = F_e = 0$), the system trajectories are u.u.b.*

Proof of Theorem 4.1 *Consider a positive definite storage function,*

$$\begin{aligned} V_1 =& \lambda\bar{e}^\top k_p\bar{e} + \frac{1}{2}\sum_{i=s,m}\Big(s_i^\top M_i s_i + \lambda e_i^\top k_J e_i + 2\lambda\int_{t-\bar{T}_i}^{t}\left(\sigma - t + \bar{T}_i\right)\dot{X}_i^\top(\sigma)k_p\dot{X}_i(\sigma)d\sigma \\ &+ \int_{t-T_i}^{t}\dot{X}_i^\top(\sigma)k_J\dot{X}_i(\sigma)d\sigma\Big) \end{aligned} \tag{4.11}$$

where $\bar{e} = X_s(t) - X_r(t)$ is the undelayed position error. Taking the derivative of V,

$$\begin{aligned} \dot{V}_1 =& \sum_{i=s,m}\Big(s_i^\top M_i\dot{s}_i + \frac{1}{2}s_i^\top\dot{M}_i s_i + \lambda e_i^\top k_J\dot{e}_i + \frac{1}{2}\dot{X}_i^\top k_J\dot{X}_i - \frac{1}{2}\dot{X}_i^\top(t-T_i)k_J\dot{X}_i(t-T_i) \\ &+ \lambda\bar{T}_i\dot{X}_i^\top k_p\dot{X}_i - \lambda\int_{t-T_i}^{t}\dot{X}_i^\top(\sigma)k_p\dot{X}_i(\sigma)d\sigma\Big) + 2\lambda\bar{e}^\top k_p\dot{\bar{e}} \end{aligned} \tag{4.12}$$

Substituting for $M_i\dot{s}_i$ from the closed loop dynamics,

$$\dot{V}_1 = \sum_{i=s,m}\Big(s_i^\top\Big(-C_i s_i - K_i s_i + Y_i(\tilde{\Theta}_i + u_i) - J_i^T\bar{\tau}_i\Big) + \frac{1}{2}s_i^\top\dot{M}_i s_i + \lambda e_i^\top k_J\dot{e}_i + \frac{1}{2}\dot{X}_i^\top k_J\dot{X}_i$$
$$-\frac{1}{2}\dot{X}_i^\top(t-T_i)k_J\dot{X}_i(t-T_i) + \lambda\bar{T}_i\dot{X}_i^\top k_p\dot{X}_i - \lambda\int_{t-T_i}^{t}\dot{X}_i^\top(\sigma)k_p\dot{X}_i(\sigma)d\sigma\Big)$$
$$+ 2\lambda\bar{e}^\top k_p\dot{\bar{e}} - s_s^\top B s_s$$

Note that it can be shown that $Y_i\left(\tilde{\Theta}_i + u_i\right) \leq \epsilon_i\rho_i/4$. Moreover, using the skew symmetry property of Lagrangian systems the time derivative of the storage function can be simplified as,

$$\dot{V}_1 \leq \sum_{i=s,m}\Big(-s_i^\top K_i s_i - s_i^\top J_i^T\left(k_p\left(J_i s_i\right) - k_J\dot{e}_i\right) + \frac{\epsilon_i\rho_i}{4} + \lambda e_i^\top k_J\dot{e}_i + \frac{1}{2}\dot{X}_i^\top k_J\dot{X}_i$$
$$-\frac{1}{2}\dot{X}_i^\top(t-T_i)k_J\dot{X}_i(t-T_i) + \lambda\bar{T}_i\dot{X}_i^\top k_p\dot{X}_i - \lambda\int_{t-T_i}^{t}\dot{X}_i^\top(\sigma)k_p\dot{X}_i(\sigma)d\sigma\Big)$$
$$+ 2\lambda\bar{e}^\top k_p\dot{\bar{e}} - s_s^\top B s_s$$

where $\bar{\tau}_i$ were substituted. Substituting for s_i and simplifying the inequality will yield,

$$\dot{V}_1 \leq \sum_{i=s,m}\Big(\frac{\epsilon_i\rho_i}{4} - s_i^\top K_i s_i - \dot{X}_i^\top k_p\dot{X}_i + 2\lambda e_i^\top k_p\dot{X}_i - \lambda^2 e_i^\top e_i + \dot{X}_i^\top k_J\dot{e}_i$$
$$+\frac{1}{2}\dot{X}_i^\top k_J\dot{X}_i - \frac{1}{2}\dot{X}_i^\top(t-T_i)k_J\dot{X}_i(t-T_i) + \lambda\bar{T}_i\dot{X}_i^\top k_p\dot{X}_i - \lambda\int_{t-T_i}^{t}\dot{X}_i^\top(\sigma)k_p\dot{X}_i(\sigma)d\sigma\Big)$$
$$+ 2\lambda\bar{e}^\top k_p\dot{\bar{e}} - s_s^\top B s_s \tag{4.13}$$

Considering $2\lambda e_i^\top k_p \dot{X}_i$ and $2\lambda \bar{e}^\top k_p \dot{\bar{e}}$ one can show that,

$$
\begin{aligned}
&2\lambda e_s^\top k_p \dot{X}_s + 2\lambda e_r^\top k_p \dot{X}_r + 2\lambda \bar{e}^\top k_p \dot{\bar{e}} && (4.14)\\
&= 2\lambda k_p \left(\dot{X}_s^\top \left(X_r(t-T_r) - X_r \right) + \dot{X}_r^\top \left(X_s(t-T_s) - X_s \right) \right)\\
&= -2\lambda k_p \left(\dot{X}_s^\top \int_{t-T_r}^{t} \dot{X}_r(\sigma) d\sigma + \dot{X}_r^\top \int_{t-T_s}^{t} \dot{X}_s(\sigma) d\sigma \right) && (4.15)
\end{aligned}
$$

Further consider,

$$
\begin{aligned}
&\sum_{i=s,m} \left(\dot{X}_i^\top k_J \dot{e}_i + \frac{1}{2}\dot{X}_i^\top k_J \dot{X}_i - \frac{1}{2}\dot{X}_i^\top(t-T_i) k_J \dot{X}_i(t-T_i) \right)\\
&= \dot{X}_s^\top k_J (X_r(t-T_r) - X_s) + \dot{X}_r^\top k_J (X_s(t-T_s) - X_r) + \frac{1}{2}\dot{X}_s^\top k_J \dot{X}_s + \frac{1}{2}\dot{X}_r^\top k_J \dot{X}_r\\
&\quad - \frac{1}{2}\dot{X}_s^\top(t-T_s) k_J \dot{X}_s(t-T_s) - \frac{1}{2}\dot{X}_r^\top(t-T_r) k_J \dot{X}_r(t-T_r)\\
&= -\frac{1}{2}(\dot{X}_s - \dot{X}_r(t-T_r))^\top k_J (\dot{X}_s - \dot{X}_r(t-T_r)) - \frac{1}{2}(\dot{X}_r - \dot{X}_s(t-T_s))^\top k_J (\dot{X}_r - \dot{X}_s(t-T_s))\\
&= -\frac{1}{2}\dot{e}_s^\top K_J \dot{e}_s - \frac{1}{2}\dot{e}_r^\top K_J \dot{e}_r && (4.16)
\end{aligned}
$$

Using the above simplifications (4.15) - (4.16) in (4.13) results,

$$
\begin{aligned}
\dot{V}_1 \leq \sum_{i=s,m} \Big(&\frac{\epsilon_i \rho_i}{4} - s_i^\top K_i s_i - \dot{X}_i^\top k_p \dot{X}_i - \lambda^2 e_i^\top e_i - \dot{e}_i^\top k_J \dot{e}_i + \lambda \bar{T}_i \dot{X}_i^\top k_p \dot{X}_i\\
&- \lambda \int_{t-T_i}^{t} \dot{X}_i^\top(\sigma) k_p \dot{X}_i(\sigma) d\sigma \Big) - 2\lambda k_p \left(\dot{X}_s^\top \int_{t-T_r}^{t} \dot{X}_r(\sigma) d\sigma + \dot{X}_r^\top \int_{t-T_s}^{t} \dot{X}_s(\sigma) d\sigma \right)\\
&- s_s^\top B s_s
\end{aligned}
$$

$$
\begin{aligned}
&\leq \sum_{i=s,m} \left(\frac{\epsilon_i \rho_i}{4} - s_i^\top K_i s_i + (\lambda \bar{T}_i - 1) \dot{X}_i^\top k_p \dot{X}_i - \lambda^2 e_i^\top e_i - \dot{e}_i^\top k_J \dot{e}_i \right) - s_s^\top B s_s \\
&\quad - \lambda \int_{t-T_s}^{t} \dot{X}_s^\top(\sigma) k_p \dot{X}_s(\sigma) d\sigma - 2\lambda k_p \dot{X}_s^\top \int_{t-T_r}^{t} \dot{X}_r(\sigma) d\sigma \\
&\quad - \lambda \int_{t-T_r}^{t} \dot{X}_r^\top(\sigma) k_p \dot{X}_r(\sigma) d\sigma - 2\lambda k_p \dot{X}_r^\top \int_{t-T_s}^{t} \dot{X}_s(\sigma) d\sigma
\end{aligned}
$$

With the use of Lemma 4.1 it can be shown that

$$
\begin{aligned}
\dot{V}_1 \leq &\sum_{i=s,m} \left(\frac{\epsilon_i \rho_i}{4} - s_i^\top K_i s_i + (\lambda \bar{T}_i - 1) \dot{X}_i^\top k_p \dot{X}_i - \lambda^2 e_i^\top e_i - \dot{e}_i^\top k_J \dot{e}_i \right) - s_s^\top B s_s \\
&+ \lambda \bar{T}_s k_p \dot{X}_r^\top \dot{X}_r + \lambda \bar{T}_r k_p \dot{X}_s^\top \dot{X}_s.
\end{aligned}
$$

Thus obtaining,

$$
\begin{aligned}
\dot{V}_1 \leq &\frac{\epsilon_s \rho_s}{4} + \frac{\epsilon_r \rho_r}{4} - s_s^\top (K_s + B) s_s - s_r^\top K_r s_r - \lambda^2 e_s^\top e_s - \lambda^2 e_r^\top e_r + (\lambda \bar{T} - 1) \dot{X}_s^\top k_p \dot{X}_s \\
&+ (\lambda \bar{T} - 1) \dot{X}_r^\top k_p \dot{X}_r - \dot{e}_s^\top k_J \dot{e}_s - \dot{e}_r^\top k_J \dot{e}_r
\end{aligned}
$$

where $\bar{T} = \bar{T}_s + \bar{T}_r$. Selecting λ such that $\lambda \bar{T} < 1$ will ensure that the coefficient of $\dot{X}_i^\top \dot{X}_i$ are negative. Define $\xi = \frac{\epsilon_s \rho_s}{4} + \frac{\epsilon_r \rho_r}{4}$. The coefficients of the quadratic terms are collected to the diagonal matrix Q. Note that $Q > 0$, and it is bounded by $\lambda_{min}(Q) \leq ||Q|| \leq \lambda_{max}(Q)$. Here λ_{min} and λ_{max} are the minimum and maximum eigenvalues of Q, respectively. Thus we have,

$$
\dot{V}_1 \leq -z^\top Q z + \xi \leq -\lambda_{min}(Q) ||z||^2 + \xi
$$

It can be guaranteed that $\dot{V}_1 < 0$ when $||z|| > \left(\frac{\xi}{\lambda_{min}(Q)} \right)^{\frac{1}{2}}$. Here λ_{min} can be found as $\lambda_{min} =$

$\min\big((K_s + B), K_r, \lambda^2 k_p, k_J, k_p(\lambda(\bar{T}_s + \bar{T}_r) - 1)\big)$. *Therefore, the synchronization errors and trajectories are ultimately bounded.*

Next consider the case when there are bounded external forces applied to the teleoperation system. First, let us consider the case when the forces are passive.

Definition 4.1 *Let $r_i(t) = J_i(t)s_i(t)$ for $i \in \{r, s\}$. The applied external force on the rigid leader robot is defined as passive with respect to $-r_r(t)$ and human force $F_h(t)$ if there exists a constant $f_h \geq 0$ such that $\forall r_r(t) \in \mathbb{R}^3, \forall t > 0$ [88],*

$$-\int_0^t F_h^\top(\sigma) r_r(\sigma) d\sigma \geq -f_h.$$

Similarly, the external environmental force on the soft follower robot is passive with respect to $r_s(t)$ and environmental force $F_e(t)$ if there exists a constant $f_e \geq 0$ such that $\forall r_s(t) \in \mathbb{R}^{2n}, \forall t >$ 0,

$$\int_0^t F_e^\top(\sigma) r_s(\sigma) d\sigma \geq -f_e.$$

The following corollary shows that the teleoperation system remains bounded in the presence of external forces.

Corollary 4.1.1 *Consider the RTSBT framework (4.8) with additional control law (4.10). If there exist positive control parameters ϵ_s, ϵ_r, K_s, K_r, k_p, k_J, nominal parameter vectors Θ_s^0, Θ_r^0, constant bounds $\gamma_s, \gamma_r \geq 0$ for the uncertainties and $\lambda < 1/\bar{T}$, then in the presence of passive external forces, the trajectories are u.u.b.*

Proof of Corollary 4.1.1 *Consider the storage function as before with the addition of storage function for the passive external forces,*

$$V_2 = V_1 - \int_0^t F_h^\top(\sigma) r_r(\sigma) d\sigma + \int_0^t F_e^\top(\sigma) r_s(\sigma) d\sigma + f_e + f_e \tag{4.17}$$

where V_1 is defined as in (4.11). Observe that due to the passivity of the external forces V_2 is positive definite. Considering the time derivative, $\dot{V}_2$, Note that as soon as the closed loop dynamics are substituted, the additional terms cancels out and we are left with a time derivative exactly same as $\dot{V}_1$ in (4.12) and the proof follows.

Let us now consider the case when the external forces are not passive, but bounded. In the following Corollary it is shown that the trajectories remain ultimately bounded in this case.

Corollary 4.1.2 *Consider the RTSBT framework (4.8) with additional control law (4.10). If there exist positive control parameters ϵ_s, ϵ_r, K_s, K_r, k_p, k_J, nominal parameter vectors Θ_s^0, Θ_r^0, constant bounds $\gamma_s, \gamma_r \geq 0$ for the uncertainties and $\lambda < 1/\bar{T}$, then in the presence of non-passive bounded external forces F_e, F_h such that $||F_e|| \leq \mu_e, ||F_h|| \leq \mu_h$, the system trajectories are u.u.b.*

Proof of Corollary 4.1.2 *Consider the storage function as V_1 in (4.11). Following similar steps as in Theorem 4.1, in the presence of non-passive forces, the derivative of the storage function* *along the system trajectories can be simplified to,*

$$\begin{aligned}
\dot{V}_1 &\leq -z^\top Q z + \xi + s_s^\top J_s^\top F_e - s_r^\top J_r^\top F_h \\
&= -z^\top Q z + \xi + (-\lambda e_s + \dot{X}_s)^\top F_e - (-\lambda e_r + \dot{X}_r)^\top F_h \\
&= -z^\top Q z + \xi + [0, 0, -\lambda F_e, \lambda F_h, 0, 0, F_e, -F_h]^\top z
\end{aligned}$$

Using Cauchy–Schwarz inequality and with the bounds $||F_e|| \leq \mu_e, ||F_h|| \leq \mu_h$*,*

$$\begin{aligned}
\dot{V}_1 &\leq -\lambda_{min}(Q)||z||^2 + \xi + ||z||\sqrt{(1+\lambda^2)(F_e^2 + F_h^2)} \\
&\leq -\lambda_{min}(Q)||z||^2 + \xi(1+\lambda^2)||z||\sqrt{\mu_e^2 + \mu_h^2} \\
&= -\lambda_{min}(Q)||z||^2 + \xi + ||z||k_\mu
\end{aligned}$$

where the definition for $\lambda_{min}(Q)$ *from proof of Theorem 4.1 is used and define* $k_\mu = (1+\lambda^2)\sqrt{\mu_e^2 + \mu_h^2}$*. It can be easily shown that* $\dot{V}_1 < 0$ *is guaranteed when,*

$$||z|| > \frac{-k_\mu + \sqrt{k_\mu^2 + 4\lambda_{min}(Q)\xi}}{2\lambda_{min}(Q)}$$

from which u.u.b follows. This completes the proof.

4.3.3 Bilateral teleoperation in the presence of disturbances

This section studies the case in the presence of passive external forces and bounded disturbance forces $w_s, w_r \in \mathbb{R}^3$ acting upon the end effectors of the robots in the teleoperation system. In this case the system dynamics are,

$$M_s\ddot{q}_s + C_s\dot{q}_s + B\dot{q}_s + Kq_s + G_s = \tau_s - J_s^\top F_e + J_s^\top w_s$$
$$M_r\ddot{q}_r + C_r\dot{q}_r + G_r = \tau_r + J_r^\top F_h + J_r^\top w_r \tag{4.18}$$

Remark 4.2 *In most practical application scenarios these nominal disturbances w_s^0, w_r^0 are not known a priori and will be considered to be null and unmodelled. Moreover, especially for soft robots if these disturbances are not handled the highly deformable nature of the soft body would render the soft system unstable. On the other hand, it is still a challenge to simultaneously and comprehensively sense both body posture and interactions with the environment despite the progress in soft sensor technologies [89]. In this chapter I propose to handle these disturbances both in rigid leader and soft follower via robust control.*

In view of the above remark, for RTSBT in the presence of disturbances the control inputs (4.6) are modified for the rigid leader and soft follower robots as,

$$\tau_r = Y_r\hat{\Theta}_r - K_r s_r - J_r^\top \bar{\tau}_r + J_r^\top \hat{w}_r$$
$$\tau_s = Y_s\hat{\Theta}_s - K_s s_s - J_s^\top \bar{\tau}_s + J_s^\top \hat{w}_s \tag{4.19}$$

where $\hat{w}_s, \hat{w}_r$ are the estimated disturbance forces. The estimated disturbance is written as,

$$\hat{w}_j = w_j^0 - \delta_j$$

where it is assumed that there exists a bounded nominal disturbance w_j^0 and a bound $\eta > 0$ such that $||w_i^0 - w_i|| \leq \eta$. Moreover, δ_i is an additional control input designed to dominate the unknown disturbance defined as,

$$\delta_i = \begin{cases} -\eta_i \frac{J_i s_i}{||J_i s_i||} & \text{if} \quad ||J_s s_i|| > \varepsilon_i \\ -\eta_j \frac{J_s s_i}{\varepsilon_i} & \text{if} \quad ||J_s s_i|| \leq \varepsilon_i \end{cases} \tag{4.20}$$

for $\varepsilon_i > 0$. The closed loop system is then,

$$\begin{aligned} M_s \dot{s}_s + C_s s_s + B s_s + K_s s_s = & \quad Y_s \left(\tilde{\Theta}_s + u_s \right) + J_s^\top (\tilde{w}_s + \delta_s) - J_s^T \bar{\tau}_s - J_s^T F_e \\ M_r \dot{s}_r + C_r s_r + K_r s_r = & \quad Y_r \left(\tilde{\Theta}_r + u_r \right) + J_r^\top (\tilde{w}_r + \delta_r) - J_r^T \bar{\tau}_r + J_r^T F_h \end{aligned} \tag{4.21}$$

The stability of the proposed RTSBT in the presence of bounded external disturbances is discussed in the following result.

Theorem 4.2 *Consider the RTSBT framework (4.21) with additional control law (4.20). If there exist positive control parameters* ϵ_s, ϵ_r, K_s, K_r, k_p, k_J, *nominal parameter vectors* Θ_s^0, Θ_r^0, *constant bounds* $\gamma_s, \gamma_r \geq 0$ *for the uncertainties, bounded nominal disturbance* w_s^0, w_r^0 , *constant bounds* $\eta_s, \eta_r > 0$ *and* $\lambda < 1/\bar{T}$, *then in free motion* $(F_e = F_h = 0)$, *the system trajectories are u.u.b.*

Proof of Theorem 4.2 *Consider the storage function same as V_1 in (4.11). Following the proof of Theorem 4.1, the time derivative of the storage function along the system trajectories of (4.21),*

$$\dot{V}_1 \leq \sum_{i=s,r} \left(\frac{\epsilon_i \rho_i}{4} - s_i^\top K_i s_i + (\lambda \bar{T}_i - 1)\dot{X}_i^\top k_p \dot{X}_i - \lambda^2 e_i^\top e_i - \dot{e}_i^\top k_J \dot{e}_i \right) - s_s^\top B s_s$$
$$+ \lambda \bar{T}_s k_p \dot{X}_r^\top \dot{X}_r + \lambda \bar{T}_r k_p \dot{X}_s^\top \dot{X}_s + s_s^\top J_s^\top (\tilde{w}_s + \delta_s) + s_r^\top J_r^\top (\tilde{w}_r + \delta_r).$$

It can be shown that $s_i^\top J_i^\top (\tilde{w}_i + \delta_i) \leq \varepsilon_i \eta_i / 4$. Thus we have,

$$\begin{aligned}\dot{V}_1 \leq & \frac{\epsilon_s \rho_s}{4} + \frac{\epsilon_r \rho_r}{4} + \frac{\varepsilon_s \eta_s}{4} + \frac{\varepsilon_r \eta_r}{4} - s_s^\top (K_s + B) s_s - s_r^\top K_r s_r - \lambda^2 e_s^\top e_s - \lambda^2 e_r^\top e_r \\ & + (\lambda \bar{T} - 1)\dot{X}_s^\top k_p \dot{X}_s + (\lambda \bar{T} - 1)\dot{X}_r^\top k_p \dot{X}_r - \dot{e}_s^\top k_J \dot{e}_s - \dot{e}_r^\top k_J \dot{e}_r \end{aligned} \tag{4.22}$$

Same as in Theorem 4.1, selecting λ such that $\lambda \bar{T} < 1$ will ensure that the coefficient of $\dot{X}_i^\top \dot{X}_i$ are negative. Define $\xi_2 = \frac{\epsilon_s \rho_s}{4} + \frac{\epsilon_r \rho_r}{4} + \frac{\varepsilon_s \eta_s}{4} + \frac{\varepsilon_r \eta_r}{4}$. Collecting the coefficients of the quadratic terms to the diagonal matrix Q_2, one can represent

$$\dot{V} \leq -z^\top Q_2 z + \xi_2$$

and the ultimate boundedness can be shown following the same arguments as in Theorem 4.1.

Now consider the case when there are external passive forces exerted on the human and remote side.

Corollary 4.2.1 *Consider the RTSBT framework (4.21) with additional control law (4.20). If there exist positive control parametersϵ_s, ϵ_r, K_s, K_r, k_p, k_J, nominal parameter vectors Θ_s^0, Θ_r^0, constant bounds $\gamma_s, \gamma_r \geq 0$ for the uncertainties, bounded nominal disturbance w_s^0, w_r^0 , constant* *bounds $\eta_s, \eta_r > 0$ and $\lambda < 1/\bar{T}$, then in the presence of passive external forces, the system trajectories are u.u.b.*

Proof of Corollary 4.2.1 *Consider the storage function same as V_2 in (4.17). As noted in the proof of Corollary 4.1.1, the time derivative of the storage function will become exactly the same as the one in (4.22) as soon as the closed loop dynamics are substituted and the proof follows.*

Next, consider the RTSBT with bounded external disturbances in the presence if non-passive human and environmental forces.

Corollary 4.2.2 *Consider the RTSBT framework (4.21) with additional control law (4.20). If there exist positive control parameters ϵ_s, ϵ_r, K_s, K_r, k_p, k_J, nominal parameter vectors Θ_s^0, Θ_r^0, constant bounds $\gamma_s, \gamma_r \geq 0$ for the uncertainties, bounded nominal disturbance w_s^0, w_r^0, constant bounds $\eta_s, \eta_r > 0$ and $\lambda < 1/\bar{T}$, then in the presence of non-passive external forces, the system trajectories are u.u.b.*

Proof of Corollary 4.2.2 *Consider the storage function same as V_1 in (4.11). In the presence of non-passive forces, the derivative of the storage function along the system trajectories can be simplified to,*

$$\begin{aligned} \dot{V}_1 &\leq -z^\top Q_2 z + \xi_2 + s_s^\top J_s^\top F_e - s_r^\top J_r^\top F_h \\ &= -z^\top Q_2 z + \xi_2 + [0, 0, -\lambda F_e, \lambda F_h, 0, 0, F_e, -F_h]^\top z \end{aligned} \tag{4.23}$$

where Q_2 and ξ_2 are defined as in the proof of Theorem 4.2. Following similar steps as done in *the proof of Corollary 4.1.2, it can be shown that* $\dot{V}_1 < 0$ *when,*

$$||z|| > \frac{-k_\mu + \sqrt{k_\mu^2 + 4\lambda_{min}(Q_2)\xi_2}}{2\lambda_{min}(Q_2)}$$

which completes the proof.

Remark 4.3 *Consider the teleoperation is under external forces, for example in the case a human force is applied such that* $F_h = K_P(X_{des} - X_r) - K_D(\dot{X}_r)$ *with gains* K_P, K_D *to guide the rigid leader toward a specific target* X_{des}. *Note that the uub results provided above do not guarantee that the human succeeds in achieving the desired target under this forcing. In this case, there could be a non-zero force being applied at the leader robot even the position synchronization between the leader and the follower is achieved as the equilibrium state of the leader robot under human forces as* $z \to 0$ *is,*

$$Y_r(q_r, 0, 0, 0)\left(\Theta_r^0 - \Theta_r + u_r\right) = -J_r(q_r)^T F_h.$$

For a similar instance with an adaptive bilateral teleoperator such as the one considered in Chapter 2 and [67, 73], the equilibrium will be

$$Y_r(q_r, 0, 0, 0)\left(\hat{\Theta}_r(t) - \Theta_r\right) = -J_r(q_r)^T F_h.$$

Here, unless the parameter estimations have converged to the true values, the human force will be non-zero. Thus unable to ensure zero-force reflection. In this case, once the position synchronization is achieved, if the human force is set to zero ($F_h = 0$*) the adaptive teleoperator* *will suffer from a sudden forcing which will drive the positions of the robots to a different state until the position synchronization is achieved again. However, for the robust teleoperator, due to the above corollary the loss of forcing does not affect the uub guarantees.*

4.4 Sub-task control in null space

In this chapter, the soft follower manipulator is assumed to be redundant considering the configuration space with regard to the task space. Thus, $null(J_s)$ has a minimum dimension of $2n - 3$ which can be exploited to accomplish sub-task control as the task space motion is unaffected by the link velocity in the null space. This is done by designing the auxiliary function $\psi_s(t)$ in (4.3) appropriately as detailed in Appendix A.4. Here the subtask of achieving collision avoidance of the soft robot is considered.

Remark 4.4 *Utilization of the passivity-based robust control for bilateral teleoperation yields trajectories that are ultimately bounded. The fact that s_s is only bounded does not guarantee the convergence of sub-task tracking errors, but still can ensure boundedness of the errors. Moreover, it guarantees ψ_s is bounded. Based on the collision avoidance subtask focused in this chapter, the boundedness of ψ_s ensures collision free points of the soft robot do not enter the regions of safe distance r, provided existence of a collision free configuration and trajectory is feasible. This can be seen from equations (A.3) and (A.4). Therefore, this guarantees the collision avoidance sub task control.*

4.5 Numerical simulations

This section presents the simulation results for RTSBT of a three-DoF elbow manipulator (leader) and a two-link (4DoF) spatial soft robotic manipulator (follower) under gravity $g = 9.81$ pointing in the negative z-axis of the world frame. The base of the soft robot is fixed by orienting it down ($\alpha_{world} = \pi, \phi_{world} = 0$). For the soft robot, each link was considered to be identical with a length of $L_i = 1$m, cross-sectional diameter of $D_i = 0.1$m, normalized mass of $\rho_i = 1$kg, stiffness of $k_i = k = 0.4$ and damping of $\beta_i = \beta = 0.25$. For the elbow manipulator, the considered the link parameters are shown in Table 1. The home position of the elbow manipulator was pointing down.

Table 4.1: Link parameters for the leader elbow manipulator

Link	a_i	α_i	d_i	θ_i	mass
1	0	90°	0	q_{r_1}	-
2	1m	0	0	q_{r_2}	1kg
3	1m	0	0	q_{r_3}	1kg

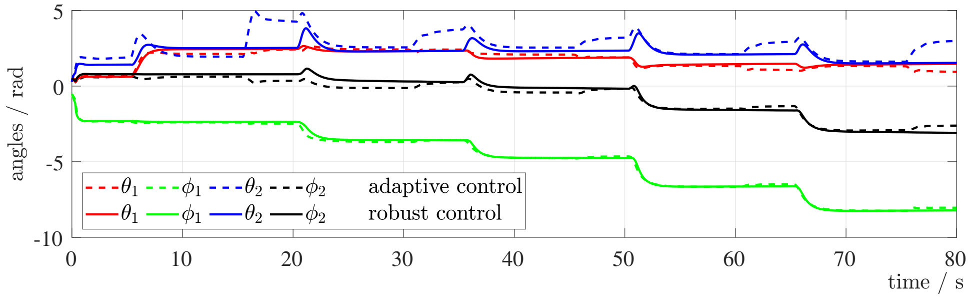
The simulations were run on Matlab Simulink with constant time delays $T_s = 0.4s$ and $T_r = 0.3s$, and control gains $\lambda = 1.4$, $k_J = 0.1$, $k_r = 1$ and $K_s = K_r = 2.5$. These were constant through all the simulations. The uncertainties were assumed in the mass of both robots, and stiffness and damping of the soft robot. Thus the parameter vectors were defined as $\Theta_s = [m_{s_1}, m_{s_2}, k, \beta]^\top$ and $\Theta_r = [m_{r_2}, m_{r_3}]^\top$. The nominal parameters for those were chosen as $\Theta_s^0 = [1.25, 1, 0.5, 0.75]^\top$ and $\Theta_r^0 = [0.5, 0.5]^\top$ and the bound on uncertainties was found to be $||\Theta_r - \Theta_r^0|| < 0.7566$, $||\Theta_s - \Theta_s^0|| < 0.7071$. Hence the uncertainty bounds were chosen as $\gamma_s = 1$, $\gamma_r = 1$ and $\epsilon_s = \epsilon_r = 0.01$ for robust control. Disturbance forces were added synthetically at the soft follower side as,

$$w_s(t) = \begin{bmatrix} 0.55\sin(2t) + 0.05\sin(5t) + 0.25\sin(17t) - 0.07\sin(13t) + 0.2\sin(23t) \\ 0.25\sin(t) + 0.15\sin(5t) - 0.15\sin(13t) - 0.47\sin(17t) + 0.17\sin(21t) \\ 0.1\sin(21t) - 0.05\sin(11t) + 0.35\sin(19t) - 0.4\sin(3t) + 0.12\sin(17t) \end{bmatrix} \tag{4.24}$$

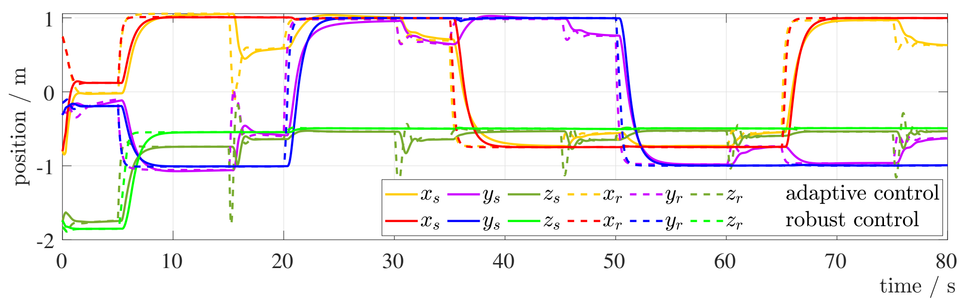
The nominal disturbance was assumed to be nonexistent $w_s^0 = [0,0,0]^\top$ with the bound found to be $||w_s^0 - w_s|| \leq 1.9264$. Hence, the bound $\eta_e = 2$ and the control parameter $\varepsilon_r = 0.01$ was selected. For the leader robot although no disturbance was added in the simulation the robust control framework with control parameters $\eta_r = 0.1$ and $\varepsilon_r = 0.1$ was employed. The initial conditions were set as $q_s(0) = [\pi/6, -\pi/6, \pi/12, \pi/6]^\top$, $\dot{q}_s(0) = [0,0,0,0]^\top$, $q_r(0) = [-\pi/16, \pi/3, \pi/12]^\top$ and $\dot{q}_r(0) = [0,0,0]$. In each of the simulations illustrated below it was considered that the same teleoperation task is executed by the human operator (i.e., leader robot). The task will be divided into phases of free motion (no external forces being applied) and forced motion (human operator exerts forces on the leader robot). The human force was modeled as a spring-damper forces with spring and damper gains set to 350 N/m and 80 Ns/m respectively in each x, y, z direction and were bounded $|F_{h_i}| \leq 25$N. The task schedule is shown in Table II.

4.5.1 Comparison with adaptive task space bilateral teleoperation

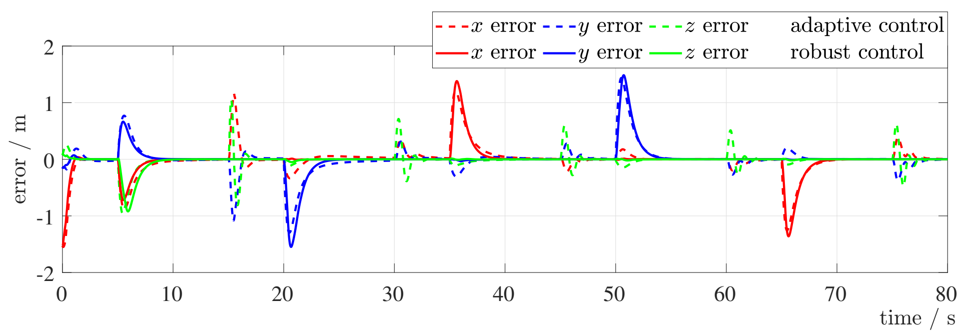
The performance of the RTSBT framework proposed in this work is compared with an adaptive bilateral teleoperation framework proposed in Chapter 2 considering the case without any disturbance forces. In the adaptive task space bilateral teleoperation framework, the parameter estimation vectors $\hat{\Theta}_i$ are considered to be time varying estimates of the parameter



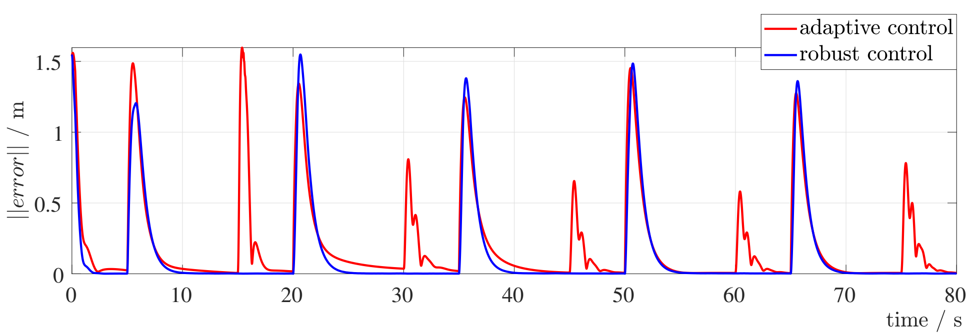
a) Evolution of the soft robot angles



b) Evolution of the positions



c) Evolution of the errors between leader and follower



d) Evolution of the error norms between leader and follower

Figure 4.3: Simulation results for adaptive vs robust bilateral teleoperation.

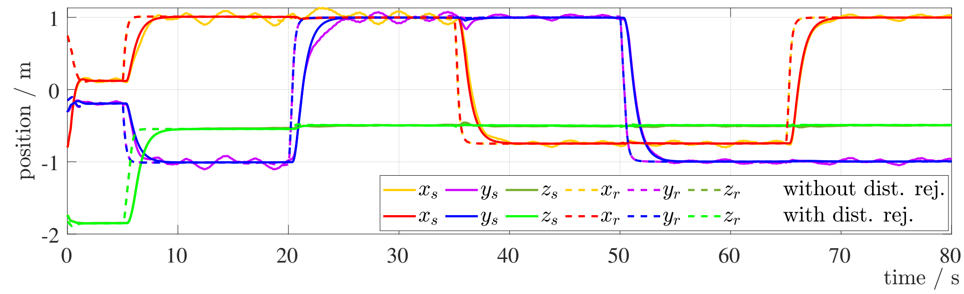
Table 4.2: Operation schedule of the bilateral teleoperation task

Time interval/ s	Operation
$0-5$	free motion
$5-15$	leader toward $[1,-1,-0.5]$
$15-20$	free motion
$20-30$	leader toward $[1,1,-0.5]$
$30-35$	free motion
$35-45$	leader toward $[-0.75,1,-0.5]$
$45-50$	free motion
$50-60$	leader toward $[-0.75,-1,-0.5]$
$60-65$	free motion
$65-75$	leader toward $[1,-1,0.5]$
$75-80$	free motion

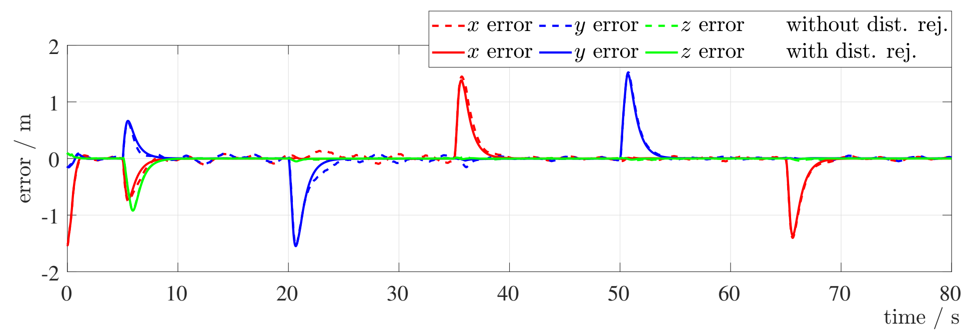
vector Θ_i for $i \in \{s, r\}$. In this case, the control input for the soft robot was defined as in (4.6) with the adaptation law for the parameter estimation defined as $\dot{\hat{\Theta}}_i = -\Gamma_i Y_i^\top s_i$ where Γ_i is a positive definite symmetric gain matrix. The reader is referred to Chapter 2 for more details on the passivity-based adaptive task space bilateral teleoperation framework. The simulation used $\Theta_s(0) = \Theta_s^0$, $\Theta_r(0) = \Theta_r^0$ as the initial parameter estimates and $\Gamma_r = \Gamma_s = 5$ as the adaptation gains, while keeping all the other control parameters as defined above. The comparison results for the teleoperation task as described above are illustrated in Figure 4.3. Note that there are jerky motions when the human forces are released (free motion starts) in the adaptive framework as discussed in Remark 4.3. The robust control framework does not exhibit such behavior.

4.5.2 Robust bilateral teleoperation without sub-task

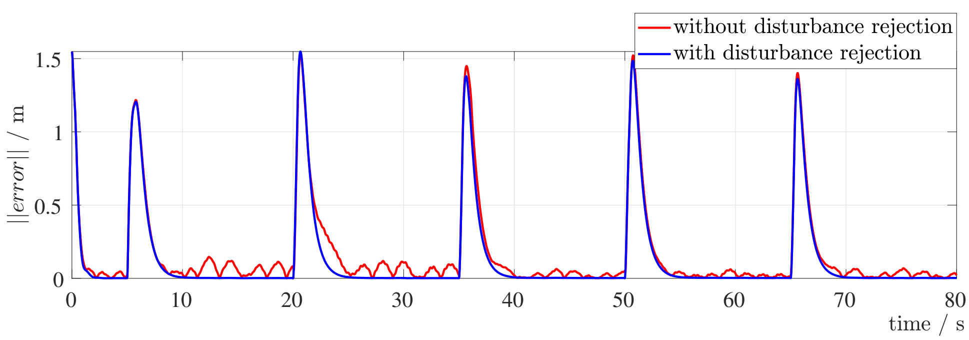
The first simulation considered the case without utilization of the collision avoidance sub-task. Results are shown in Figure 4.4. Clearly, the disturbance rejection control is able to handle the external disturbances and driving the error norms to the bounded set.



a) Evolution of the positions

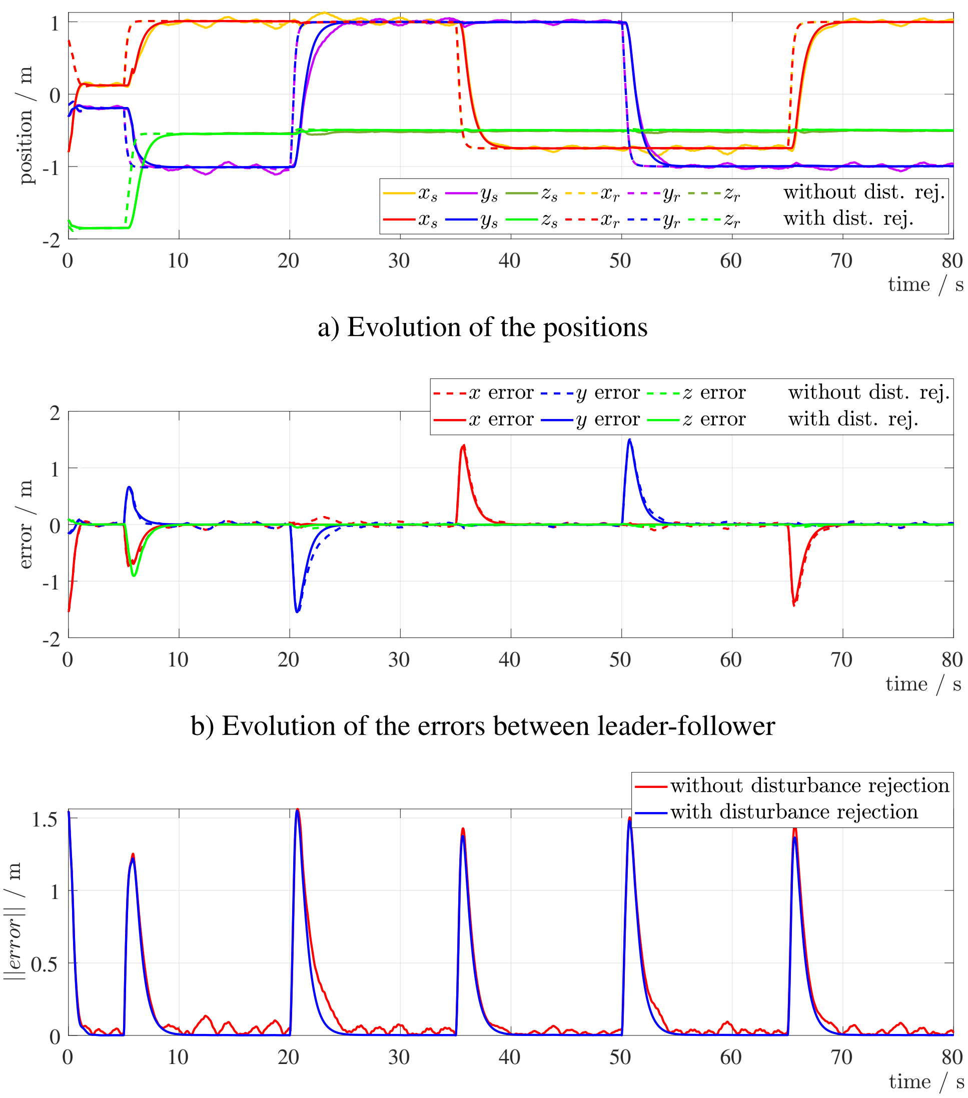


b) Evolution of the errors between leader-follower



c) Evolution of the error norms between leader-follower

Figure 4.4: Simulation results for RTSBT without sub-task.



a) Evolution of the positions

b) Evolution of the errors between leader-follower

c) Evolution of the error norms between leader-follower

Figure 4.5: Simulation results for RTSBT with collision avoidance sub-task.

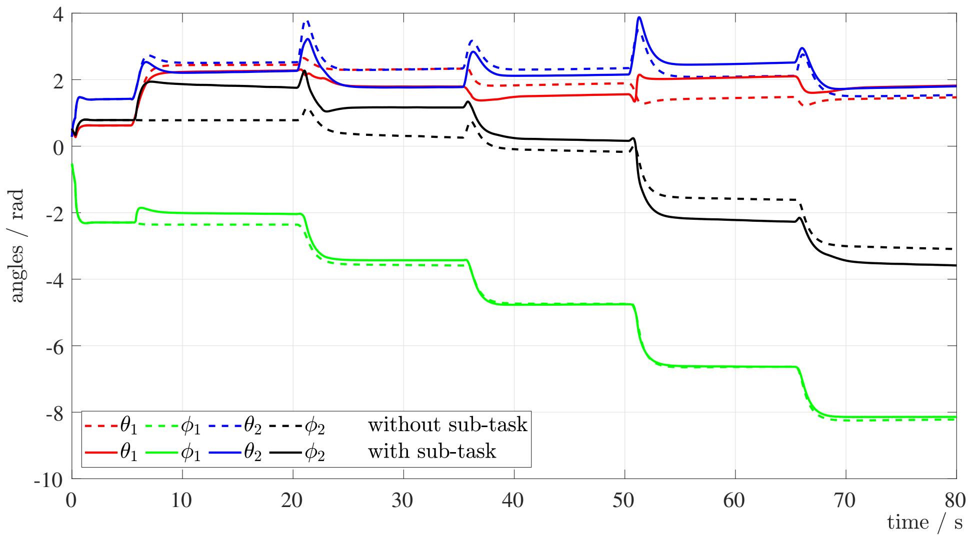
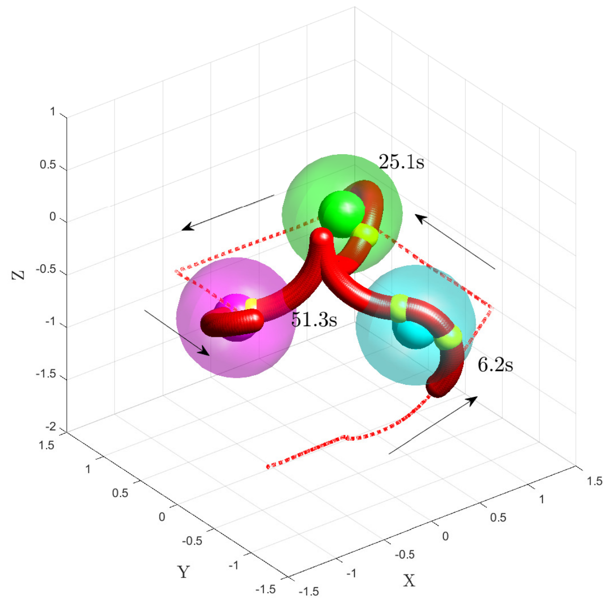


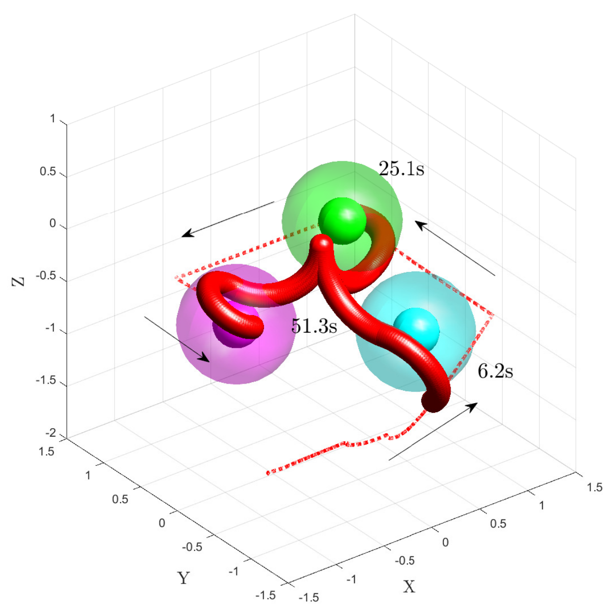
Figure 4.6: Evolution of the angles with and without the sub-task of collision avoidance in the presence of disturbance.

4.5.3 Robust bilateral teleoperation with collision avoidance sub-task

The sub-task of collision avoidance is utilized in this simulation. The points of interest were selected as the tip, middle point of the second link and the base of the second link. Three obstacles (point obstacles) were added in the follower environment with known locations at $[-0.5, 0.5, -0.8]^\top$, $[0.6, 0.5, -0.2]^\top$, $[0.6, -0.5, -0.8]^\top$. The smallest safe distance for all the obstacles was set to $r = 0.2m$ and the avoidance distance was set to $R = 0.6m$. The simulation results for this case of utilizing the sub-task control for collision avoidance are shown in Figure 4.5. The evolution of the angles with and without the sub-task of collision avoidance in the presence of disturbance is illustrated in Figure 4.6. A 3D visualization of the soft robot in the environment with the obstacles and the utilization of the sub-task control for obstacle avoidance is shown in Figure 4.7.



a) Teleoperation without the activation of collision avoidance sub-task.



b) Teleoperation with the activation of collision avoidance sub-task.

Figure 4.7: Soft robot visualization for the teleoperation task with and without the activation of collision avoidance subtask at 6.2s, 25.1s and 51.3s. Panel a) shows the collisions indicated in yellow blobs. Panel b) illustrates the successful avoidance of collisions with the obstacles for the same time instances with the subtask control activated. In the plots, the smallest safe distance $r = 0.2$m for obstacle regions are indicated in solid globes and the avoidance distance $R = 0.6$m is indicated in transparent globes.

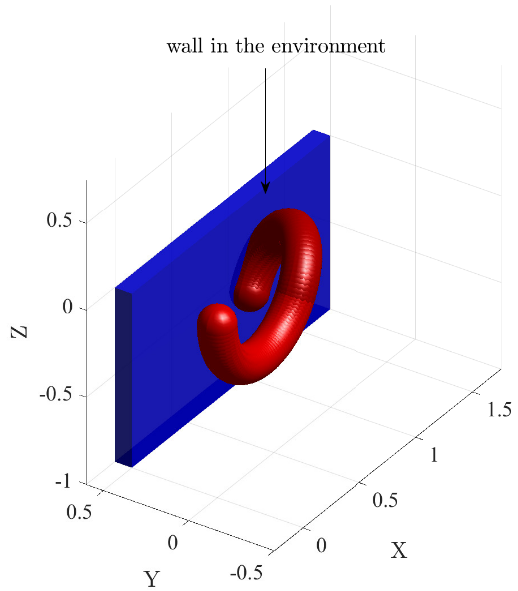


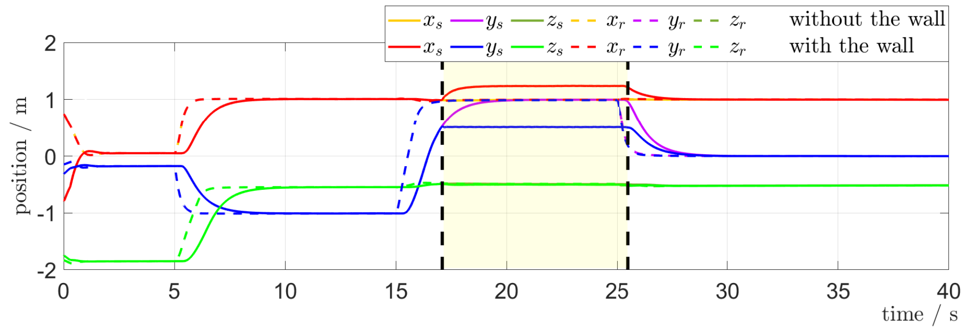
Figure 4.8: Illustration of the wall in the remote environment at $y = 0.5$m

4.5.4 Robust bilateral teleoperation with hard contact

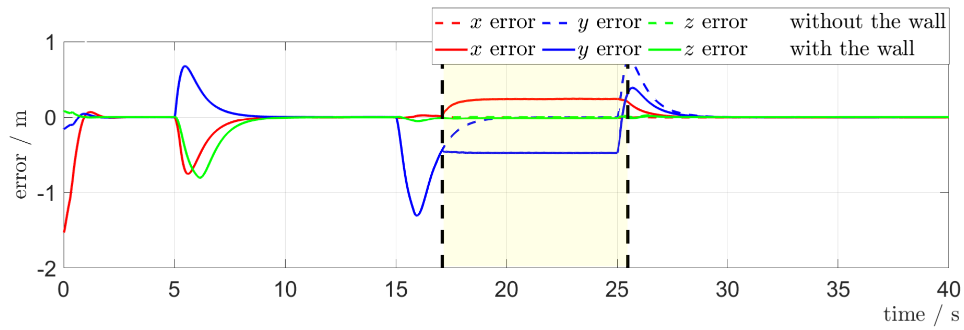
This scenario considers the environment to have a wall at $y = 0.5$ as shown in Figure 4.8. In this simulation, the robots are in free motion from $0 - 5$s. Next from $5 - 15$s the human operator moves the rigid leader toward $[1, -1, -0.5]^\top$. At 15s the human operator starts to move the rigid leader toward $[1, 1, -0.5]^\top$ and try to do this task until 25s. During this motion, the soft follower makes hard contact with the wall around 17s. The position errors do not approach the origin in this case as seen in Figure 4.9a-b. The forces felt at the leader side is shown in Figure 4.9c and the environment forces as felt at the soft follower robot is shown in Figure 4.9d.

4.6 Experimental Results

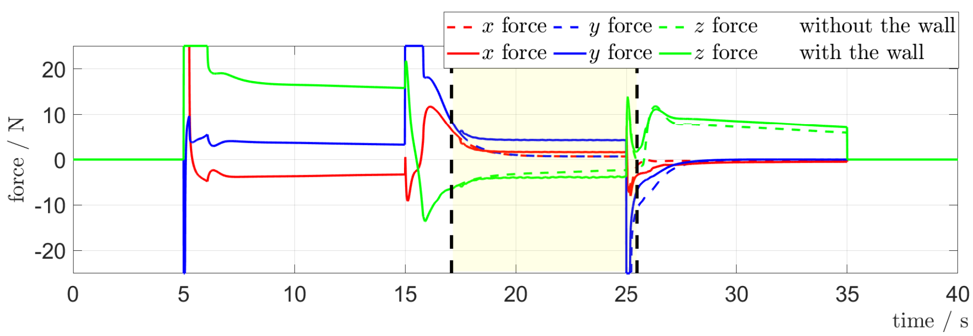
The planar version of the proposed robust bilateral teleoperation of soft robots was experimentally evaluated using a three-link pleated type planar soft follower robot and a virtual



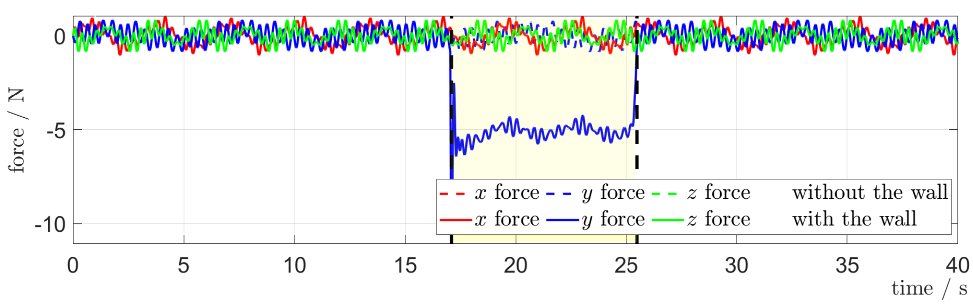
a) Evolution of the positions



b) Evolution of the errors between leader-follower



c) Evolution of the human forces to rigid leader



d) Evolution of the environmental forces to soft follower robot

Figure 4.9: Simulation results for the scenario which the soft follower robot contacts with wall. The interval in which the robot is in contact is highlighted in yellow.

two-DoF rigid link manipulator(using Matlab/Simulink 2021a in real time using the Simulink Desktop Real-Time library block). The soft robot control was run on Matlab 2021a on a different PC. The communication between the two systems, i.e. between the virtual rigid leader and soft robot was achieved through the TCP/IP protocol over WiFi. The network time delays were measured to be $T_m = T_s = 0.04s$. The experimental setup was similar to the one used in Chapter 2.6, while a slightly modified soft robot design was usd.

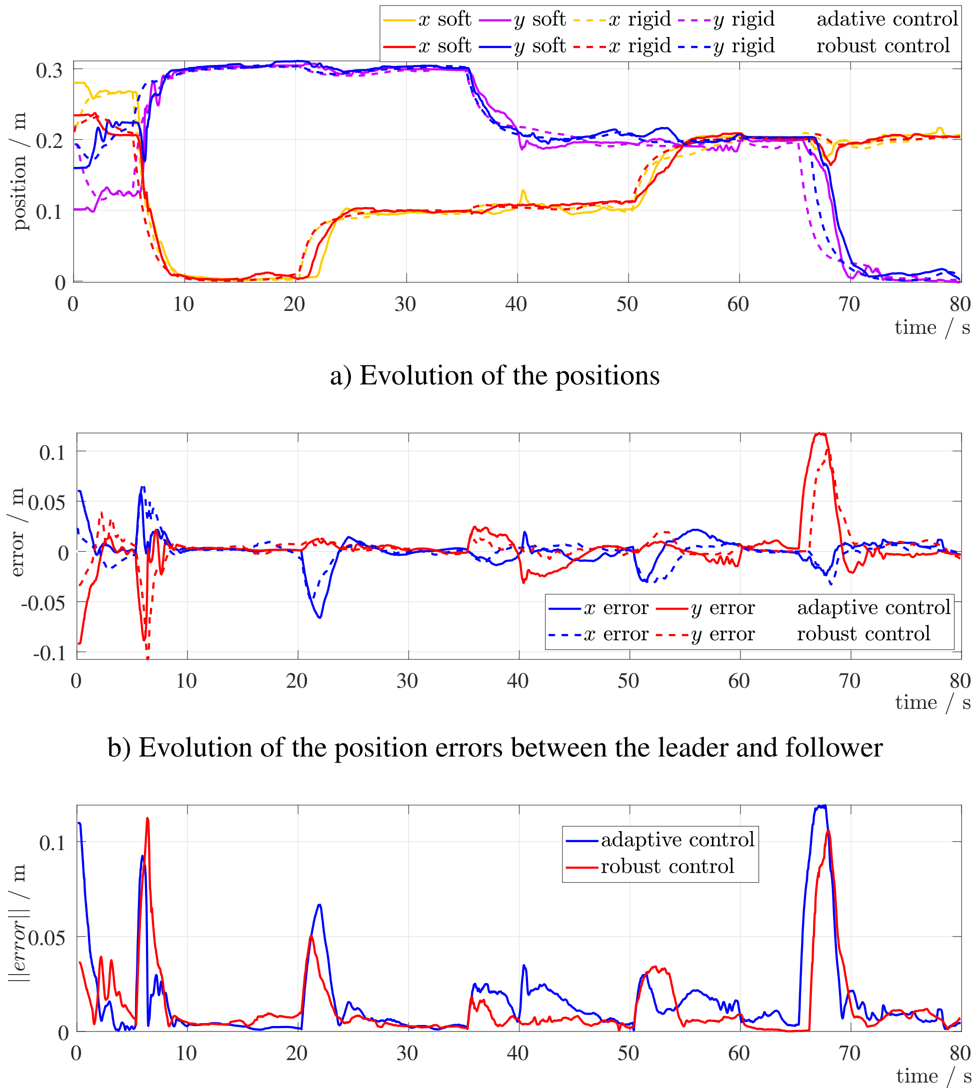
The segment lengths along the inextensible middle layer were measured to be $L_{s_i} = 0.125$ m. The segment masses $m_{s_i} = 0.080$ kg were measured prior to joining the segments together. The material properties of each segment were assumed to be identical and the nominal value of the torsional stiffness and damping of a segment $k = 0.75$Nm/rad and $d = 0.2$ Nms/rad, respectively, were calculated using a system identification process.

For the virtual rigid robot, the link lengths were chosen as $L_m = [0.2, 0.2]^\top$m. The masses and inertia were chosen as $m_m = [0.2, 0.2]^\top$kg and $I_m = [0.003, 0.003]^\top$kgm^2 respectively. The human force was modeled as a spring-damper forces with spring and damper gains set to 150 N/m and 150 Ns/m respectively.

For the soft follower robot, the uncertainty was assumed in the segment masses, end effector mass, torsional stiffness and torsional damping. Thus the parameter vector was chosen as $\Theta_s = [m_{s_1}, m_{s_2}, m_{s_3}, k, d]^\top$. The nominal parameter vector $\hat{\Theta}_s^0 = [0.5, 0.5, 0.5, 2, 1]^\top$ was set different from the measured nominal values. The uncertainty in the rigid leader robot was assumed to be in the rigid robot's link masses and in the link inertia. Thus the parameter vector was chosen as $\Theta_r = [m_{r_1}, m_{r_2}, I_{r_1}, I_{r_2}]^\top$. The nominal parameter estimate vector $\hat{\Theta}_r^0 = [0.05, 0.05, 0.001, 0.001]^\top$. The control gains were set to $\lambda = 1$, $k_r = 1$, $k_J = 5$ and $K_s = K_r = 2$.

In the experiment, 0-5s is in free motion for both the robots to synchronize. Then, at $t = 5$s the human operator exert forces to moves the leader toward $X_{des} = [0, 0.3]^\top$ and at $t = 15$s the robots are set to free motion. Then, at 20s the human operator exert a force to moves the leader to $X_{des} = [0.1, 0.3]^\top$ and again set to free at 30. At $t = 35$s the leader is moved toward $X_{des} = [0.1, 0.2]^\top$ and set to free motion at 45s. At $t = 50$s the leader is moved toward $X_{des} = [0.2, 0.2]^\top$ and the system is set to operate again in free motion at 60s. Finally At $t = 65$s the leader is moved toward $X_{des} = [0.2, 0]^\top$ and the system is set to operate in free motion at 75s.

The performance of the teleoperation system was compared against adaptive control based bilateral teleoperation framework as in Chapter 4 with the same level of uncertainty with the control gains for that controller set as $\Gamma_s = \Gamma_m = 0.1$, $\Lambda_s = \Lambda_m = 1.1$, $K_{rs} = K_{rm} = 1$, $K_s = K_m = 1.5$ and $K_{js} = K_{jm} = 10$. Figure 4.10 illustrates the tracking performance. Figure 4.11 depicts the configurations of the soft robot during the experiment for the robust controller proposed in this chapter.



a) Evolution of the positions

b) Evolution of the position errors between the leader and follower

c) Evolution of the position error norm between the leader and follower

Figure 4.10: Experimental results from planar soft robot teleoperation comparing robust and adaptive teleoperation frameworks

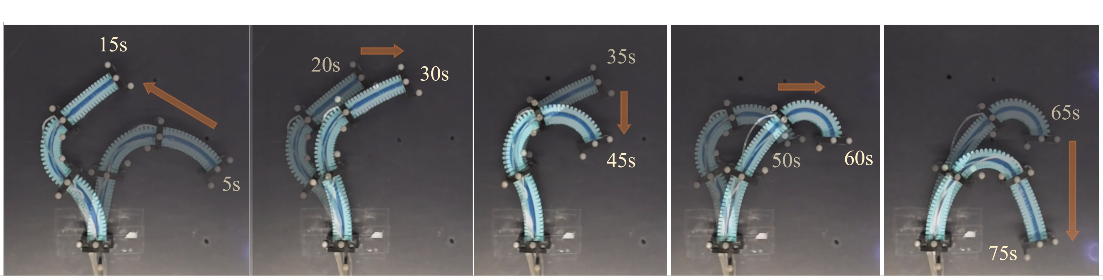


Figure 4.11: Photo sequence of the experiment for robust teleoperation

4.7 Summary

This chapter discussed robust task space bilateral teleoepration of soft robots using a rigid leader robot and a soft follower robot. Passivity-based robust control has been used to develop task space bilateral teleoperators in the presence of unmodelled external disturbances with dynamics uncertainty and constant asymmetric time delays. It was shown that the teleoperators are uniformly ultimately bounded under the influence of passive or non-passive external human or environment forces. Moreover, the null space velocity tracking of the redundant soft follower robot was used for achieving collision avoidance sub-task. The efficacy of the proposed framework is illustrated via extensive simulations and experiments.

Chapter 5: Adaptive Tracking Control of Soft Robots Using Integrated Sensing Skins

This chapter studies integrated estimation and control of soft robots. To that end, this chapter introduces a new method of estimating the degree of curvature of a soft robot using a stretchable sensing skin, which is a spray-coated piezoresistive sensing layer on a latex membrane. The mapping from the strain signal to the degree of curvature is estimated by using a recurrent neural network (RNN). Uni-directional bending as well as bi-directional bending of a single-segment soft robot is investigated. Moreover, an adaptive controller is developed to track the degree of curvature of the soft robot in the presence of dynamic uncertainties. Subsequently, using the integrated soft sensing skin, successful curvature tracking control of the soft robot is demonstrated experimentally.

5.1 Overview

An important recent focus of the soft robotics community has been the development of integrated sensors for soft robotic perception (e.g., [52, 58]). Integrated sensing would potentially enable a soft robot to perceive the world without external sensors. The sensory signals acquired from integrated sensors can then be utilized for state estimation and in closed loop control. Several methods have been proposed for developing integrated sensors for soft

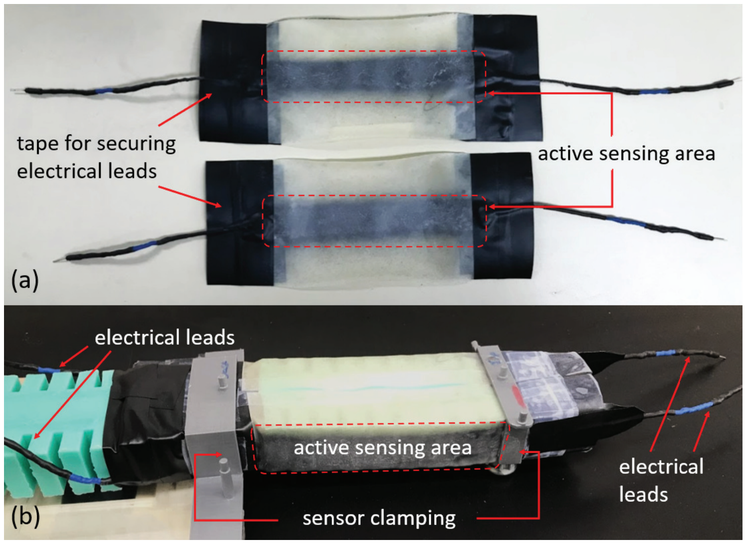


Figure 5.1: The soft sensing skins (a) and the soft robot retrofitted with the sensors (b).

robots [52, 90]. However, only a few sensing technologies have been demonstrated that are readily amenable to closed loop dynamic control for a wide range of soft robots [58].

This chapter investigates an integrated sensing and control framework for soft robots with a simple sensing skin that can be easily retrofitted to estimate the degree of curvature and employing an adaptive tracking controller. While this study considers a planar single-segment soft robot capable of bi-directional bending with a constant curvature along the length of the segment, the proposed advances could also be utilized for multi-segment 3D soft robots. The sensing skin consisted of a piezoresistive sensing layer spray coated onto a latex membrane [91, 92]. A strip-shaped sensing area was created, and electrical leads were attached at either end. The sensing skin and the soft robot retrofitted with the sensors are shown in Figure 5.1. A data driven model, namely a long short term memory (LSTM) network [93], which is a special RNN, was used to determine the relationship between the sensor signals and the degree of curvature. Both uni-directional bending and bi-directional bending were investigated. The utilization of the proposed integrated sensing strategy in an adaptive control framework [61] for dynamic tracking control of soft robots is successfully demonstrated in experiments. The adaptive controller was developed to track the degree of curvature of the soft robot assuming uncertainty in the dynamic parameters of the soft robot.

To the best of my knowledge, this chapter demonstrates the first steps toward utilizing retrofitted soft sensor skins for degree of curvature estimation in adaptive tracking control of soft robots for bi-directional bending. The results semming from this work was partly presented in [94].

The rest of the chapter is organized as follows. Section 5.2 discuss the related work on embedded sensing technologies for soft robots. The adaptive tracking control framework is discussed in Section 5.3. Section 5.4 introduces the soft sensing skin and the soft robot, and discusses degree of curvature estimation using integrated sensing skins. The experimental results for curvature tracking control using the integrated sensing are presented in Section 5.5 and the results are discussed in Section 5.6.

5.2 Related work on embedded sensing for soft robots

Over the past decade, a considerable amount of work on developing embedded sensing for soft robots has emerged. Some studies demonstrated the use of commercially available flex sensors embedded in a soft robot to measure the bending of the body. In one study [48], the integration of commercial flex sensors within a soft bending module actuated by pressure-driven fluidic actuators was attempted. Commercial flex sensors were used in another study [49] to estimate bending angle through a data-driven approach, in which the bending angle control was achieved by utilizing the predicted angle in a classical PID controller (heuristically tuned, rather than model-based). Flex sensors were also used in soft elastomer composite actuators for bending angle estimation [50, 51] . The estimates were then utilized in learning based control frameworks. In another work [95], a flex sensor was used for bending angle estimation, which was then employed in a back-stepping control algorithm for bending angle tracking. More recently, a data driven model was proposed to estimate the bending angle from a commercially available flex sensor as well as the pressure data [96]. The paper demonstrated the use of the estimates in a model-free static controller for bending angle control.

One drawback of the commercial flex sensors is that they stiffen the soft bodies since the flex sensors are not as soft as the soft robot body [52, 53]. Specifically, the flex sensors bend but they do not stretch. Therefore, they are embedded in the non-stretching region at the center of the robot segment [54]. Flex sensors could not be used in extensible soft robots because of their lack of stretchability. Thus, some groups have focused on developing soft embedded sensors that do not impact the mechanical compliance of the soft robots. Such embedded sensors for estimating soft robot position, actuation pressure, and force sensing have been fabricated recently [53, 54, 55]. A method to fabricate soft somatosensitive actuators by embedding 3D printed ionically conductive gels was proposed in [90]. In [97], a McKibben-type actuator with an embedded soft sensor was fabricated using a self-coagulating conductive Pickering emulsion and was used in closed loop control for slow movements with considerable error. In [98], a differential sensing method for the application of soft robot angle sensing using an embedded coiled conductive polymer fiber was proposed. A closed-loop multidimensional angle control system based on PID control using the differential sensing method was then developed to verify the sensing performance. The review paper [56] discusses soft pneumatic actuators fabricated entirely with additive manufacturing methods and suggests learning based control for soft robots with self-sensing capability.

Recent efforts in embedded sensing technology have used polydimethylsiloxane (PDMS) filled with carbon nanotubes (cPDMS) [52, 57]. The resistance of these polymers increases with strain [99]. By embedding cPDMSs in the soft robot body and measuring the resistance of these areas, the bending of the soft body could be estimated. In [52], the authors discussed a strategy for data-driven multi-modal sensing, namely the robot tip position and exerted force at the tip, using a cPDMS embedded sensor. The fabrication of cPDMS soft skins and their use for tactile sensing for haptic visualization was discussed in [57] .

The authors in [58] used off-the-shelf conductive silicone elastomer sheets laser cut into Kirigami patters and bonded to the soft robot skin as soft piezoresistive silicone sensors. Using these sensors, the steady state 3D configuration of the soft robot was predicted using a trained RNN. This strategy has been used for developing data–driven disturbance observers for estimating external forces on soft robots [59].

5.3 Adaptive curvature tracking controller

This section develops the adaptive control framework for curvature tracking. While the sensing skin and the experimental results are developed for a single segment soft robot, the approach is scalable, and hence the general multi segment dynamics and control strategy is discussed here. The dynamics of the soft robot are formulated as a Lagrangian system through the dynamically consistent *augmented formulation*[22] as described in Section 2.2. Specifically, the PCC model (2.3) will be used in this controller design.

The motivation for developing an adaptive controller is that the estimated model for the soft robot in the form of a PCC segment based rigid robot manipulator is not exact. Also the parameters of the model are not known precisely. Therefore a control mechanism that adapts the parameters as the soft robot operates would be beneficial for good performance. It should be noted that the main objective of the controller is to track the desired curvature. If the PCC model (2.3) is viewed as a rigid manipulator model, one can apply classical methods developed for rigid robots as shown in [61] to develop the adaptive tracking controller.

Define the degree of curvature error vector as,

$$\tilde{q}_s(t) = q_s(t) - q_d(t)$$

where $q_d(t)$ is the desired curvature. Define the virtual reference trajectory

$$\dot{q}_r(t) = \dot{q}_d(t) - \lambda \tilde{q}_s(t)$$

and let

$$s(t) = \dot{\tilde{q}}_s(t) - \lambda \tilde{q}_s(t),$$

where λ is a positive definite parameter matrix which needs to be tuned.

Denote the *equivalent parameter vector* of the model as Θ_s, whose elements are combinations of the variables m_i, L_i, K_s, and D_s. Note that the K_s and D_s terms will be explicitly included in Θ_s. Using the properties of Lagrangian systems (see Appendix A.2), define the regressor $(Y_s(q_s, \dot{q}_s, \dot{q}_r, \ddot{q}_r))$ and parameter (Θ_s) vector pair for the augmented soft robot model,

$$Y_s\Theta_s = M_s\ddot{q}_r + (C_s + D_s)\dot{q}_r + K_s q_s + G_s.$$

The estimated equivalent parameter vector is denoted by $\hat{\Theta}_s$, and hence the estimation error is defined as $\tilde{\Theta}_s = \hat{\Theta}_s - \Theta_s$. Now the control law is proposed as,

$$\tau_s = Y_s\hat{\Theta}_s - K_D s, \tag{5.1}$$

where K_D is a gain term that needs to be tuned. The adaptation law with the positive definite adaptation gain matrix Γ_c is

$$\dot{\hat{\Theta}}_s = -\Gamma_c Y_s^T s. \tag{5.2}$$

The stability of the designed controller (5.1)-(5.2) can be demonstrated using Lyapunov analysis [61, 74] which is sketched here.

Consider the Lyapunov function candidate

$$V(t) = \frac{1}{2}[s^T M_s s + \tilde{\Theta}_s^T \Gamma_c^{-1}\tilde{\Theta}_s] \tag{5.3}$$

where Γ_c is chosen to be a symmetric positive definite matrix. Therefore given the positive definiteness of M_s (*Property A.1*), the Lyapunov function candidate is positive definite.

Differentiating $V(t)$ with respect to time,

$$\begin{aligned}\dot{V}(t) &= s^T M_s \dot{s} + \frac{1}{2} s^T \dot{M}_s s + \dot{\hat{\Theta}}_s^T \Gamma_c^{-1} \tilde{\Theta}_s \\ &= s^T M_s \left(\ddot{q}_s - \ddot{q}_r\right) + \frac{1}{2} s^T \dot{M}_s s + \dot{\hat{\Theta}}_s^T \Gamma_c^{-1} \tilde{\Theta}_s \\ &= s^T \left(\tau - \left(C_s(q_s, \dot{q}_s) + D_s\right)\left(s + \dot{q}_r\right) - K_s q_s - G_s - M_s \ddot{q}_r\right) + \frac{1}{2} s^T \dot{M}_s s + \dot{\hat{\Theta}}^T \Gamma_c^{-1} \tilde{\Theta}_s\end{aligned} \tag{5.4}$$

then using the *Property* A.2, $\dot{M}_s - 2C_s$ is skew symmetric and, substituting the proposed control law (5.1) and the adaptation law (5.2) in (5.4) yields

$$\begin{aligned}\dot{V}(t) &= s^T \left(Y_s \hat{\Theta}_s - K_D s - Y_s \Theta_s\right) + \left(-\Gamma_c Y_s^T s\right)^T \Gamma_c^{-1} \tilde{\Theta}_s \\ &= -\, s^T K_D s\end{aligned} \tag{5.5}$$

which shows that $\dot{V}(t)$ is negative semi-definite. Now invoking Barbalat's Lemma it can be shown that the convergence of the tracking error is guaranteed while the system remains globally stable.

5.4 Integrated sensing

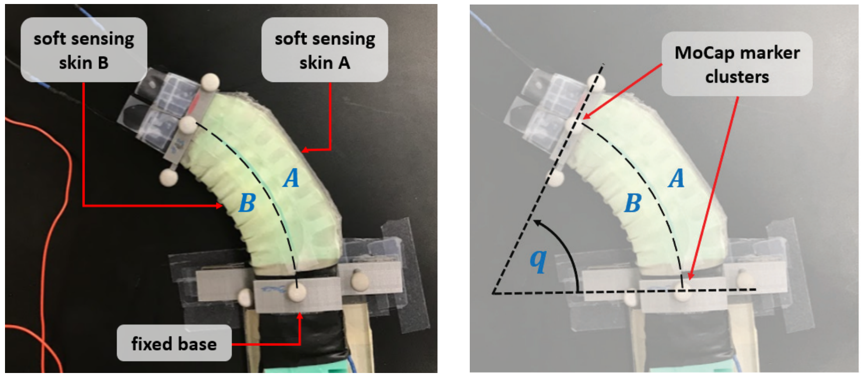
This section discusses the proposed method for degree of curvature estimation using soft sensing skins retrofitted onto a soft robot. First, the soft sensing skin [91, 92] is described and the soft robot is characterized. Then the experimental setup is introduced and the approach for degree of curvature estimation is discussed.

5.4.1 Soft sensing skin

The soft sensors used in this work were fabricated by spray coating a stretchable piezoresistive sensing layer in the shape of a strip onto a latex membrane. The coating consisted of a latex host filled with exfoliated graphite (EG). Electromechanical connections to the sensing layer were made using carbon fiber yarn attached with the same latex/EG solution serving as a "glue" [91]. A top coating of latex was added to protect the sensing layer from mechanical damage. Further, adhesive tape was used at the ends of the sensor strips to protect the electromechanical connections of the sensor. The carbon fiber yarn was joined to a copper wire by winding the copper wire around the carbon fiber yarn and then using a heat-shrink tubing to secure the joint.

Upon stretching the sensor, the resistance of the film increases. The response of these sensors is substantially linear over a large range of strain [100], and the gauge factor (GF), or sensitivity, is on the order of 10 (ΔR/R = GF*ϵ, where ΔR is the change in resistance, R is the original resistance, and ϵ is the strain).

Two stretchable sensing skins were retrofitted over the soft robot, one on each side, to measure strain when the segment was bending. Figure 5.1 (a) shows an image of the soft sensing skins. Figure 5.1 (b) shows the skins over the outer surface of an actuated segment, mechanically held in place using rectangular rings around the un-actuated ends of the segment; these rings are typically used to hold the markers for the motion capture system, and here served both functions. Two voltage divider circuits were used to measure the resistance of the sensing skins, and an Arduino board serially transmitted these as analog signals, which are referred to as "raw strain signals".



(a) (b)

Figure 5.2: Soft robot retrofitted with the stretchable sensing skins. The *compartments* of the soft robot and the sensing skins are labeled in panel (a). Panel (b) shows the degree of curvature and the markers placed for the motion capture (MoCap) system.

5.4.2 The soft robot

The soft robot used in this chapter is the same soft robot described in Section 2.6.1.1, except that only the distal segment was actuated. The base of this distal segment was fixed and the segment was constrained to move on a horizontal table. This actuated single-segment is referred to as the soft robot for this study. The segment had two *compartments*, named A and B as shown in Figure 5.2, that were individually actuated pneumatically, and were assumed to deform with a constant curvature along the length of the segment under the applied pressure. The *middle layer* of the segment (the joint between the two chambers) was inextensible due to the restrained material layer.

The segment length along the inextensible *middle layer* was measured to be $L_1 = 124$ mm. The segment mass m_1 is uncertain due to the retrofitting the soft sensing skin on the original design, for which the segment mass 0.110 kg was measured prior to joining the segments together.

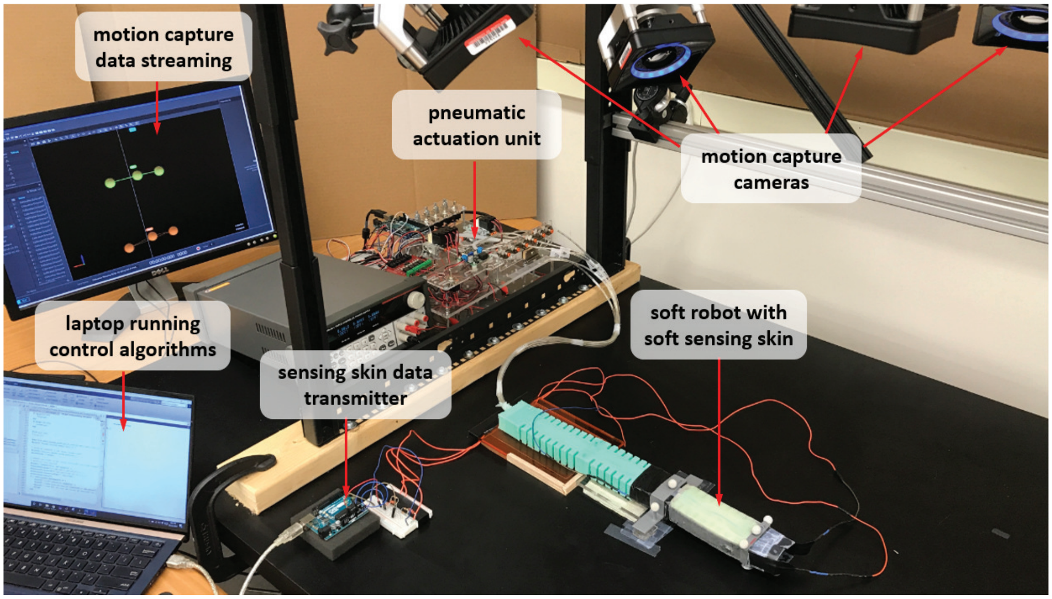


Figure 5.3: The experimental setup for testing the soft sensing skins

5.4.3 Experimental setup

The experimental setup, shown in Figure 5.3, consisted of the single-segment soft robot retrofitted with the sensing skin, the pneumatic actuation unit, an Arduino board to acquire the sensor skin strain signal, an OptiTrack motion capture system for ground truth measurements, and an i7 16GB RAM Windows 10 laptop to train the neural networks and run the control algorithm on MATLAB 2019a. Before each experiment, either data collection or control, two cycles of inflating and deflating each compartment for 5 s intervals were carried out to eliminate *first cycle effects* in the elastomers.

5.4.4 Degree of curvature estimation

The degree of curvature estimation is considered for two cases. First, uni-directional bending of the soft robot with the utilization of a single soft sensing skin on one side. Second, bi-directional bending utilizing both skins on two sides. It is assumed that the soft robot

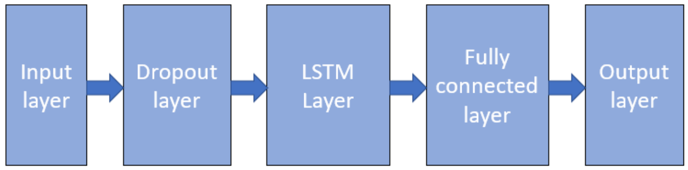
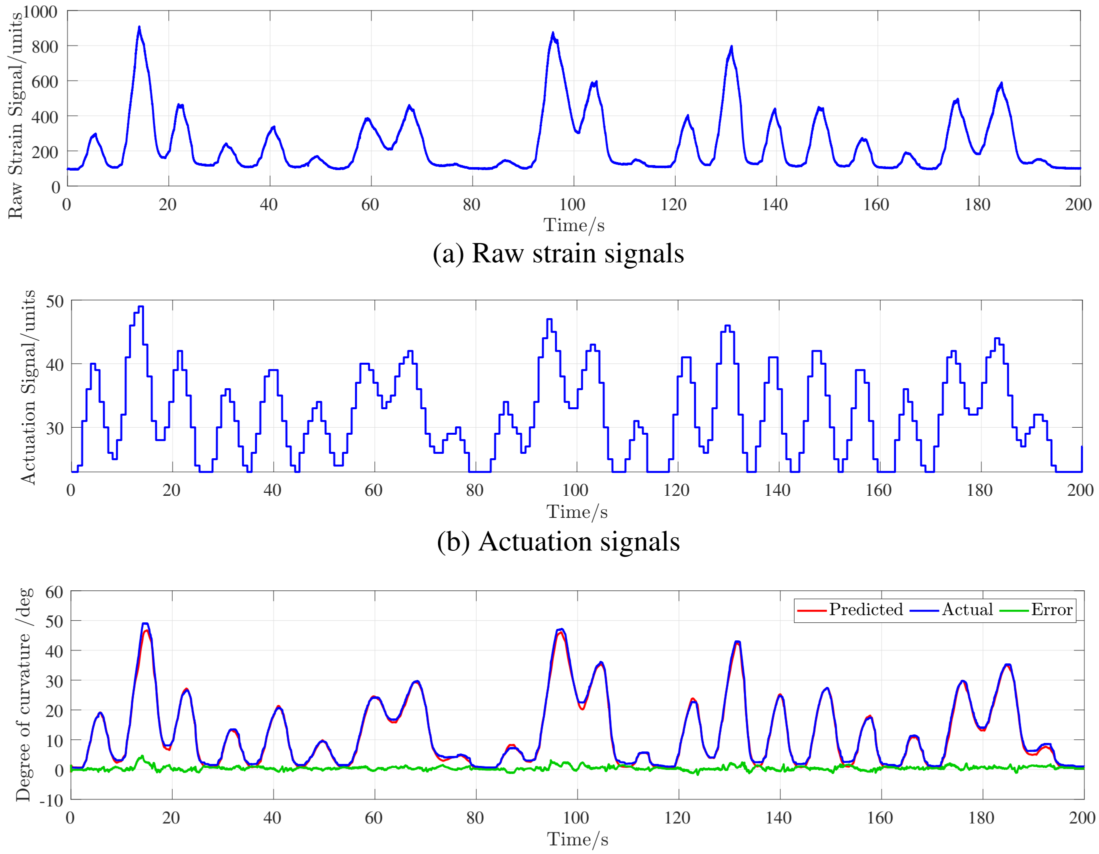


Figure 5.4: The Neural Network architecture used in this work

segment has the same curvature along its length (CC assumption). A data driven approach was used to identify the relationship between the strain signals from the sensors and the degree of curvature. Specifically, an RNN named Long Short Term Memory (LSTM) network [93] is utilized to learn the time series mapping.

For both the scenarios, the same network architecture designed using the MATLAB Deep Learning Toolbox was used. A dropout layer with a rate of 0.1 after the input layer was used to prevent over-fitting and make predictions more robust to noise. Next, an LSTM layer was used. The number of hidden units for the this layer was selected to be as small as possible to prevent overfitting via a validation set. Then a fully connected layer was added to compute the outputs. The network architecture of the LSTM network used in this work is shown in Figure 5.4.

For the uni-directional bending case, the actuator signal (PWM signal) for the actuated *compartment A* and the raw strain signal from the soft sensing skin A were the inputs to the network. For the bi-directional bending case, the actuator signals for both compartments and the raw strain signals from the two sensing skins were the inputs. The networks' output was the degree of curvature. The degree of curvature measured using the motion capture (MoCap) system was used as the ground truth when training the networks. Both the networks were trained using the Adam optimizer. L2 regularization with the default value (0.0001) was used. Further, two separate validation sets were used: an α-validation set with a frequency of 25 and patience of 5 to



(a) Raw strain signals

(b) Actuation signals

(c) Test set prediction from learned model

Figure 5.5: Learned model test set performance: Uni-directional bending

minimize over-fitting by early stopping, and a β-validation set for selecting the number of hidden units for the LSTM layer by manually inspecting the root mean squared error (RMSE) value for the β-validation set after the training had stopped.

5.4.4.1 Uni-directional bending

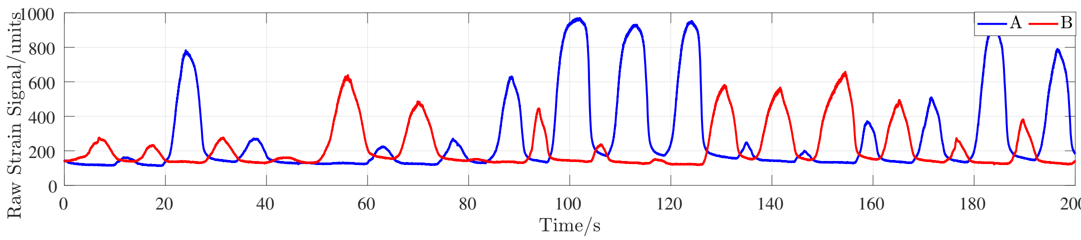
For testing the uni- directional bending, only the *compartment A* of the soft robot was actuated, and the strain signals from the soft sensing skin retrofitted on the *compartment A* were used. Eight experiments were conducted to collect data for training. A random actuation pattern was generated at a rate of approximately 1 Hz to actuate the soft robot and each experiment was run for a period of $2-3$ minutes. For each experiment, the actuation signal, the strain signal, and the actual degree of curvature were recorded at a rate of 85 Hz. The collected data were joined together later to constitute the total data set, which consisted of $115,410$ data points. This data set was then divided into a training set of $80,786$ points and two validation sets, α,β-validation sets, of $17,312$ points each, from which the network was trained. The optimum number of hidden units for the LSTM layer was 30.

Once the RNN was learned, the degree of curvature estimation performance was evaluated in real time for a random actuation pattern. The predicted degree of curvature by the LSTM network superimposed with the actual values for this experiment are illustrated in Figure 5.5 along with the raw strain signals and the actuation signals. The RMSE for this test experiment was $0.86°$.

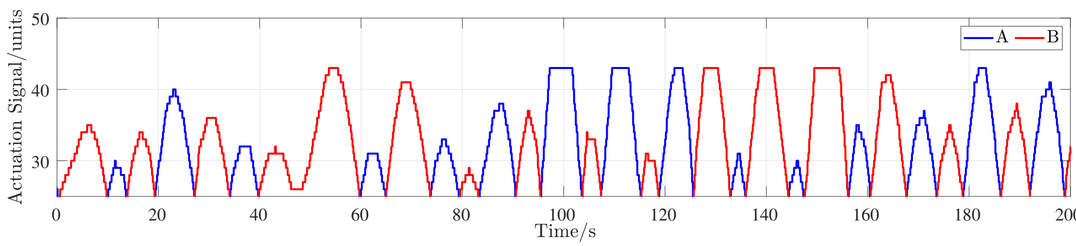
5.4.4.2 Bi-directional bending

For this case both *compartments A* and *B* were actuated, and the strain signals from both the sensing skins were used. The training data set was collected at a rate of 60 Hz by conducting seven experiments using randomly generated actuation patterns at a varying rate of $1-4$ Hz. The cumulative length of the experiments was 30 mins, resulting in a total data set that consisted of $107,090$ data points. This data set was then divided into a training set of $74,962$ points and two validation sets, α,β-validation sets, of $16,064$ points each, from which the network was trained. The optimum number of hidden units for the LSTM layer was 30.

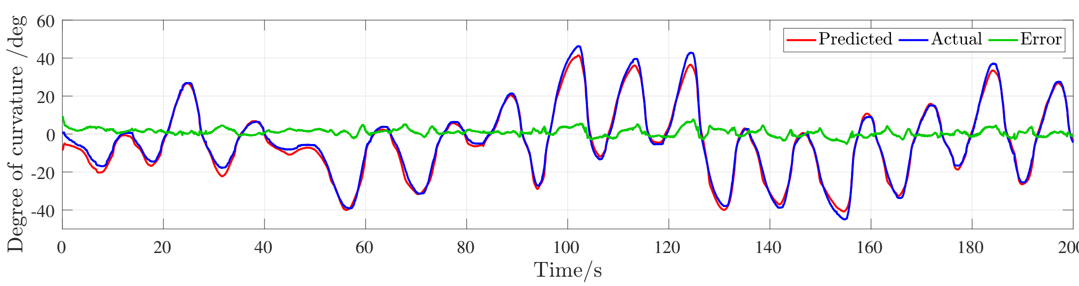
The degree of curvature estimation performance of the learned model was evaluated in real time for a random actuation pattern. The predicted degree of curvature by the LSTM network



(a) Raw strain signals



(b) Actuation signals



(c) Test set prediction from learned model

Figure 5.6: Learned model test set performance: Bi-directional bending

superimposed with the actual values for this experiment are illustrated in Figure 5.6 along with the raw strain signals and the actuation signals. The RMSE for this test experiment was $1.95°$.

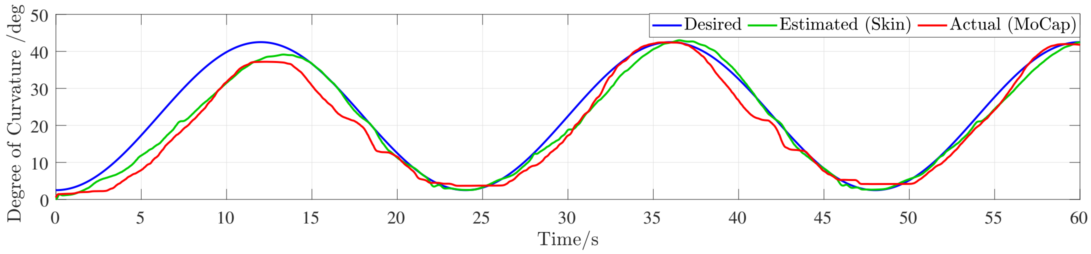
5.5 Experimental results

This section illustrates the efficacy of the integrated sensing strategy for dynamic tracking control of soft robots using the adaptive control framework (5.1)-(5.2) for degree of curvature tracking. The results are reported for both uni-directional and bi-directional bending. The tracking errors are computed and related to the degree of curvature ground truth measured by the MoCap system.

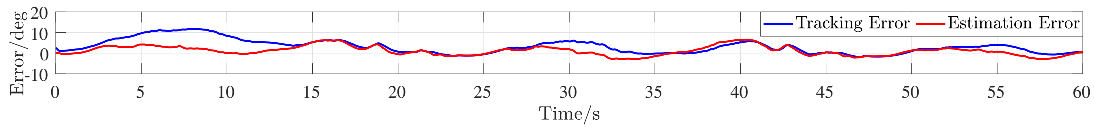
Uncertainty was assumed in the segment mass, torsional stiffness and torsional damping. Thus the parameter vector was chosen as $\Theta_s = \left[\mathrm{m}_1\mathrm{L}_1^2,\ \mathrm{K}_s,\ \mathrm{D}_s\right]^\top$. The initial parameter estimates $\hat{\Theta}_s(0) = [0.6,\ 0.1,\ 0.1]^\top$ were set different from the measured nominal values. The control gains were constant throughout the experiments and were set to $\Gamma = 1.2$, $\lambda = 3.2$, $K_D = 0.8$.

5.5.1 Uni-directional bending

Here only the sensor strain signals from the skin retrofitted onto *compartment A* and the actuator signals for *compartment A* were used for degree of curvature estimation via the learned model, although both the compartments were allowed to be actuated. Two experiments were conducted, one with a low frequency target trajectory, and the other with a high frequency target trajectory. For the low frequency target, the desired degree of curvature was set to $q_d(t) = (\pi/8) - (\pi/9)\cos(\pi t/12)$. The results are shown in Figure 5.7 wherein the tracking RMSE was $4.35°$, and the estimation RMSE was $2.78°$. The results for the high frequency trajectory tracking

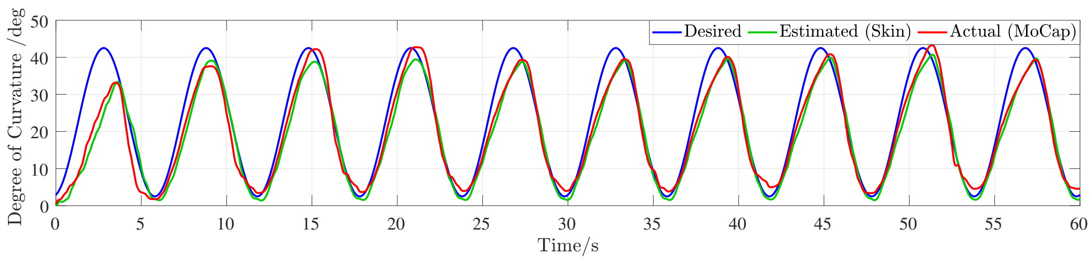


(a) Tracking performance

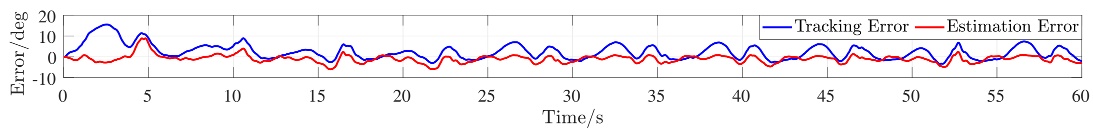


(b) Error plot

Figure 5.7: Uni-directional low frequency target trajectory tracking



(a) Tracking performance



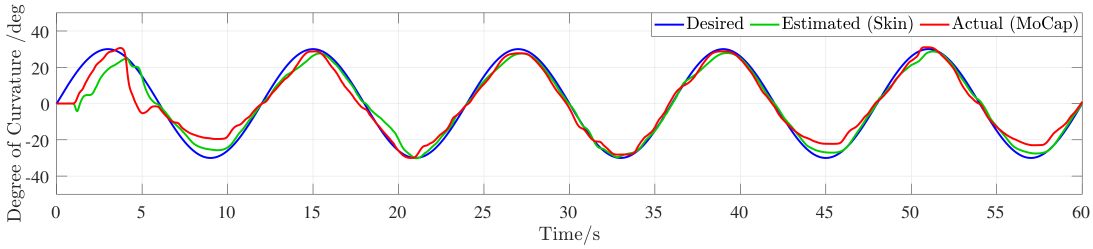
(b) Error plot

Figure 5.8: Uni-directional high frequency target trajectory tracking

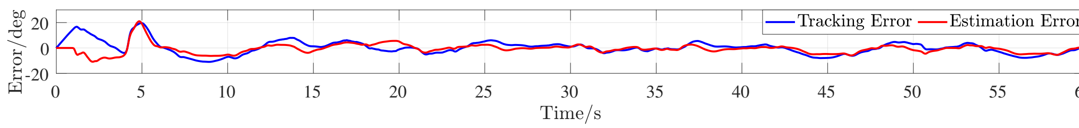
are shown in Figure 5.8, where $q_d(t) = (\pi/8) - (\pi/9)\cos(\pi t/3)$. In this case the tracking and estimation RMSE was found to be $4.09°$ and $2.27°$ respectively.

5.5.2 Bi-directional bending

Here both *compartments* were actuated, and strain signals from both the sensing skins were used for degree of curvature estimation via the learned model for bi-directional bending. Two experiments were conducted, one with a low frequency target trajectory and one with a high frequency target trajectory. For the low frequency target, the desired degree of curvature

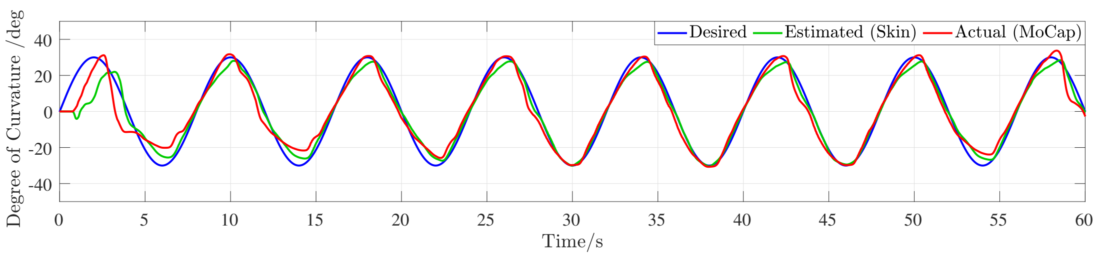


(a) Tracking performance

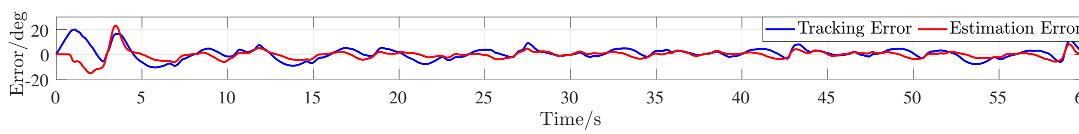


(b) Error plot

Figure 5.9: Bi-directional low frequency target trajectory tracking



(a) Tracking performance



(b) Error plot

Figure 5.10: Bi-directional high frequency target trajectory tracking

was set to $q_d(t) = (\pi/6)\sin(\pi t/6)$. The results are shown in Figure 5.9, in which the tracking and estimation RMSE was found to be $5.05°$ and $3.79°$ respectively. The results for the high frequency trajectory tracking are shown in Figure 5.10, where $q_d(t) = (\pi/6)\sin(\pi t/4)$. Here the tracking RMSE was $5.10°$ and the estimation RMSE was $3.73°$.

5.6 Discussion

The experimental results exhibit the successful utilization of the retrofitted soft sensing skin for the degree of curvature estimation for adaptive tracking control of a desired curvature trajectory. The uni-directional bending illustrates the use of a single soft sensing skin for degree of curvature estimation when the soft robot only bends in a certain direction. This capability is useful for sensing and estimation of soft segments, such as in wearable robots, that have only a single compartment and only bend in a one direction. The bi-directional experiments demonstrate the use of two sensing skins retrofitted on the two compartments of the soft robot for curvature estimation. The bi-directional bending is especially important in soft robot manipulation.

Considering both the uni-directional and bi-directional bending, the capability of the integrated sensing skins to estimate the curvature for slow as well as fast manipulations are shown, and the fast response of the sensors are reflected in the satisfactory tracking of the target trajectory. In the starting of the experiments the higher tracking error maybe due to uncertain parameters which in time gets better due to parameter adaptation.

5.7 Summary

This chapter demonstrated the use of integrated sensing for dynamic control of soft robots under the PCC modeling hypothesis. The soft sensing skins proposed in this work could be retrofitted to many soft robots, and the degree of curvature estimation can be learned using an LSTM network, only requiring the strain signals from the sensing skin and the actuator inputs. Moreover, an adaptive controller was designed to track a desired degree of curvature trajectory. The satisfactory degree of curvature tracking using the adaptive controller for low and high frequency target trajectories demonstrates that the proposed soft skins are capable of estimating the degree of curvature robustly for inclusion in a dynamic control framework.

Chapter 6: Passivity-Based Task Space Control of Hybrid Rigid-Soft (HyRiSo) Robots with Parametric Uncertainty

This chapter introduces a novel robotic system, coined as a hybrid rigid-soft (HyRiSo) robot composed of serially attached rigid and soft links for dexterous and precise motion. Due to the heterogeneous modes of actuation for the revolute joints and soft link bending, it is challenging to design an integrated controller for this class of robots. This chapter demonstrates that the well-known passivity-based adaptive and robust controllers can be utilized to address this challenge. Specifically, these controllers are used for task space tracking in the presence of uncertain stiffness, damping and actuation parameters in complex environments. Numerical examples are provided using a 2-rigid-2-soft HyRiSo robot illustrating the efficacy of these frameworks.

6.1 Overview

Despite the progress, it is challenging to achieve precise motion control of complex multi-link soft robots in a large workspace. It has been observed that increasing the length of the soft links induces workspace warping and makes the control problem harder [101]. While modern day rigid robots do not suffer from these challenges due to their precision, they lack the dexterity for completing task space operations, especially in cluttered environments [102]. On the other hand, the new class of robots discussed in this chapter, the integrated hybrid rigid-soft (HyRiSo) robots, integrates the dexterity of a soft manipulator with the payload capacity and precision of a rigid robot.

In a previous related study, the authors in [14] proposed a control framework to control a rigid arm with an extruding soft manipulator used for berry picking. However, the rigid links and the soft link were separately controlled in this work- kinematic inversion was utilized to control the rigid portion of the arm to reach a specific pose while a reinforcement learning policy was used for the soft link to reach the target position. While the performance was satisfactory, it was slow and required several user commands. An important observation for proposed class of hybrid robots is that heterogeneous actuation modalities are utilized for the revolute joints and the soft links. Thus, incorporating the uncertain actuator mapping from the pressure/current variables to the realized torques is important in the proposed control framework. Hence, developing a model-based integrated dynamic controller would be beneficial for fast autonomous operation as well as to understand the behavior of the HyRiSo robot and to provides guarantee for stability and performance. Furthermore, as there could be uncertainties arising in the dynamic parameters of the robot, stiffness and damping terms of the soft links as well as lack of knowledge of the actuator mapping, it would be desirable to design controllers that are robust to such uncertainties.

There have been several fundamental advancements to address the challenge of parametric uncertainty in classical rigid robotic control. Several adaptive control laws for rigid robots that adapts to the parameter uncertainty has been proposed over the years [61, 79, 103, 104]. Robust controllers that are robust to uncertainty as well as the unmodelled disturbances have also been proposed [80, 81]. In [105], the authors introduced an adaptive controller for handling uncertainty in dynamic, kinematic as well as actuator parameters. In [106], the authors proposed an adaptive control framework to compensate for actuator failure in cooperative robots with uncertain parameters. Recently, the soft robotics community has also made progress in trying to handle the uncertainty in the control design. Among the few works, curvature space adaptive control of soft robots using an augmented model has been proposed used in our previous work [94] as presented in Section 5.3. A recent study by the authors in [44] have used adaptive control for curvature tracking of a 3D soft robot. Task space adaptive control for soft robots was presented in Section 2.3 and was extended for a bilateral teleoperation framework in our previous work [73]. Several robust control approaches have also been proposed for soft robots [45]. Recently, in [46], a nonlinear adaptive position and stiffness controller for pneumatic soft robots was proposed.

However, an integrated control framework that addresses the model and actuation uncertainties in the class of HyRiSo robots is yet to be explored. To that end, in this chapter, first, the HyRiSo robot is proposed and then passivity-based adaptive and robust controllers for task space trajectory tracking are designed for this class of robots. Moreover, as the task space operation involve sub-tasks such as conforming to joint/curvature limits and collision avoidance with obstacles, the null space velocity is exploited to design sub-tasks to handle these issues and demonstrate the dexterity of the HyRiSo robot.

The main contributions of this chapter is to propose the integrated hybrid-rigid-soft (HyRiSo) robot for dexterous task space manipulation and to develop passivity-based adaptive and robust controllers for this new system in the presence of parametric uncertainty in system dynamics as well in the actuator mapping. Additionally, the efficacy of HyRiSo robot for task space manipulation is demonstrated by exploiting the system redundancy to enable tracking in the presence of complex obstacles, joint limits, and unmodeled disturbances. This work was presented partly in [107].

The rest of the chapter is organized as follows. Section 6.2 introduces the novel hybrid robot model. Next, Section 6.3 discusses the control design of the passivity-based adaptive controller and the robust controller. Finally, the numerical examples with simulation results are illustrated in Section 6.4.

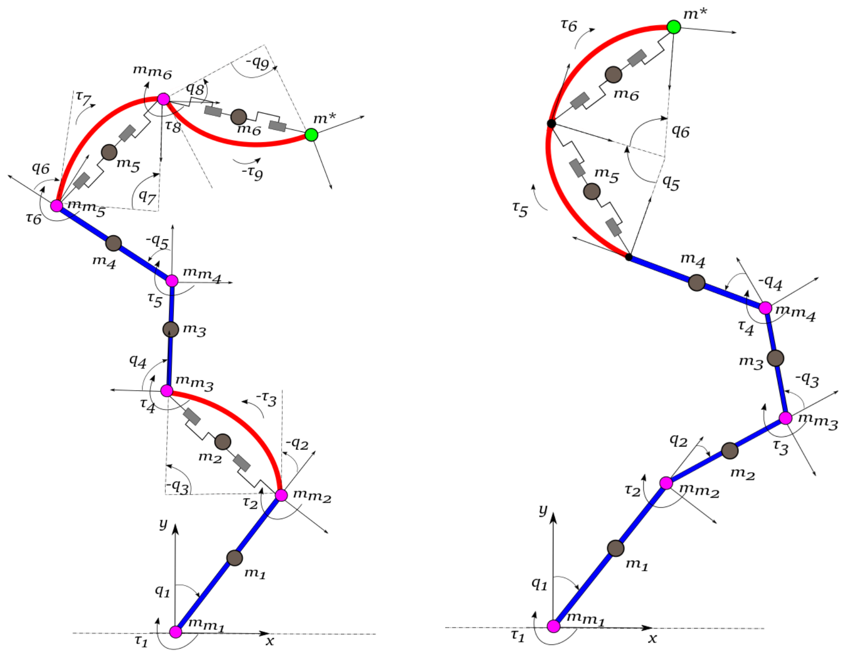
6.2 The HyRiSo robotic system

This section introduces the novel robotic system proposed in this chapter - a fully actuated planar HyRiSo robot which is composed of rigid links as well as soft links.

6.2.1 Dynamics

Let us consider a robotic system with n_r rigid links and n_s soft links serially connected together forming an open chain of $n_s + n_r$ links. In the general setting, all the joints, rigid & rigid, rigid & soft and soft & soft to be revolute joints are considered. The soft links are assumed to be non-extensible and have a constant curvature (CC) [16]. In general, the hybrid robotic system will have $\alpha = n_r + 2n_s$ configuration variables for the system, namely the revolute joint angles (for all $n_r + n_s$ links), and the degree of curvature of the n_s soft links. Those are denoted as q_i, with $i = 1, 2, ..., \alpha$ and collect them to the vector $q = [q_1, q_2, ..., q_\alpha]^\top \in \mathbb{R}^\alpha$. It should be noted that, in certain hybrid robot designs, one can also consider fixed joints between the rigid & soft or soft & soft links. Two representative HyRiSo robots are shown in Figure 6.1.

In this chapter it is assumed that the rigid links are uniform and the masses m_i, with $i \in \{\text{rigid link indices}\}$ are lumped at the centroid of the links. To accommodate motors at the



(a) RSRRSS HyRiSo robot (b) RRRRSS HyRiSo robot

Figure 6.1: Two examples of HyRiSo robots composed of rigid (R) links (blue) and CC soft (S) links (red) modelled using the augmented formulation [22] are illustrated. The lumped link masses m_i (brown), lumped motor masses m_{m_i} (pink) and the lumped end effector mass m^* (green) are also shown here. The RSRRSS design (a) considers a fully fledged HyRiSo robot with three rigid and three soft links with all actuated revolute joints. The RRRRSS design (b) considers a HyRiSo robot with four rigid and two soft links with fixed joints between rigid & soft and soft & soft links.

revolute joints, assume the motor masses m_{m_i}, with $i \in \{$revolute joint indices$\}$ to be lumped at the revolute joints. Also assume that the end effector mass m^* to be lumped at the hybrid robot's tip. Moreover, as the soft links are assumed to be CC, the *augmented formulation* [22] as described in Section 2.2 is used to model the soft links. Using this formulation, the moment of inertias of the soft links are neglected since rotational kinetic energy is much smaller than the translational energy components [108]. The soft links masses m_i, with $i \in \{$soft link indices$\}$ are at the center of the main chord connecting the ends of the soft link. The dynamics of

HyRiSo in the Lagrangian form is then written as,

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + D\dot{q} + Kq + G(q) = \tau \tag{6.1}$$

where $M(q) \in \mathbb{R}^{\alpha\times\alpha}$ is the inertia matrix, $C(q,\dot{q})\dot{q} \in \mathbb{R}^{\alpha}$ is the Coriolis and centrifugal terms, $G(q) \in \mathbb{R}^{\alpha}$ is the gravity vector and $\tau \in \mathbb{R}^{\alpha}$ is the control input vector. $D \in \mathbb{R}^{\alpha\times\alpha}$ and $K \in \mathbb{R}^{\alpha\times\alpha}$ are the damping and stiffness matrices respectively, which are assumed to be positive semi-definite diagonal matrices. Note that only the entries corresponding to the soft link will be non-zero in these damping and stiffness matrices. Here, as only the planar case is considered, the task space of the robot is considered to be $\mathbb{R}^2$, and hence HyRiSo is a redundant robot.

6.2.2 Actuator mapping via a transmission matrix

This section briefly discusses the implications of the actuator mapping via a transmission matrix. Assume a linear actuator mapping for the hybrid robot, $\tau = A\,p$, where $A \in \mathbb{R}^{\alpha\times\alpha}$ is the positive definite diagonal transmission matrix that maps the actuator signals $p \in \mathbb{R}^{\alpha}$ to the control torque τ. In HyRiSo robot, the actuator signals (p) that need to be sent to the actuation units could be, for example, the current signals for motor torques or the pressure signals for pneumatic actuation for bending. While calibrating and obtaining the exact parameters for motor actuation can be done easily, obtaining these parameters for soft links is cumbersome. In the soft links, the torques applied by the soft links could be order of magnitude less than the torques applied by the motors. This poses challenges in practical implementations due to incorrect calibration. Moreover, the actuator parameters could change over time (albeit very slowly) due to several factors such as changes in the temperature [105]. Therefore, incorporating the actuator parameters and accounting for their uncertainty in the control design is of paramount importance. In several prior work that considered uncertainty in the actuator parameters, for example in [105], the authors have used a separate parameter update law to update uncertain actuator parameters along with other parameter update laws for uncertain dynamics and kinematics.

This chapter notes a simple way to incorporate the actuator parameters as suggested in [46, 109]. Using the relationship of the actuator mapping for the torque on (6.1) we obtain,

$$A^{-1}\left(M\left(q\right)\ddot{q}+C\left(q,\dot{q}\right)\dot{q}+D\dot{q}+Kq+G\left(q\right)\right)=p,$$

and defining $M_0 = A^{-1}M$, $C_0 = A^{-1}C$, $D_0 = A^{-1}D$, $K_0 = A^{-1}K$ and $G_0 = A^{-1}G$ one can rewrite,

$$M_0(q)\ddot{q}+C_0(q,\dot{q})\dot{q}+D_0\dot{q}+K_0q+G_0(q)=p. \tag{6.2}$$

Note that, since M, C and G are linear in the dynamic parameters (see *Property A.3*), so is M_0, C_0 and G_0 as pre-multiplication by A^{-1} is a linear operation. Therefore, M_0, C_0 and G_0 are linear in suitably selected parameters which are nonlinearly related with the dynamic parameters and the actuator mapping parameters.

Therefore, the *Property A.3* can be extended to the system (6.2) such that for any differentiable vector $\gamma \in \mathbb{R}^{\alpha}$ there exists a regressor $Y_2(q,\dot{q},\gamma,\dot{\gamma}) \in \mathbb{R}^{\alpha\times\beta_2}$ and a constant parameter vector $\Theta_2 \in \mathbb{R}^{\beta_2}$ such that

$$M_0(q)\dot{\gamma} + C_0(q,\dot{q})\gamma + D_0(q)\gamma + K_0 q + G_0(q) = Y_2(q,\dot{q},\gamma,\dot{\gamma})\Theta_2.$$

Example 6.1 *Consider a simple planar HyRiSo robot system with one rigid link and one soft link with the actuator mapping $A = diag(a_1, a_2)$ operating on the horizontal plane. The elements of the inertia matrix $M(q) \in \mathbb{R}^2$ of this robot can be written as, $M_{11} = m_1\phi_1(L,q) + m_2\phi_2(L,q) + I_1$, $M_{12} = M_{21} = m_2\phi_3(L,q)$ and $M_{22} = m_2\phi_4(L,q)$, where $\phi_i(L,q)$'s are the regressor terms, which are nonlinear functions of the link lengths $L = [L_1, L_2]^\top$ and the configuration variables $q = [q_1, q_2]^\top$. These are defined explicitly in the Appendix A.5. For simplicity let us consider only the masses of the two links m_1, m_2 and the moment of inertia of the rigid link I_1 as inertial parameters. Since the Christoffel symbols are functions of elements of the inertia matrix, no additional parameters are required for the C matrix. Considering the stiffness K and damping D matrices, let the stiffness of the soft link k and the damping of the soft link d be considered as parameters (as noted in the Section 6.2.1, the stiffness and damping of the rigid link joint is null). Also, gravity terms are not considered as only the planar horizontal case is studied. Now, with the dynamics represented as in (6.1) the parameter vector becomes $\Theta = [m_1, I_1, m_2, k, d]^\top$ with the corresponding regressor Y. On pre-multiplying the dynamics with the inverse of the transmission matrix to obtain the dynamics in the form (6.2), observe that the elements in M_0, $M_{0_{11}} = (m_1/a_1)\phi_1(L,q) + (m_2/a_1)\phi_2(L,q) + (I_1/a_1)$, $M_{0_{12}} = (m_2/a_1)\phi_3(L,q)$,*

$M_{0_{21}} = (m_2/a_2)\phi_3(L,q)$ *and* $M_{0_{22}} = (m_2/a_2)\phi_4(L,q)$. *Consequently, inspecting* C_0 *as well, the inertial parameters in this case become* $\frac{m_1}{a_1}$, $\frac{I_1}{a_1}$, $\frac{m_2}{a_1}$ *and* $\frac{m_2}{a_2}$. *The parameters related to the stiffness and damping matrices are given as* $\frac{k}{a_2}$ *and* $\frac{d}{a_2}$. *Hence the new parameter vector is* $\Theta_2 = [\frac{m_1}{a_1}, \frac{I_1}{a_1}, \frac{m_2}{a_1}, \frac{m_2}{a_2} \frac{k}{a_2}, \frac{d}{a_2}]^\top$ *with the corresponding regressor* Y_2.

6.3 Passivity-based control for HyRiSo robots

This section presents the design of two passivity-based controllers, an adaptive controller and a robust controller, for task space trajectory tracking for HyRiSo robot. The proposed controllers are developed based on the Lagrangian formulation of the HyRiSo robots as given by (6.1), and follow similar ideas as provided in [61, 80]. However, the treatment for task space trajectory tracking closely relates to the methods implemented in Section 2.3.

Consider the direct forward kinematics $h(\cdot) : \mathbb{R}^\alpha \to \mathbb{R}^2$ which maps the configuration space to the task space. Thus, the end effector position and the velocity are defined as, $X(t) = h(q)$ and $\dot{X}(t) = J(q)\dot{q}$, where, $J(q) = \frac{\partial h(q)}{\partial q} \in \mathbb{R}^{2\times\alpha}$ is the Jacobian matrix. Let $X_d(t)$ be the desired reference task space trajectory. Then, the tracking error is defined as,

$$e(t) = X(t) - X_d(t).$$

In order to drive the tracking error to approach origin, the system trajectories are restricted to the sliding surface,

$$s(t) = J^+(q)\dot{e}(t) + J^+(q)\Lambda e(t) - \left(\mathbb{I}_\alpha - J^+(q)J(q)\right)\psi_s(t) \tag{6.3}$$

where, $\Lambda \in \mathbb{R}^{2\times 2}$ is a positive definite gain matrix. $\psi_s(t) \in \mathbb{R}^{\alpha}$ is the negative gradient of an appropriately defined convex function which is utilized for the sub-task control (see Appendix A.4). Here, $J^{+} \triangleq J^{\top}(JJ^{\top})^{-1} \in \mathbb{R}^{\alpha\times 2}$ is the pseudo inverse of J and satisfies the property $JJ^{+} = \mathbb{I}_{\alpha}$ where $\mathbb{I}_{\alpha}$ is the $\alpha \times \alpha$ identity matrix. It is seen that as once the trajectories reach $s = 0$,

$$\dot{e}(t) = -JJ^{+}\Lambda e(t) + J(\mathbb{I}_{\alpha} - J^{+}J)\psi_s = -\Lambda e(t) \tag{6.4}$$

and hence, the errors will reach the origin when $s = 0$.

Let us define signals, $v = \dot{q} - s$ and $a = \ddot{q} - \dot{s}$. Assuming uncertainty in the parameters, let us use the notation $\hat{(\cdot)}$ to denote the estimated values and $\tilde{(\cdot)}$ to denote the estimation error. Using the extended *Property A.3* of Lagrangian systems for the system (6.2), the regressor $(Y_2\,(q, \dot{q}, v, a))$ and parameter (Θ_2) vector pair for the estimated systems is defined as,

$$Y_2(q, \dot{q}, v, a)\hat{\Theta}_2 = \hat{M}_0(q)a + \hat{C}_0(q, \dot{q})v + \hat{D}_0 v + \hat{K}_0 q + \hat{G}_0(q). \tag{6.5}$$

Here time invariant uncertainties in the dynamic terms, stiffness, damping and actuator parameters are assumed. Thus, the parameter vector Θ_2 is a constant.

6.3.1 Passivity-based adaptive control

Following the adaptive control approach [61], the control input for the HyRiSo robot is defined as,

$$p = Y_2(q, \dot{q}, v, a)\hat{\Theta}_2 - K_s s \tag{6.6}$$

where K_s is a positive definite diagonal gain matrix.

The dynamics of the closed loop system can be found by substituting the proposed control (6.6) in the HyRiSo robot dynamics (6.1), and using (6.5),

$$M_0(q)\dot{s} + C_0(q, \dot{q})s + D_0 s + K_s s = Y_2(q, \dot{q}, v, a)\tilde{\Theta}_2 \tag{6.7}$$

where $\tilde{\Theta}_2 = \hat{\Theta}_2 - \Theta_2$. Let the adaptation law for the parameter estimation defined as,

$$\dot{\hat{\Theta}}_2 = -\Gamma Y_2^\top s \tag{6.8}$$

where Γ is a positive definite symmetric gain matrix that needs to be tuned.

Theorem 6.1 *Consider the closed loop system (6.7) with the parameter adaptation law (6.8) and sliding surface (6.3). In the absence of any external wrenches, the task space position error* (e) *and velocity error* $(\dot{e})$ *asymptotically reach the origin while the parameter estimation error* $(\tilde{\Theta}_2)$ *remains bounded.*

Proof of Theorem 6.1 *Consider a Lyapunov like function for the system defined as,*

$$V = \frac{1}{2}\left(s^\top M_0 s + \tilde{\Theta}_2^\top \Gamma^{-1} \tilde{\Theta}_2\right) \geq 0$$

Differentiating V with respect to time yields,

$$\begin{aligned}\dot{V} =& \frac{1}{2} s^\top \dot{M}_0 s + s^\top M_0 \dot{s} + \tilde{\Theta}_2^\top \Gamma^{-1} \dot{\tilde{\Theta}}_2 \\ =& \frac{1}{2} s^\top \dot{M}_0 s + s^\top \left(-C_0 s - D_0 s - K_s s + Y_2 \tilde{\Theta}_2\right) + \dot{\tilde{\Theta}}_2^\top \Gamma^{-1} \tilde{\Theta}_2\end{aligned}$$

Using the skew symmetry property (Property A.2) and using the chosen adaptation law (6.8) we obtain,

$$\dot{V} = -s^\top D_0 s - s^\top K_s s \leq 0.$$

As $V \geq 0$ and $\dot{V} \leq 0$, $\lim_{t\to\infty} V$ is finite. Therefore, $s \in \mathcal{L}_2$ and $s, \tilde{\Theta} \in \mathcal{L}_\infty$. From (6.7), noting the properties of Lagrangian systems, observe that $\dot{s} \in \mathcal{L}_\infty$. Therefore, since $s \in \mathcal{L}_2$ and $\dot{s} \in \mathcal{L}_\infty$, it can be shown that $s \to 0$ as $t \to 0$. Now from (6.4), $e, \dot{e} \to 0$ once $s = 0$.

6.3.2 Passivity-based robust control

Following the robust control approach in [80], and based off the adaptive control design, the control input as (6.6) is used with the parameter estimation vector $\hat{\Theta}_2$ now chosen as,

$$\hat{\Theta}_2 = \Theta_0 + u \tag{6.9}$$

where Θ_0 is a fixed nominal parameter vector and u is an additional control term which will be designed for achieving robustness for uncertain parameters. Hence, now no adaptation for the estimation of parameters are done.

Thus, using (6.9) in the control input (6.6) and substituting it in the HyRiSo robot dynamics (6.1) yields,

$$M_0(q)\dot{s} + C_0(q,\dot{q})s + D_0 s + K_s s = Y_2(q,\dot{q},v,a)(\tilde{\Theta}_0 + u) \tag{6.10}$$

where $\tilde{\Theta}_0 = \Theta_0 - \Theta_2$ is the parameter uncertainty which is constant. Suppose the uncertainty is bounded such that a constant bound $\rho \geq 0$ can be found satisfying,

$$||\tilde{\Theta}_0|| = ||\Theta_0 - \Theta_2|| \leq \rho. \tag{6.11}$$

Then, letting $\epsilon > 0$, the control term u is designed as,

$$u = \begin{cases} -\rho \frac{Y_2^\top s}{||Y_2^\top s||} & \text{if} \qquad ||Y_2^\top s|| > \epsilon \\ -\frac{\rho}{\epsilon} Y_2^\top s & \text{if} \qquad ||Y_2^\top s|| \leq \epsilon \end{cases} \tag{6.12}$$

Theorem 6.2 *Consider the closed loop system (6.10) with bounded parameter uncertainty as (6.11), the additional control u defined as (6.12) and the sliding surface (6.3). Then, the tracking error is uniformly ultimately bounded (u.u.b)).*

Proof of Theorem 6.2 *Consider a Lyapunov like function for the system defined as,*

$$V = \frac{1}{2} s^\top M_0 s$$

Differentiating V with respect to time yields,

$$\begin{aligned}
\dot{V} =& \frac{1}{2} s^\top \dot{M}_0 s + s^\top M_0 \dot{s} \\
=& \frac{1}{2} s^\top \dot{M}_0 s + s^\top \left(-C_0 s - D_0 s - K_s s + Y_2(\tilde{\Theta}_0 + u) \right) \\
=& -s^\top Q s + s^\top Y_2(\tilde{\Theta}_0 + u)
\end{aligned}$$

where the skew symmetric property is utilized and $Q := D_0 + K_s$ which is a positive definite diagonal matrix. Considering the term $s^\top Y_2(\tilde{\Theta}_0 + u)$, observe that if $||Y_2^\top s|| > \epsilon$ then,

$$\begin{aligned}
s^\top Y_2(\tilde{\Theta}_0 + u) =& (Y_2^\top s)^\top \left(\tilde{\Theta}_0 - \rho \frac{Y_2^\top s}{||Y_2^\top s||} \right) \\
\leq& ||Y_2^\top s|| \left(||\tilde{\Theta}_0|| - \rho \right) < 0.
\end{aligned}$$

which implies that $\dot{V} < 0$ with respect to s. Note that $||\tilde{\Theta}_0|| \leq \rho$ and $\rho \geq 0$. Hence, $\tilde{\Theta}_0 \leq \rho \frac{Y_2^\top s}{||Y_2^\top s||}$. Now, if $||Y_2^\top s|| \leq \epsilon$ then,

$$\begin{aligned}
s^\top Y_2(\tilde{\Theta}_0 + u) =& (Y_2^\top s)^\top (\tilde{\Theta}_0 + u) \\
\leq& (Y_2^\top s)^\top \left(\rho \frac{Y_2^\top s}{||Y_2^\top s||} + u \right) \\
=& \rho ||Y_2^\top s|| - \frac{\rho}{\epsilon} ||Y_2^\top s||^2.
\end{aligned}$$

The maximum of the R.H.S in the above expression is $\epsilon\rho/4$ which is achieved when $||Y_2^\top s|| = \epsilon/2$. Therefore,

$$\dot{V} \leq -s^\top Qs + \epsilon\rho/4$$

and see that $\dot{V} < 0$ if $s^\top Qs > \epsilon\rho/4$. Using the bounds on the quadratic form, $\lambda_{min}(Q)||s||^2 \leq s^\top Qs \leq \lambda_{max}(Q)||s||^2$ where $\lambda_{min}(Q)$ and $\lambda_{max}(Q)$ are, respectively, the minimum and maximum eigenvalues of the matrix Q, we have that $\dot{V} < 0$ if $\lambda_{min}(Q)||s||^2 > \epsilon\rho/4$ or, equivalently

$$||s|| > \left(\frac{\epsilon\rho}{4\lambda_{min}(Q)}\right) =: \delta.$$

The u.u.b follows from this result using δ to define the radius of the ultimate boundedness set.

6.4 Numerical simulations

This section presents numerical simulation results illustrating the efficacy of the proposed passivity-based controllers for the proposed HyRiSo robots. First, the HyRiSo robot parameters are introduced and the simulation results are discussed subsequently.

6.4.1 The HyRiSo robot parameters

A four DoF HyRiSo robot is employed on the horizontal plane. The robot is composed of two rigid links and two soft links where the first two links are rigid and the last two links are soft. The joints between rigid & soft and soft & soft links are considered to be fixed. The lengths of the manipulators are $L = [1, 1, 1, 1]^\top$m. The masses of the links are $m = [0.3, 0.3, 0.51, 0.51]^\top$kg. Ignoring the base motor as it does not contribute to the dynamics, the motor mass at the rigid-rigid joint is $m_m = 0.1$kg. The end effector mass is $m^* = 0.25$kg. The moment of inertia of the rigid links is given as $I = [0.025, 0.025]^\top \mathrm{kg\,m}^2$. The stiffness of the two links $k = 0.03$ N/m, and the damping of the two links $d = 0.05$ Ns/m was assumed to be the same. The base of the robot is attached to the environment at the origin. Considering the actuator mapping, it is assumed that the two motor signals for the two revolute joints have the same scaling and the two bending signals for the two soft links have the same scaling, but different from that of the motor signals. Hence, define $A = \mathrm{diag}(a_1, a_1, a_2, a_2)$ with $a_1 = 2$ and $a_2 = 0.1$.

The parameter vector was chosen as $\Theta_2 = [\frac{m_1}{a_1}, \frac{I_1}{a_1}, \frac{m_m}{a_1}, \frac{m_2}{a_1}, \frac{I_2}{a_1}, \frac{m_3}{a_1}, \frac{m_3}{a_2}, \frac{m_4}{a_1}, \frac{m_4}{a_2}, \frac{m^*}{a_1}, \frac{m^*}{a_2}, \frac{d}{a_2}, \frac{k}{a_2}]^\top$ with the nominal values chosen as $\hat{\Theta}_2(0) = 0.05\,\mathbf{1}^{13\times1} = \Theta_0$. Here $\mathbf{1}^{13\times1}$ is a 13×1 vector containing 1 as all entries. With this selection the uncertainty bound can be found as $||\tilde{\Theta}_0|| \leq 7.576$, and $\rho = 8$ is used in the robust controller along with $\epsilon = 0.1$. For the adaptive controller the adaptation gain $\Gamma = 2.75$ and the control gain $K_s = \mathbb{I}$ were chosen for all the scenarios. The tracking gain $\Lambda = 18\,\mathbb{I}_2$ was used in the adaptive controller, and $\Lambda = 12\,\mathbb{I}_2$ in the robust controller. The initial pose of the HyRiSo robot was $q(0) = [-\frac{\pi}{16}, -\frac{\pi}{6}, \frac{\pi}{5}, \frac{\pi}{3}]^\top$ with zero initial velocities for all the simulations.

6.4.2 Simulation results

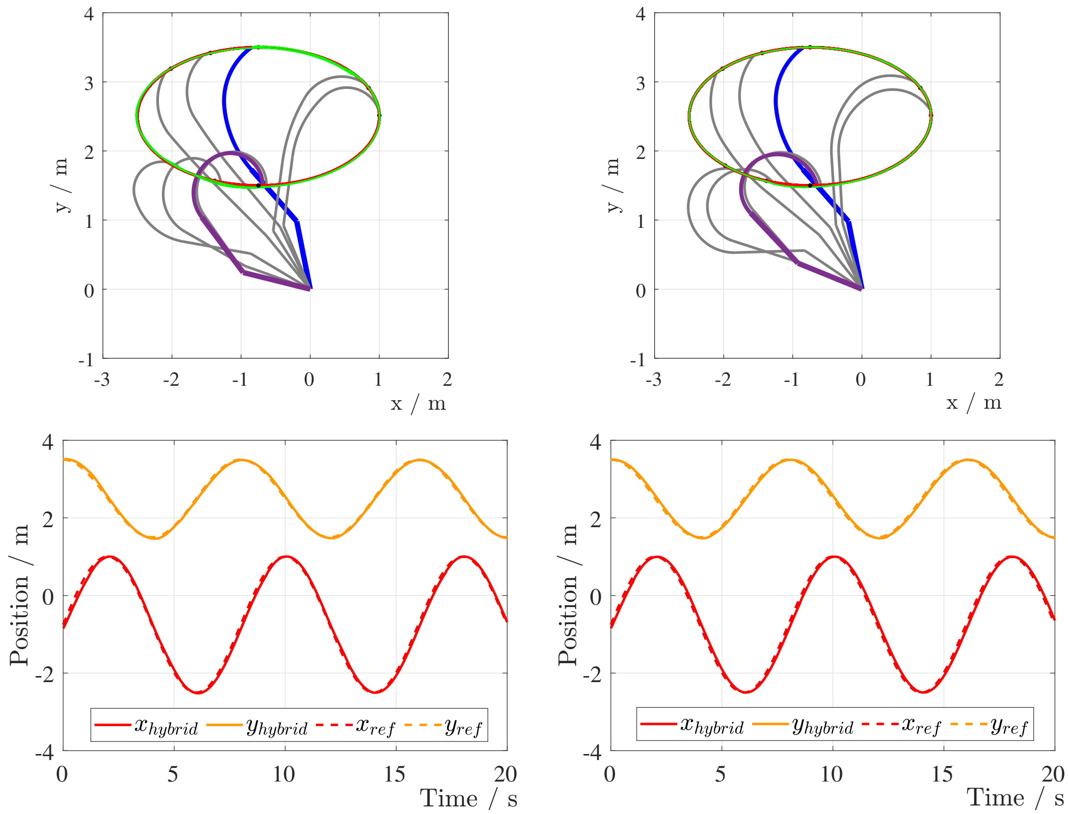
Two scenarios are considered - the first scenario is task space trajectory tracking of an oblique circle shape in an uncluttered environment and the second scenario is task space trajectory tracking of line segments in a cluttered environment. In both these scenarios the adaptive controller and the robust controller are tested under different conditions that will be discussed subsequently in the scenario descriptions.

6.4.2.1 Trajectory tracking of an oblique circle shape in uncluttered environment

In this scenario the reference trajectory $X_d(t) = [x_{ref}(t), y_{ref}(t)]^\top$ is defined as,

$$\begin{aligned} x_{ref}(t) = & -0.75 + 1.75\sin(0.25\pi t) \\ y_{ref}(t) = & \; 2.5 + \cos(0.25\pi t) \end{aligned}$$

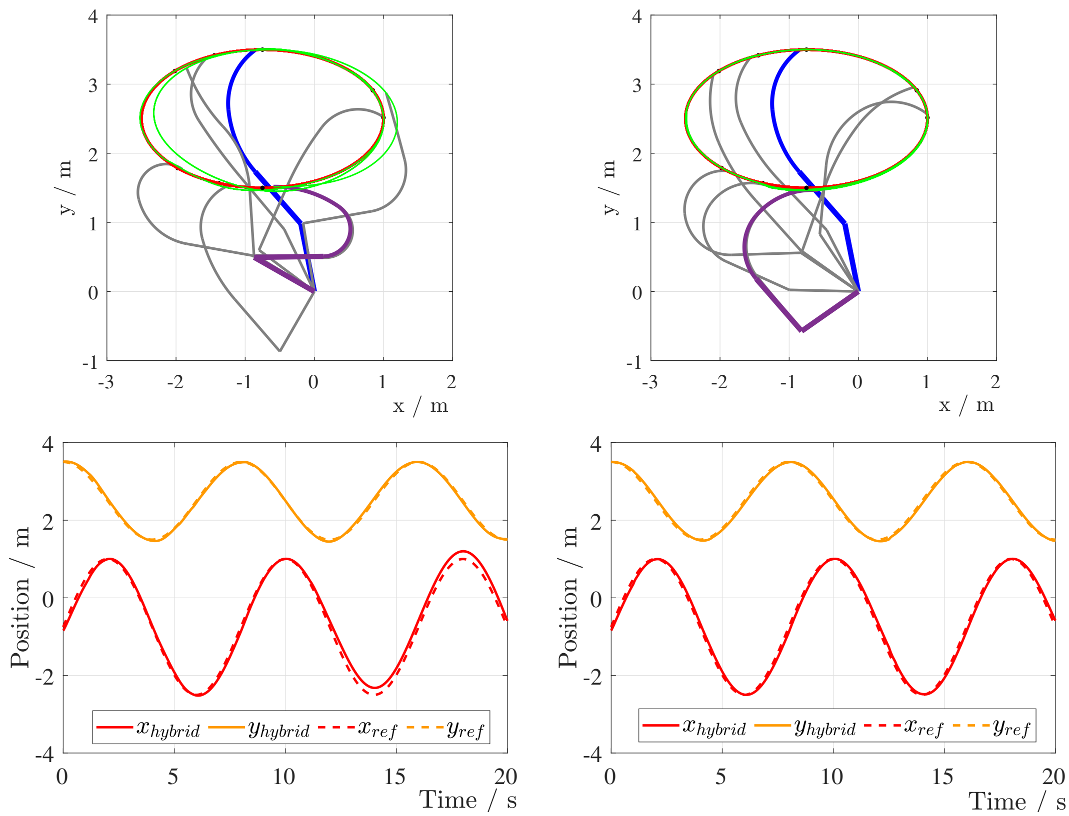
Two simulated experiments were considered, one in which no external disturbance was involved and another with a constant disturbance of $F = [-2, 1]^\top N$ on the robot tip appearing at $t = 8s$ and remaining till the end of the simulation. The simulation results for these experiments are illustrated in Figure 6.2 and Figure 6.3. The tracking errors are shown in Figure 6.4. Observe that the two controllers have comparable performance in the experiment without any external disturbance. However, as soon as the external disturbance is introduced, the adaptive controller performs poorly while the robust controller performs similar to the experiment without any disturbances.



(a) Adaptive controller

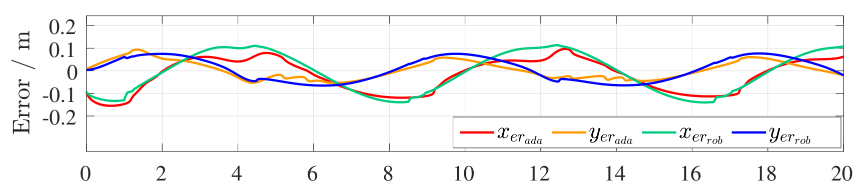
(b) Robust controller

Figure 6.2: Performance of the HyRiSo robot utilizing the designed controllers for trajectory tracking in an uncluttered environment in the absence of disturbance. The configuration plots on the top illustrates the initial pose in blue, end pose in purple and intermediate poses in gray. The reference trajectory is plotted in red and the actual tip trajectory is overlaid in green.

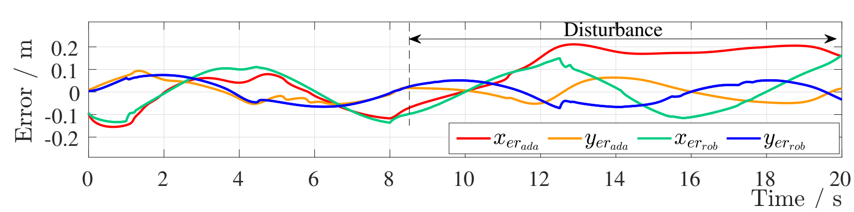


(a) Adaptive controller (b) Robust controller

Figure 6.3: Performance of the HyRiSo robot utilizing the designed controllers for the trajectory tracking in an uncluttered environment. In this simulation a disturbance force of $F = [-2, 1]^\top N$ was applied to the tip of the robot at $t = 8s$ and continued to be applied till the end. The configuration plots on the top illustrates the initial pose in blue, end pose in purple and intermediate poses in gray. The reference trajectory is plotted in red and the actual tip trajectory is overlaid in green.



(a) Errors is the absence of any disturbance



(b) Errors with a constant disturbance after $t = 8s$

Figure 6.4: Trajectory tracking errors for tracking an oblique circle.

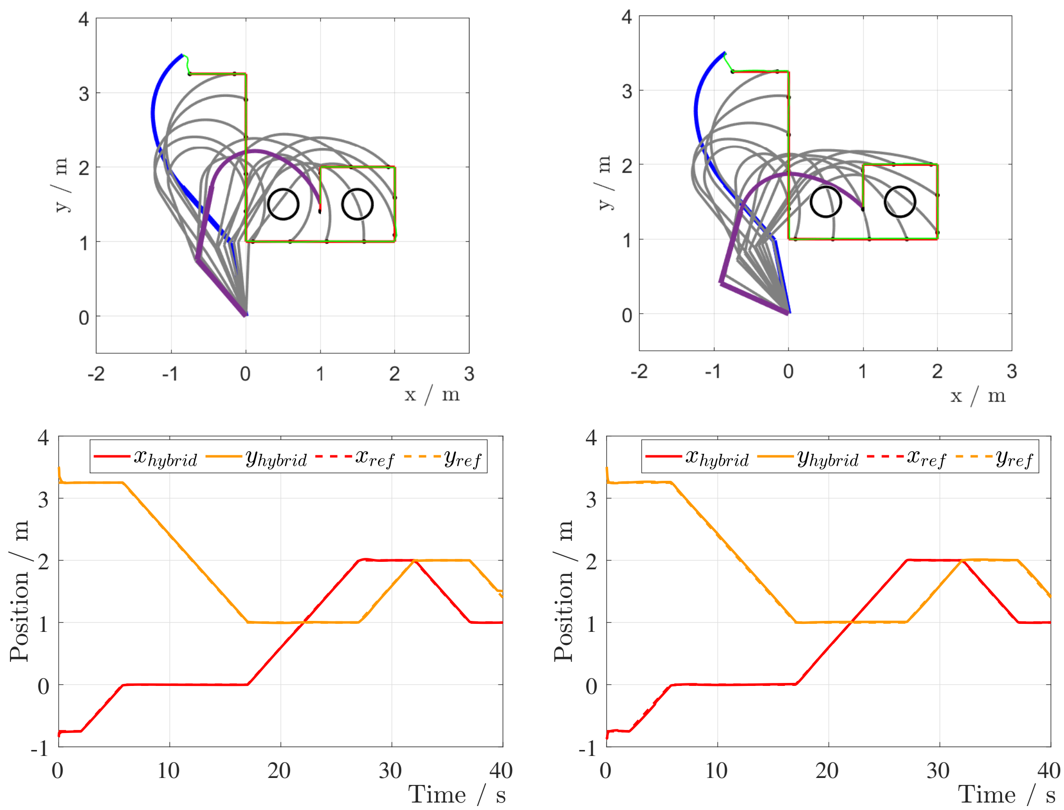
Table 6.1: Reference trajectory for collision avoidance scenario

Time / s	Reference trajectory ($X_d(t)$) / m
$0 \leq t < 2$	$[-0.75, \quad 3.25]^\top$
$2 \leq t < 5.75$	$[-0.75 + 0.2(t-2), \quad 3.25]^\top$
$5.75 \leq t < 17$	$[0, \quad 3.25 - 0.2(t-5.75)]^\top$
$17 \leq t < 27$	$[0.2(t-17), \quad 1]^\top$
$27 \leq t < 32$	$[2, \quad 1 + 0.2(t-27)]^\top$
$32 \leq t < 37$	$[2 - 0.2(t-32), \quad 2]^\top$
$37 \leq t \leq 40$	$[1, \quad 2 - 0.2(t-37)]^\top$

6.4.2.2 Trajectory tracking of line segments in a cluttered environment

In this scenario, the environment has two obstacles placed at $X_1 = [0.5,\ 1.5]^\top$ and $X_2 = [1.5,\ 1.5]^\top$. The smallest safe distance for both the obstacles was set to $r = 0.3m$ and the avoidance distance was set to $R = 0.7m$. The base of the first and second soft links and the mid point of the second (last) soft link were selected as the points on the robot to avoid obstacles. The reference trajectory in this scenario was defined as in Table 6.1.

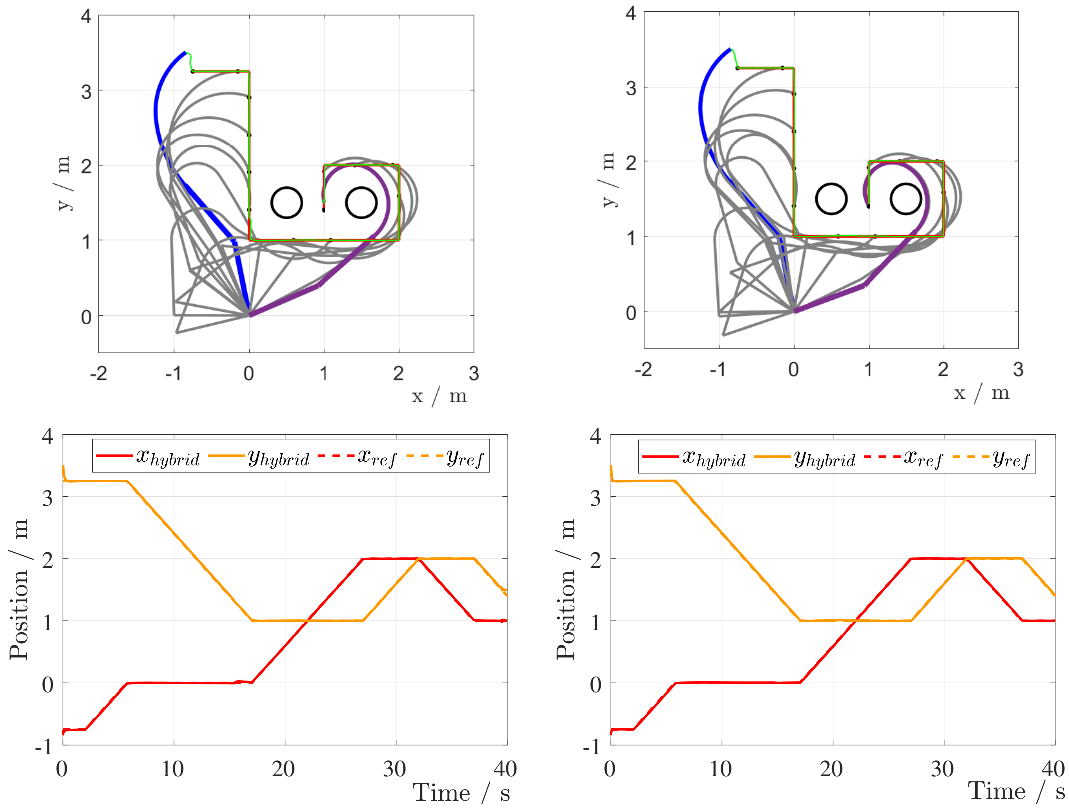
Figure 6.5 illustrates the performance of the HyRiSo robot utilizing the designed controllers without activating the subtask control of collision avoidance. It is clearly seen that the robot collides with the obstacles in this case. However, one can assume that this simulation illustrates the trajectory tracking of line segments if the environment was uncluttered. Figure 6.6 illustrates the efficacy of the sub task control which avoids any collisions with the obstacles while achieving trajectory tracking. This also showcases the dexterity of the HyRiSo robot. The tracking errors for these simulations are illustrated in Figure 6.7.



(a) Adaptive controller

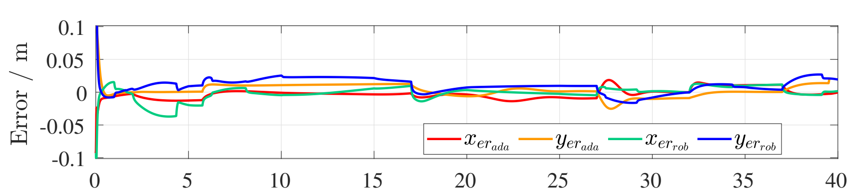
(b) Robust controller

Figure 6.5: Performance of the HyRiSo robot utilizing the passivity-based controllers for trajectory tracking in the cluttered environment without the use of the sub-task for collision avoidance. The configuration plots on the top illustrates the initial pose in blue, end pose in purple and intermediate poses in gray. The reference trajectory is plotted in red and the actual tip trajectory is overlaid in green. The obstacles are shown by black circles.

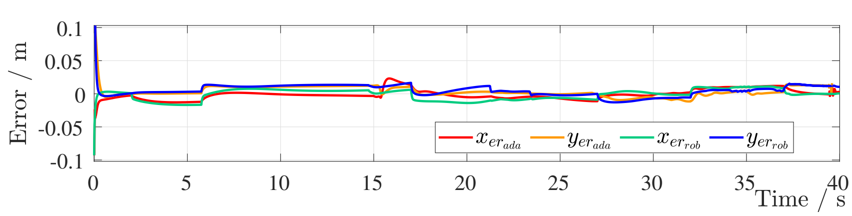


(a) Adaptive controller (b) Robust controller

Figure 6.6: Performance of the HyRiSo robot utilizing the passivity-based controllers for trajectory tracking amongst obstacles in the environment with the use of the sub-task control for collision avoidance. The configuration plots on the top illustrates the initial pose in blue, end pose in purple and intermediate poses in gray. The reference trajectory is plotted in red and the actual tip trajectory is overlaid in green. The obstacles are shown by black circles.



(a) Errors without using the sub-task control.



(b) Errors with the sub-task control of collision avoidance

Figure 6.7: Trajectory tracking errors for the simulated experiments in the cluttered environment.

6.5 Summary

In this chapter, a novel robotic system coined as a hybrid rigid-soft (HyRiSo) robot, was introduced. This robot is composed of rigid links and soft links serially attached together. This class of robots are particularly interesting as they possess enhanced dexterity properties thanks to the additional soft links in the system. The hybrid nature of the robot introduces uncertainties in the parameters. An important note is that the heterogeneity in the mode of actuation for the revolute joints and the soft links. Hence, the uncertain actuator mapping plays an important role in controlling the class of hybrid robots. To that end, two passivity-based controllers, an adaptive controller and a robust controller were developed in this work to remedy this challenge. The efficacy of the proposed hybrid robotic system with the designed controllers was illustrated using numerical examples showcasing task space trajectory tracking in challenging workspace environments.

Chapter 7: Swing up Control of a Soft Inverted Pendulum

This chapter introduces a novel soft robotic system, which is a soft inverted pendulum with a revolute joint at the base. This is an underactuated system since the revolute joint is not actuated. The soft body is hypothesized to be of constant curvature and it is actuated. Motivated by swing up controllers for classical underactuated systems, a switching based swing up and stabilization control of the proposed soft robot system is studied. The simulation results are depicted to illustrate the effectiveness of the proposed control approach for the soft inverted pendulum system.

7.1 Overview

Recently, research efforts have largely focused on developing modeling and control frameworks for soft robots [5, 22]. However, the main challenge has been to robustly control soft systems taking into account the highly underactuated nature of the soft robots. The study of control frameworks for underactuated soft robots can benefit from the corresponding development in underactuated rigid robots, which in the past four decades led to extensive advances, impacting industrial manipulators, wheeled and flying vehicles, and locomotion systems, among others.

As a first step toward understanding how the softness impacts the control performance of

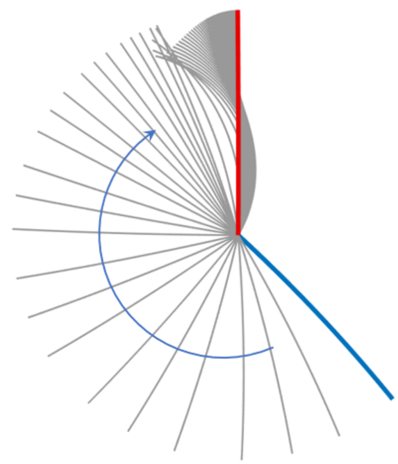


Figure 7.1: Illustration of the swing up control of SIPR. Here the initial position is shown in blue and the final position in red. The intermediate positions are shown in gray.

an underactuated soft inverted pendulum system, in this chapter, a novel soft system inspired by the classical inverted pendulum - soft inverted pendulum with revolute base (SIPR) is introduced. Essentially, this is a non-extensible soft bodied pendulum which is equipped with a revolute joint at the base. The soft body's curvature, which is hypothesized to be constant along its length, is actuated. This simplifying assumption made in this study allows the soft robot's curvature space to be fully actuated. However, the revolute joint is not actuated. Thus, the angular acceleration of the soft pendulum cannot be controlled directly making the SIPR system an underactuated mechanical system. It should be noted that the system considered in this chapter is different from the recently introduced soft inverted pendulum with affine curvature [35], wherein the base was considered to be fixed. In [110], the authors considered a soft appendage mounted on a rotating base with a torque input at the base with no actuation to the soft body. These are fundamentally different from the system considered in this chapter as this chapter considers a soft inverted pendulum with an unactuated base and an actuated soft body.

The dynamics of the SIPR system are derived using the recent results of Della Santina et al., in [34, 35]. This system can be viewed as an extension of the prevailing rigid robot template models (e.g. acrobot and pendubot) for testing nonlinear control strategies. Although the structure of the dynamics of the SIPR system is similar to that of the classical acrobot [111], it is more complex and highly nonlinear.

In view of the classical control problem for rigid underactuated robots, this chapter investigates the swing up control of the SIPR and the stabilization of the SIPR in its vertical upright position, as illustrated in Figure 7.1, initializing from below the horizontal level. To the best of our knowledge, the system considered here is novel, and is a first provably correct controller development for the swing up problem for any soft robot. The results discussed in this chapter was presented in [112].

In general, control of nonlinear underactuated systems, has been a challenging problem [113, 114]. This is more so, as the control algorithms developed for fully actuated robots cannot be directly utilized to control underactuated mechanical systems [113, 115]. In the literature, the researchers have considered specific underactuated systems such as the pendubot [116], acrobot [111], cart-and-pole system [117] and bipedal robots [118], and have developed controllers case-by-case for these systems. While all these are rigid systems, recently, an underactuated soft inverted pendulum with affine curvature has been analysed in [35], and an energy shaping control of a soft robot with in-plane disturbances has been developed in [119].

Swing up control for the class of acrobots and gymnist robots using partial feedback linearization was proposed in [111, 120, 121]. In this method the inherent nonlinearities are canceled before the control design. This requires the exact knowledge of the system parameters. Also, such a linearization step might cancel out desirable nonlinear properties of the system as well. This is more concerning in the case of soft robotics because the primary idea of using the soft robots is to make use of the natural nonlinear properties of these type of robots [122]. Lozano et al., have proposed control algorithms for the same class of robots considering the total energy of the system [117]. As this energy-shaping approaches does not require high gains, it can potentially preserve the compliance of soft systems in the closed loop system.

Other potential methods to control such under-actuated systems include interconnection and damping assignment based control [123, 124]. Recently, an energy shaping method circumventing the solution of partial differential equations was introduced in in [125]. However, the underlying assumptions in their work, precluded the utilization for considered SIPR system.

The main contribution of this chapter is in the swing up and upright stabilization for the introduced novel SIPR system. The swing up control algorithm proposed here closely follows the methodology outlined in [116] which was developed to control the well known Pendubot [126]. An energy-based approach was used to develop the proposed control design. For stabilizing the soft pendulum in the vertical upright position, the control is switch to a linear quadratic regulator (LQR) when the swing up control guides the system to the region of attraction of the desired equilibrium. The simulation results are also presented illustrating the efficacy of the proposed control method for the novel SIPR system.

The rest of the chapter is organized as follows. Section 7.2 introduces the soft robot model. The swing up control is developed in Section 7.3. Then, the simulation results are presented in Section 7.4.

7.2 The soft robotic system

In this section, first, the kinematics of the of the considered SIPR system is discussed. Then, the dynamics model of the system is introduced and its properties such as the equilibrium points and the total energy of the system are discussed.

7.2.1 Kinematics

Let us consider the SIPR system, a planar non-extensible soft robot with CC [16] that has a revolute joint at its base as shown in Figure 7.2. Let the length of the soft robot along the central axis be L and thickness D. Following the characterization in [35], the positions along the central axis of the soft robot is parameterized by $s \in [0, 1]$ such that Ls is the arc length along the robot's central axis to the point s from the base. The lateral points at each position is parameterized by $d \in [-0.5, 0.5]$ such that Dd is the lateral distance to the point d from the central axis. At each point s along the soft robot's body, reference frames $\{S_s\}$ are attached where as the base frame $\{\bar{S}_0\}$ is fixed in space. These frames can be used to describe any point on the soft robot along with the parameters (s, d).

Using the CC hypothesis, the configuration variables of this system are selcted as the degree of curvature of the soft robot $q_0(t) \in \mathbb{R}^1$ and the base rotation of the soft robot $\theta(t) \in \mathbb{S}^1$. Note that the degree of curvature $q_0(t)$ and the radius of curvature $r(t)$ has the relationship $q_0(t)r(t) = L$. Moreover, due to practical reasons it is assumed that the material properties will only allow $q_0 \in [-n\pi, n\pi]$ for some finite $n > 0$. For concise representations, the configuration variables are collected to $\mathbf{q}(t) = [q_0(t), \theta(t)]^\top$.

The orientation $\alpha_s(t)$ of the reference frame $\{S_s\}$ at point s along the central axis with

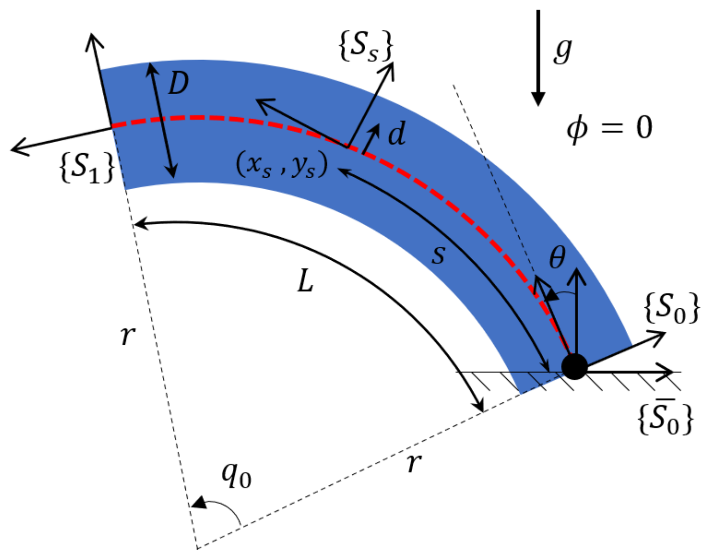


Figure 7.2: Exaggerated illustration of the SIPR. The central axis is shown in dashed red line. The non-extensible length of the soft is L and the width is D.

reference to the base frame $\{\bar{S}_0\}$, can be written as the sum of integral of the curvature and the base rotation,

$$\begin{aligned}\alpha_s(t) &= \theta(t) + \int_0^s q_0(t)\mathrm{d}l \\ &= \theta(t) + q_0(t)s.\end{aligned}$$

Thus the Cartesian coordinates $(x_{s,d}(t), y_{s,d}(t))$ at a general point parameterized by (s, d) on the soft robot is given by,

$$\begin{aligned}x_{s,d}(t) &= Dd\cos\alpha_s(t) - L\int_0^s \sin\alpha_s(t)\mathrm{d}s \\ y_{s,d}(t) &= Dd\sin\alpha_s(t) + L\int_0^s \cos\alpha_s(t)\mathrm{d}s.\end{aligned}$$

7.2.2 Dynamics

The dynamics of the soft robotic system were derived using a similar approach as in [34, 35] assuming a uniform mass distribution $\rho(s,d) \equiv m$. In the following, where obvious, the time arguments will be suppressed in the expressions due to brevity and clear representation.

For the SIPR system, the inertia matrix $\mathbf{M}(q_0,\theta) \in \mathbb{R}^{2\times 2}$ is evaluated as,

$$\mathbf{M}(q_0,\theta) = \int_0^1 \int_{-0.5}^{0.5} m \nabla_q (x_{s,d}, y_{s,d})^\top \nabla_q (x_{s,d}, y_{s,d}) \mathrm{d}d \, \mathrm{d}s$$

where ∇_q is the gradient operator $\nabla_q(\cdot) = \frac{\partial(\cdot)}{\partial q}$. Then the centrifugal and Coriolis terms matrix $\mathbf{C}(q_0,\theta,\dot{q}_0,\dot{\theta}) \in \mathbb{R}^{2\times 2}$ is evaluated using the standard Christoffel symbols [74].

The gravitational potential energy $P_g \in \mathbb{R}$ of the SIPR system is given by,

$$P_g = \int_0^1 \int_{-0.5}^{0.5} mg(x_{s,d} \sin(\phi) + y_{s,d} \cos(\phi)) \mathrm{d}d \, \mathrm{d}s$$

where ϕ defines the direction of the gravitational field which in this work $\phi = 0$. Therefore the gravity terms vetcor $\mathbf{G}(q_0,\theta) \in \mathbb{R}^2$ is evaluated as,

$$\mathbf{G}(q_0,\theta) = \nabla_q (P_g) \, .$$

The explicit terms for $\mathbf{M}(q_0,\theta)$, $\mathbf{C}(q_0,\theta,\dot{q}_0,\dot{\theta})$ and $\mathbf{G}(q_0,\theta)$ are given in the Appendix A.6.

Finally, adding the stiffness and damping terms the complete dynamics are,

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{B}_\beta \dot{\mathbf{q}} + \mathbf{K}_k \mathbf{q} + \mathbf{G}(\mathbf{q}) = \begin{bmatrix} \tau \\ 0 \end{bmatrix} \tag{7.1}$$

where the stiffness $\mathbf{K}_k = \left(\begin{smallmatrix} k & 0 \\ 0 & 0 \end{smallmatrix}\right)$ and damping $\mathbf{B}_\beta = \left(\begin{smallmatrix} \beta & 0 \\ 0 & 0 \end{smallmatrix}\right)$. Note that, only the degree of freedom associated with the curvature has damping and stiffness, and the base rotation is free.

Suppressing the arguments due to space and substituting for stiffness $\mathbf{K}_k$ and damping $\mathbf{B}_\beta$, the dynamics (7.1) can be re-written as,

$$\begin{bmatrix} \ddot{q}_0 \\ \ddot{\theta} \end{bmatrix} = \mathbf{M}^{-1} \begin{bmatrix} \tau \\ 0 \end{bmatrix} - \mathbf{M}^{-1} \left(\mathbf{C} \begin{bmatrix} \dot{q}_0 \\ \dot{\theta} \end{bmatrix} + \begin{bmatrix} \beta \dot{q}_0 \\ 0 \end{bmatrix} + \begin{bmatrix} k q_0 \\ 0 \end{bmatrix} + \mathbf{G} \right)$$

and for concise representation defining $f_{11}, f_{12}, f_{21}, f_{22}$ accordingly,

$$\begin{aligned} \ddot{q}_0 &= f_{11}\tau + f_{12} \\ \ddot{\theta} &= f_{21}\tau + f_{22}. \end{aligned} \tag{7.2}$$

7.2.3 Equilibria

The soft robot model (7.1) has two equilibrium points for $\tau = 0$. One is $(q_o, \theta, \dot{q}_0, \dot{\theta}) = (0, 0, 0, 0)$ which is the unstable vertical upright equilibrium positions. The other $(q_o, \theta, \dot{q}_0, \dot{\theta}) = (0, \pi, 0, 0)$ is the bottom-down stable equilibrium position. In this chapter, the bottom-down position should be avoided and the upright position should be reached.

7.2.4 Energy

Total energy of the SIPR system is given by,

$$\begin{aligned}
E =& \frac{1}{2}\dot{\mathbf{q}}^\top \mathbf{M}\dot{\mathbf{q}} + \frac{1}{2}\mathbf{q}^\top \mathbf{K}_k \mathbf{q} + P_g \\
=& \frac{1}{2}\begin{bmatrix} \dot{q}_0 \\ \dot{\theta} \end{bmatrix}^\top \mathbf{M} \begin{bmatrix} \dot{q}_0 \\ \dot{\theta} \end{bmatrix} + \frac{1}{2}kq_0^2 + \frac{Lgm}{q_0^2}\left(\cos\theta - q_0\sin\theta + \cos\left(q_0 - \theta\right)\right)
\end{aligned} \tag{7.3}$$

Now, differentiating the total energy (7.3) with respect to time and using (7.1) yields,

$$\begin{aligned}
\dot{E} =& \dot{\mathbf{q}}^\top \mathbf{M}\ddot{\mathbf{q}} + \frac{1}{2}\dot{\mathbf{q}}^\top \dot{\mathbf{M}}\dot{\mathbf{q}} + \dot{\mathbf{q}}^\top \mathbf{K}_k \mathbf{q} + \dot{\mathbf{q}}^\top \mathbf{G} \\
=& \dot{\mathbf{q}}^\top \left(\begin{bmatrix} \tau \\ 0 \end{bmatrix} - \mathbf{C}\dot{\mathbf{q}} - \mathbf{B}_\beta \dot{\mathbf{q}} - \mathbf{K}_k \mathbf{q} - \mathbf{G} \right) + \frac{1}{2}\dot{\mathbf{q}}^\top \dot{\mathbf{M}}\dot{\mathbf{q}} + \dot{\mathbf{q}}^\top \mathbf{K}_k \mathbf{q} + \dot{\mathbf{q}}^\top \mathbf{G} \\
=& \dot{\mathbf{q}}^\top \begin{bmatrix} \tau \\ 0 \end{bmatrix} - \dot{\mathbf{q}}^\top \mathbf{B}_\beta \dot{\mathbf{q}} \\
=& \tau\dot{q}_0 - \beta\dot{q}_0{}^2
\end{aligned} \tag{7.4}$$

where the skew symmetric property (*Property A.2*) of the Lagrangian systems is used.

7.3 Swing up control

This section develops the swing up control to bring the SIPR system near the vertical upright position. To that end, an energy-based swing up control inspired by [116] is used.

7.3.1 Control design

Recall that the control objective is to reach the top upright position $(q_o, \theta, \dot{q}_0, \dot{\theta}) = (0, 0, 0, 0)$. The total energy of the system in this configuration is,

$$E_d = \frac{mgL}{2}.$$

Now let us consider the following three conditions,

$$c1)\ E = E_d, \quad c2)\ q_0 = 0, \quad c3)\ \dot{q}_0 = 0. \tag{7.5}$$

If the above conditions $c1) - c3)$ are satisfied, then substituting L.H.S and R.H.S in (7.3) with explicit terms (see Appendix A.6) yields,

$$\frac{mgL}{2} = \frac{m(D^2 + 4L^2)\dot{\theta}^2}{24} + \frac{Lgm\cos\theta}{2}$$

resulting in the homoclinic orbit,

$$\alpha\dot{\theta}^2 = \frac{gL}{2}\left(1 - \cos\theta\right) \tag{7.6}$$

where $\alpha = \frac{m(D^2+4L^2)}{24}$. This will ensure that the soft pendulum will rotate clockwise or counter-clockwise until it reaches $(\theta, \dot{\theta}) = (0, 0)$. Therefore, for $(q_0, \dot{q}_0) = (0, 0)$, if the system can be driven to the above homoclinic orbit, it will solve the objective of "swinging up". Once the system reaches near the top upright position the control can be switched to a stabilizing controller.

This chapter considers an LQR for this purpose and it is discussed in Appendix A.7.

In view of the control objective for the "swing up", let us define the error in energy

$$\tilde{E} = E - E_d$$

and, the errors in degree of curvature and rate of change of degree of curvature as $\tilde{q_0} = q_0 - 0 = q_0$ and $\dot{\tilde{q_0}} = \dot{q}_0 - 0 = \dot{q}_0$, respectively. Following [116], we shall seek a Lyapunov based control by selecting the Lyapunov function candidate,

$$V(q,\dot{q}) = \frac{k_e \tilde{E}^2}{2} + \frac{k_d {\dot{q}_0}^2}{2} + \frac{k_p {q_0}^2}{2} \tag{7.7}$$

with $k_e, k_d, k_p > 0$ constant gains. Then differentiating V we get,

$$\begin{aligned}
\dot{V} &= k_e \tilde{E}\dot{E} + k_d \dot{q}_0 \ddot{q}_0 + k_p q_0 \dot{q}_0 \\
&= k_e \tilde{E}(\tau \dot{q}_0 - \beta {\dot{q}_0}^2) + k_d \dot{q}_0 \ddot{q}_0 + k_p q_0 \dot{q}_0 \\
&= \dot{q}_0 \left(k_e \tilde{E}(\tau - \beta \dot{q}_0) + k_d \ddot{q}_0 + k_p q_0 \right).
\end{aligned}$$

Let us now choose the control to satisfy,

$$\begin{aligned}
-\dot{q}_0 &= \left(k_e \tilde{E}(\tau - \beta \dot{q}_0) + k_d \ddot{q}_0 + k_p q_0 \right) \qquad (7.8) \\
&= \left(k_e \tilde{E}(\tau - \beta \dot{q}_0) + k_d (f_{11}\tau + f_{12}) + k_p q_0 \right) \\
&= \tau (k_e \tilde{E} + k_d f_{11}) + (-k_e \tilde{E} \beta \dot{q}_0 + k_d f_{12} + k_p q_0)
\end{aligned}$$

and therefore let,

$$\tau = \frac{\left(-\dot{q}_0 + k_e \tilde{E} \beta \dot{q}_0 - k_d f_{12} - k_p q_0\right)}{(k_e \tilde{E} + k_d f_{11})}. \tag{7.9}$$

This yields,

$$\dot{V} = -\dot{q}_0^2 \leq 0$$

which is negative semi-definite. The stability for this selection of the Lyapunov function candidate will be proved using the Lasalle's invariance principle subsequently in Theorem 7.1.

Observe that, here the control law (7.9) will have no singularities provided that,

$$(k_e \tilde{E} + k_d f_{11}) \neq 0 \qquad \forall t \geq 0. \tag{7.10}$$

which holds if,

$$\frac{k_d}{k_e} > \max_q \left(\frac{|\tilde{E}|}{|f_{11}|} \right) \tag{7.11}$$

Note that since f_{11} is the $(1,1)$ element of $\mathbf{M}^{-1}$ (inverse of the inertia matrix) and the inertia matrix $\mathbf{M}$ is bounded (*Property A.1*), f_{11} is bounded.

Also note that the control will produce no action if the soft robot is in either of the equilibrium positions. Hence, in order to exclude being stuck in the undesirable bottom-down equilibrium, it is required that,

$$|\tilde{E}| < |E_d - E_{bottom}| = mgL = c. \tag{7.12}$$

Since $\dot{V} \leq 0$, the Lyapunov function candidate V is non increasing and therefore the condition for $\tilde{E}$ will hold if the initial conditions are such that $V(0) \leq c^2/2$ for $k_e \geq 1$.

7.3.2 Stability analysis

This section presents the main result in Theorem 7.1 and use similar arguments as in [116, 117] to prove the stability using LaSalle's invariance principle.

Theorem 7.1 *Let the initial conditions are such that $|\tilde{E}| < c$ and $V(0) \leq \frac{c^2}{2}$ for the choice of Lyapunov function candidate (7.7). Then, with appropriately chosen $k_e, k_d, k_p > 0$ and satisfying the condition (7.11), the control law,*

$$\tau = \frac{\left(-\dot{q}_0 + k_e\tilde{E}\beta\dot{q}_0 - k_d f_{12} - k_p q_0\right)}{(k_e\tilde{E} + k_d f_{11})} \tag{7.13}$$

will drive the SPIR system (2.3) to the invariant set given by

$$Q = \left\{(q_0, \theta, \dot{q}_0, \dot{\theta}) : q_0 \equiv 0, \dot{q}_0 \equiv 0, \alpha\dot{\theta}^2 = \frac{gL}{2}(1 - \cos\theta)\right\}$$
$$\bigcup \left\{(q_0, \theta, \dot{q}_0, \dot{\theta}) : (q_0, \theta, \dot{q}_0, \dot{\theta}) = (\epsilon, 0, \epsilon, 0), |\epsilon| < \epsilon^*\right\}$$

where $\alpha = \frac{m(D^2+4L^2)}{24}$ and ϵ^* is arbitrarily small.

Proof of Theorem 7.1 *LaSalle's invariance principle is used to show that the controller is stable. Since* $\dot{V} = -\dot{q}_0^2 \leq 0$, V *is non-increasing. Therefore* $q_0, \dot{q}_0, \theta, \dot{\theta}$ *are bounded. Further,* $\tilde{E}$ *is bounded. Let the set* Ω *be the compact set where every solution for the soft robot system (2.3) remains in the same for all future time. Let* $\Gamma \in \Omega$ *such that* $\Gamma = \{(q_0, \theta, \dot{q}_0, \dot{\theta}) : \dot{V} \equiv 0\}$. *Let* $Q \in \Gamma$ *be the largest invariant set in* Γ. *LaSalle's Invariance Principle ensures that every solution starting in* Ω *approaches* Q *as* $t \to \infty$. *In the following, the largest invariant set* Q *is computed using the steps in [116].*

Considering Γ, $\dot{V} \equiv 0$ *which implies* $\dot{q}_0 \equiv 0$. *This implies that* $q_0 =$ *constant and* $\ddot{q}_0 \equiv 0$. *Also,* V *is constant. Therefore from (7.7)* $\tilde{E}$ *is constant. This implies that either* $\tilde{E} = 0$ *or* $\tilde{E} \neq 0$. *From control design (7.8) we obtain that,*

$$0 = k_e \tilde{E} \tau + k_p q_0. \tag{7.14}$$

Now, from (7.14), if $\tilde{E} = 0$, *then* $q_0 = 0$ *which means that since it was considered* $\dot{q}_0 = 0$, *the conditions* $c1) - c3)$ *in (7.5) are satisfied and the trajectory belongs to the homoclinic orbit (7.6).*

Next, considering $\tilde{E}$ *constant and* $\tilde{E} \neq 0$, *the condition (7.14) implies that* τ *is constant. Since* V *is nonincreasing,*

$$\begin{aligned} V = \frac{k_e \tilde{E}^2}{2} + \frac{k_p q_0^2}{2} &\leq V(0) \\ \frac{k_p q_0^2}{2} \leq V &\leq V(0) \\ \sqrt{k_p}|q_0| &\leq \sqrt{2V(0)} \end{aligned}$$

From (7.14),

$$
\begin{aligned}
k_e\tilde{E}\tau &= -\sqrt{k_p}\sqrt{k_p}q_0 \\
k_e|\tilde{E}\tau| = \sqrt{k_p}\sqrt{k_p}|q_0| &\leq \sqrt{k_p}\sqrt{2V(0)} \\
k_e|\tilde{E}\tau| &\leq \sqrt{k_p}c \\
|\tilde{E}\tau| &\leq \frac{\sqrt{k_p}}{k_e}c.
\end{aligned}
$$

Therefore, one can select k_p and k_e appropriately such that, $|\tilde{E}\tau|$ will be small implying that τ is small since $|\tilde{E}|$ is bounded.

Now consider that, since $\dot{q}_0 \equiv 0$, q_0 is constant. Considering the base rotation, it should be either constant (i.e, $\dot{\theta} = 0$) or rotating (i.e, $\dot{\theta} \neq 0$). If $\dot{\theta} \neq 0$ (rotating) then it will impose a gravity induced torque to the soft robot body that will change the degree of curvature q_0, which contradicts the fact that the curvature $q_0 =$ constant. Therefore, it is concluded that $\dot{\theta} \equiv 0$ and hence θ is a constant. Note that the curvature and the base rotation are stationary (i.e, $q_0 =$ constant and $\theta =$ constant) only if τ is exactly compensating gravity and the stiffness induced forces. It was earlier obtained that τ is chosen to be small. If q_0 is far from 0 then, τ will always be large to compensate for the stiffness. Therefore, q_0 has to be close to zero. For small τ, then $\theta = 0$ (upright) or $\theta = \pi$ (bottom-down). Since $\theta = \pi$ is excluded considering the initial conditions, wit can be concluded that θ is close to zero. Thus, both $|q_0| < \epsilon^*$ and $|\theta| < \epsilon^*$ for $\epsilon*$ arbitrarily small. Moreover, if $\tau = 0, \dot{q}_0 = 0, \ddot{q}_0 = 0$ and $q_0 = 0$, using (7.2) it can be easily shown that $\theta \equiv 0$.

Therefore, it is concluded that the largest invariant set Q is given by the set satisfying the homoclinic orbit (7.6) with $q_0 \equiv 0$ and $\dot{q}_0 \equiv 0$, and the interval $(q_0, \theta, \dot{q}_0, \dot{\theta}) = (\epsilon, 0, \epsilon, 0)$ where

$|\epsilon| < \epsilon^*$ *with* ϵ^* *arbitrarily small. This completes the proof.*

7.4 Simulation results

The simulation results for the combined swing up control and upright stabilization are presented in this section. This section considered a soft robotic system with length $L = 1$ m, width $D = 0.1$ m, mass $m = 2$ kg, stiffness $k = 0.5$ Nm/rad and damping $\beta = 0.1$ Nms/rad. The simulations were performed in MATLAB 2019a using the ode45 function. For the swing up control, $k_p = 1.1$, $k_d = 1.25$ and $k_e = 3$ was used.

Considering the LQR, the computed gain was $\mathbf{K} = [-84.74, -251.19, -23.54, -65.46]$. The region of attraction (ROA) for the LQR was computed using numerical simulation applying the controller at different initial conditions on the following ranges: $q_0 \in [-1.5, 1.5]$, $\theta \in [-1.5, 1.5]$, $\dot{q}_0 \in [-5, 5]$ and $\dot{\theta} \in [-5, 5]$. Here q_0, θ are in radians and $\dot{q}_0$, $\dot{\theta}$ are in rad/s.

First, the performance of the swing up control with the initial conditions $(q_o, \theta, \dot{q}_0, \dot{\theta}) = (\pi/72, \pi/1.2, 0, 0)$ is illustrated showing that as the curvature is getting closer to zero, the base rotation will remain swinging and it converges to a homoclinic orbit as in Figure 7.3. Note that the base rotation $\theta \in \mathbb{S}^1$ thus θ wraps around on itself every 2π. Therefore, once converged, the equilibrium points at the start and end of the orbit are the same.

Next, the simulation results are presented for the swing up control and stabilization at the upright position using LQR. Once the swing up control takes the system to the ROA, the control is switched to the LQR and the soft robot is stabilized in the upright position. The results are shown in Figure 7.4. Here also the system was initialized $(q_o, \theta, \dot{q}_0, \dot{\theta}) = (\pi/72, \pi/1.2, 0, 0)$. Around 9.35s the control switches to LQR as indicated in the Figure 7.4.

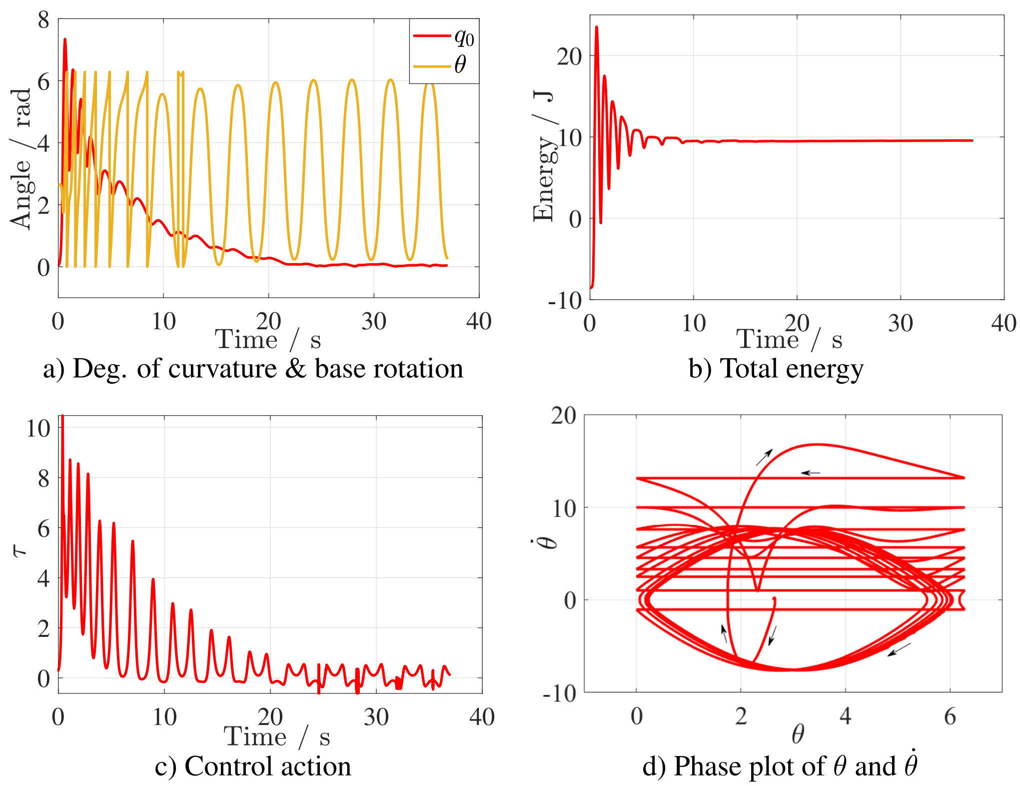


Figure 7.3: Illustration of the simulation results for swing up only starting from $(q_o, \theta, \dot{q}_0, \dot{\theta}) = (\pi/72, \pi/1.2, 0, 0)$. The phase plot shows that the trajectory is converging to a homoclinic orbit. The horizontal lines on the phase plot are due to the fact that $\theta \in \mathbb{S}^1$.

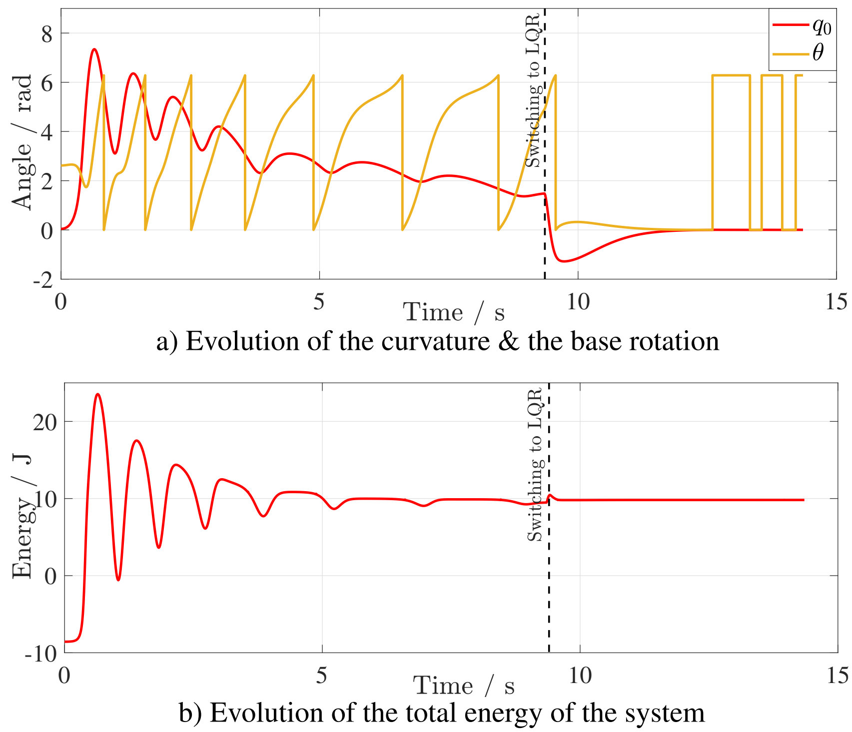


Figure 7.4: Simulation results for combined control of swing up and LQR are shown here. The control is switched from swing up to LQR around 9.35s.

7.5 Summary

A novel underactuated soft robotic system- soft inverted pendulum with revolute base (SIRP) was introduced in this chapter. The soft robot was considered to have a constant curvature which was actuated while the base was allowed to rotate freely. The dynamics of the system were derived and a switching based control was developed. For swing up, an energy-based control method was developed and for stabilization of the soft system at the upright unstable equilibrium an LQR was used. The stability of the proposed control approach was analyzed and simulation results were presented to illustrate the efficacy of the proposed control framework.

Chapter 8: Output Partial Feedback Linearization for Soft Inverted Pendulum

This chapter proposes a general control architecture for the swing-up and stabilization of underactuated mechanical systems and its application to stabilize the SIPR at its upright equilibrium. This is achieved by employing output partial feedback linearization (OPFL) and linear control. First, a suitable output, namely the center of mass angle for the considered underactuated system, is chosen. Next, OPFL is done to stabilize the origin of the output dynamics, neglecting possible disturbance forces from a simultaneous additional control action. An LQR is then designed using this output partial feedback linearized system around the upright equilibrium point. Using numerical examples, it is shown that the region of attraction is increased significantly compared to a nominal LQR without the OPFL step. Simulation results are presented for the SIPR, illustrating the efficacy of the proposed control method. Further, in the Appendix A.8 simulation results are presented for two additional underactuated systems, the Reaction Wheel Pendulum and Acrobot, showing the general applicability of the proposed control architecture.

8.1 Introduction

The SIPR is an underactuated mechanical system as discussed in Chapter 7. An underactuated mechanical system has fewer independent control inputs than the number of generalized coordinates (degrees of freedom - DoFs) [74]. This underactuation is mainly due to the nature of the dynamics of the system, such as in quadcopters [127], or by design, such as in theoretically motivated low order example systems like SIPR in Chapter 7, Acrobot [111], Ball and Beam system [128] and Cart-pole system [117]. The underactuation can also be due to actuator failure in an otherwise fully actuated system [129]. In general, control of nonlinear underactuated systems is a challenging problem as the number of actuators is less than the DOFs to be controlled, and many nonlinear control methods for traditional fully actuated systems are not directly applicable [114, 115].

In underactuated systems resembling inverted pendulums (e.g., SIPR, Acrobot, Pendubot, Cart-pole), the primary control approach involves a two-step process [130]. Initially, the system is swung up from its downward position using a swing-up controller, such as using partial feedback linearization [130] and energy-based methods [117]. Subsequently, a balancing controller is engaged to maintain the pendulum upright. This balancing controller is often designed through linearization or gain scheduling [131]. Recall that in Chapter 7, an energy-based swing up controller and an LQR for swinging up and stabilizing the SIPR system was used.

Recent efforts have been on methods to achieve swing up and stabilization using a single continuous controller to avoid switching between controllers. To that end, a nonlinear optimal control-based swing up and stabilization of the Acrobot via a stable manifold approach is proposed in [132]. In [133], a nonlinear controller using quotient manifolds is proposed to locally asymptotically stabilize the Acrobot in the neighborhood of the upwards position and increasing the domain of attraction by tuning to include the lower equilibrium point so that the controller can achieve both swing-up and balancing. The authors in [134] proposed a controller for the swing up and balancing of an Acrobot based on an equivalent input-disturbance approach. While these are designs of controllers for specific underactuated systems, [135] considers the utilization of backstepping for the problem of stabilization of a class of underactuated systems whose inertia matrix is dependent only on the actuated configuration variables (*Class-I underactuated systems* according to the definition in [131]). However, the backstepping approach requires the system to be represented in the cascade normal form [131], which is nontrivial. Authors in [124] proposed an Interconnection and Damping Assignment (IDA-PBC) approach to swing up and balance without switching between controllers. However, this method requires solving a set of partial differential equations (PDEs), which is generally a challenging problem. In recent work, the authors in [125] proposed an energy-shaping swing-up controller to avoid solving PDEs. However, both of these control frameworks are only applicable for a class of underactuated systems whose inertia matrix is dependent only on the unactuated configuration variables (*Class-II underactuated systems* [131]).

This chapter focuses on the SIPR system and observe that it is in the class of *Class-I underactuated systems*. To that end, in view of developing a general strategy to advance soft robotics as well as control of underactuated systems at large, this chapter analyzes the *Class-I underactuated systems* and proposes a simple control architecture that enlarges the ROA of a linear controller (e.g., LQR) for these underactuated systems linearized around a desired equilibrium. Specifically, the swing-up and stabilizing problem of pendulum-like *Class-I underactuated systems* is investigated. The proposed control framework consists of two simultaneous control inputs to the system. The first is an OPFL based control computed for swinging up the system. The second is a linear controller (LQR in this analysis) computed by linearizing the output partial feedback linearized system around the desired upright equilibrium.

Note that when deriving the OPFL-based control, only the nominal system is considered neglecting the LQR control input.

The main contribution of this chapter is in the development of the simultaneous control strategy to enlarge the ROA of a linear controller for swing up and stabilization of *Class-I underactuated systems* circumventing the need for switching based control or needing to transform the system into cascade normal form. It is also possible to utilize the proposed control instead of the widely used LQR control as the switched controller for stabilization after employing a swing-up controller. Moreover, an extensive numerical analysis is performed on the SIPR system showing that the proposed control framework enlarges the linearized controller's ROA compared to implementing a linearized controller without the OPFL control. Additionally, numerical examples are presented on two more *Class-I underactuated systems*- Reaction Wheel Pendulum (RWP) and Acrobot [111] in Appendix A.8 illustrating the efficacy of the proposed control framework for simultaneous control of swing-up and stabilization of different underactuated mechanical systems in the considered class.

The rest of the chapter is organized as follows. Section 8.2 defines the nonlinear underactuated mechanical system. The control design is discussed in Section 8.3 and the numerical simulations for the SIPR are presented in Section 8.4. Finally, the chapter is summarized in Section 8.5.

8.2 The class of underactuated system

The variables that appear in the inertia matrix of the system are called *shape variables* [131]. A system is defined as a *Class-I underactuated system* if and only if the shape variables are actuated. SIPR is such a system. Additional examples of such systems are the Acrobot [111], RWP [136] and the Translational Oscillator with Rotational Actuator (TORA) system [137]. Letting the configuration variables of the system be $\mathbf{q}(t) = [q_1(t), q_2(t)]^\top \in \mathbb{R}^2$, the dynamics of these underactuated systems can be represented as,

$$m_{11}(q_2)\ddot{q}_1 + m_{12}(q_2)\ddot{q}_2 + h_1(q_1, q_2, \dot{q}_1, \dot{q}_2) = 0 \tag{8.1}$$

$$m_{21}(q_2)\ddot{q}_1 + m_{22}(q_2)\ddot{q}_2 + h_2(q_1, q_2, \dot{q}_1, \dot{q}_2) = \tau(t) \tag{8.2}$$

with inertia terms $m_{i,j} \in \mathbb{R}$ for $i, j = 1, 2$ and $h_1, h_2 \in \mathbb{R}$ collecting the terms such as centripetal, Coriolis, and gravity. Observe in the dynamics that the shape variable is $q_2(t)$ and it is actuated by the control input $\tau(t) \in \mathbb{R}$. Thus, it is in *Class-I underactuated* form. Note that, a slight modification in the notation used for the SIPR in Chapter 7 is done in this chapter by defining the base rotation $\theta = q_1$ and degree of curvature $q_0 = q_2$ for the dynamics to be written in the above form.

In this chapter *Class-I underactuated systems* are analyzed with an emphasis on pendulum-like systems that can be swung up and stabilized since our main focus is on swing up and stabilizing the SIPR system. From a general stand point, it is considered that the dynamics of these systems are formulated such that $[0, 0]^\top$ is the upright equilibrium point. Moreover, when obvious, the parameter dependencies are excluded for conciseness in the subsequent sections and the control architecture is developed considering a general *Class-I underactuated system*.

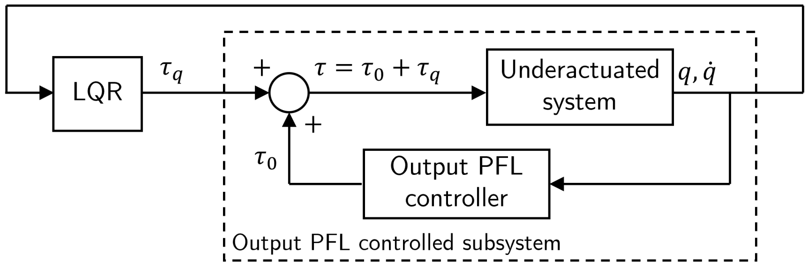


Figure 8.1: The control system architecture.

8.3 Control design

The objective of the control design is to swing up and drive pendulum-like *Class-I underactuated systems* to the upright equilibrium point $\mathbf{q_d} = [0, 0]^\top$. In this chapter, a new control architecture is proposed by employing an OPFL-based control and a LQR around the linearized system. The control system is illustrated in Figure 8.1. Specifically, the control torque $\tau(t)$ is considered as a combined torque consisting two separate control torques

$$\tau(t) = \tau_0(t) + \tau_q(t).$$

Here τ_0 is the control term derived from an OPFL-based controller neglecting τ_q, using the nominal dynamics,

$$\begin{aligned} m_{11}\ddot{q}_1 + m_{12}\ddot{q}_2 + h_1 &= 0 \\ m_{21}\ddot{q}_1 + m_{22}\ddot{q}_2 + h_2 &= \tau_0 \end{aligned} \tag{8.3}$$

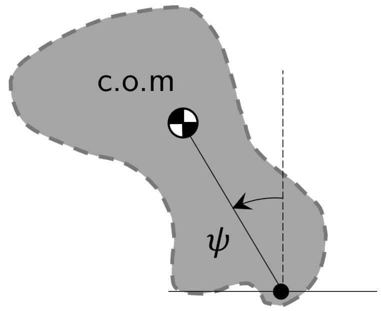


Figure 8.2: Center of mass angle of a system

The other control term, τ_q, is derived using LQR by linearizing the dynamics,

$$
\begin{aligned}
m_{11}\ddot{q}_1 + m_{12}\ddot{q}_2 + h_1 &= 0 \\
m_{21}\ddot{q}_1 + m_{22}\ddot{q}_2 + h_2 - \tau_0 &= \tau_q
\end{aligned}
\tag{8.4}
$$

around the desired upright equilibrium $\mathbf{q_d}$.

In the remainder of this section, the control derivation is discussed. First, the output signal is identified and then the OPFL control and the LQR input are introduced.

8.3.1 Output

Consider as the output a signal $y(q_1, q_2) \in \mathbb{R}$ for which the desired upright equilibrium point $\mathbf{q_d} = [0, 0]^\top$ is a solution to $y(q_1, q_2) = 0$. Moreover, it it assumed that $y(q_1, q_2)$ is at least twice differentiable with a relative degree 2. To that end, the center of mass (COM) angle as shown in Figure 8.2 is an obvious candidate for the output signal. In fact, in this chapter the COM angle is used as the output in the presented numerical examples.

8.3.2 Output partial feedback linearization control

The OPFL control is derived from the nominal dynamics (8.3), neglecting the control input τ_q. First, rewrite (8.3) in the normal form [138] for clarity in the presentation. Using,

$$\tau_0 = h_2 - \frac{m_{21}}{m_{11}} h_1 + \bar{M} u_2 \tag{8.5}$$

where $u_2(t) \in \mathbb{R}$ is an additional outer loop control and $\bar{M} = \left(m_{22} - \frac{m_{12} m_{21}}{m_{11}} \right)$ yields the normal form

$$m_{11} \ddot{q}_1 = -h_1 - m_{12} u_2, \qquad \ddot{q}_2 = u_2. \tag{8.6}$$

Considering the output $y(q_1, q_2)$ and differentiating it yields $\dot{y} = J_1 \dot{q}_1 + J_2 \dot{q}_2$ where $J_1(q_1, q_2) = \frac{\partial y}{\partial q_1}$ and $J_2(q_1, q_2) = \frac{\partial y}{\partial q_2}$. Computing the second derivative of $y(q_1, q_2)$ yields,

$$\ddot{y} = J_1 \ddot{q}_1 + J_2 \ddot{q}_2 + \dot{J}_1 \dot{q}_1 + \dot{J}_2 \dot{q}_2.$$

and substituting for $\ddot{q}_1, \ddot{q}_2$ from (8.6) yields,

$$\ddot{y} = \bar{J} u_2 + \eta.$$

Here $\bar{J} = J_2 - \frac{J_1 m_{21}}{m_{11}}$ and $\eta = \dot{J}_1 \dot{q}_1 + \dot{J}_2 \dot{q}_2 - \frac{J_1 h_1}{m_{11}}$.

Let the outer loop control a and define the control law

$$u_2 = \bar{J}^{+}(a - \eta) \tag{8.7}$$

where $\bar{J}^{+}$ is the pseudo inverse. This will drive the output dynamics to $\ddot{y} = a$. Choosing the outer loop control as

$$a = -k_p y - k_d \dot{y} \tag{8.8}$$

with control gains $k_p, k_d > 0$ will exponentially stabilize the output at the origin considering the nominal dynamics.

8.3.3 The linear controller (LQR)

Let the state vector, $\mathbf{x}(t) \in \mathbb{R}^4$ with state variables $x_1 = q_1$, $x_2 = q_2$, $x_3 =, \dot{q}_1$, $x_4 = \dot{q}_2$ and input $u(t) = \tau_q(t)$. One can write the state equations $\dot{\mathbf{x}} = f(\mathbf{x})$ with elements,

$$\dot{x}_1 = x_3, \qquad \dot{x}_2 = x_4, \qquad \dot{x}_3 = \bar{J}^{+}a + \left(\frac{1}{\bar{M}}\right) u - \bar{J}^{+}\eta,$$
$$\dot{x}_4 = -\left(\frac{m_{21}}{m_{22}}\bar{J}^{+}\right) a - \left(\frac{m_{21}}{m_{22}}\frac{1}{\bar{M}}\right) u + \left(\frac{m_{21}}{m_{22}}\bar{J}^{+}\eta - \frac{h_2}{m_{22}}\right)$$

where the dynamics given in (8.4) have been used with the control τ_0 as given in (8.5) along with the control u_2 as in (8.7). The outer loop control a is defined as in (8.8).

Linearizing $\dot{\mathbf{x}} = f(\mathbf{x})$ around the reference equilibrium $\mathbf{x}_0 = [0, 0, 0, 0]^\top$ and the reference input $u_0 = 0$, the linear time-invariant system is obtained as,

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(\mathbf{t}) + \mathbf{B}u(t)$$

where $\mathbf{A} = \frac{\partial f}{\partial \mathbf{x}}\big|_{(\mathbf{x}_0, u_0)} \in \mathbb{R}^{4\times 4}$ and $\mathbf{B} = \frac{\partial f}{\partial u}\big|_{(\mathbf{x}_0, u_0)} \in \mathbb{R}^{4\times 1}$. For stabilization at the upright equilibrium, the LQR problem can be formulated,

$$\min_{u} \quad \int_0^T \frac{1}{2}\left(\mathbf{x}^\top \mathbf{Q}\mathbf{x} + \mathbf{R}u^2\right) dt$$

to find the control input $u(t)$ which is of the form,

$$u(t) = -(\mathbf{R}^{-1}\mathbf{B}^\top \mathbf{P})\mathbf{x}(t) = -\mathbf{K}\mathbf{x}(t). \tag{8.9}$$

Here $\mathbf{R} > 0$ and $\mathbf{P}$ is the solution to the algebraic Riccati equation $\mathbf{A}^\top\mathbf{P} + \mathbf{P}\mathbf{A} - \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^\top\mathbf{P} + \mathbf{Q} = 0$ where $\mathbf{Q}$ is a positive semidefinite matrix. Thus, with the use of $\tau_q = u = -\mathbf{K}\mathbf{x}$ as derived above, the origin of the linearized system is asymptotically stable. Hence, from Lyapunov's indirect method [139], it can be established that the origin of the nonlinear system is locally asymptotically stable.

Remark 8.1 *Only local asymptotic stability of the origin if guaranteed by employing the LQR for the linearized system with the control from the OPFL step. However, it is observed that the proposed approach enlarges the ROA considerably compared to using LQR without the OPFL input. This chapter presents numerical examples illustrating this claim, wherein the formal proof of this claim is left for future work.*

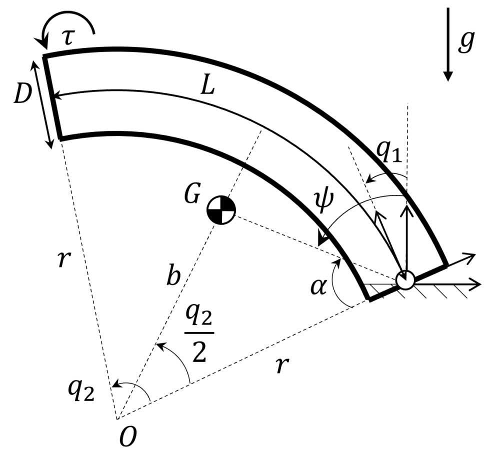


Figure 8.3: The soft inverted pendulum with a revolute base (SIPR).

8.4 Numerical analysis of OPFL based control for SIPR

In this section, the proposed control architecture is applied for swing up and stabilization of the SIPR system. The performance is compared with a nominal LQR without the OPFL control input. The numerical simulations were run on Matlab 2022b using `ode45` as the ODE solver.

Applying the derivation method given in Chapter 7 for the SIPR as shown in Figure 8.3 with base rotation $q_1(t)$ and curvature $q_2(t)$, the dynamics of the SIPR is found as

$$m_{11}\ddot{q}_1 + m_{12}\ddot{q}_2 + h_1 = 0$$

$$m_{21}\ddot{q}_1 + m_{22}\ddot{q}_2 + h_2 = \tau$$

with the mass terms,

$$
\begin{aligned}
m_{11} &= \frac{mD^2}{12} + \frac{2mL^2}{q_2^3}\left(q_2 - \sin q_2\right) \\
m_{12} &= \frac{mD^2}{24} + \frac{mL^2}{4q_2^4}\left(1 - \cos q_2 - q_2 \sin q_2 + \frac{q_2^2}{2}\right) \\
m_{21} &= m_{12} \\
m_{22} &= \frac{mD^2}{36} + \frac{2mL^2}{q_2^5}\left(q_2 - 2\sin q_2 + q_2 \cos q_2 + \frac{q_2^3}{6}\right).
\end{aligned}
$$

Here h_1 and h_2 collects the centripetal, Coriolis, gravity, stiffness (k) and damping (β) terms as,

$$
\begin{aligned}
h_1 =& c_{11}\dot{q}_1 + c_{12}\dot{q}_2 + g_1 + kq_1 + \beta\dot{q}_1 \\
h_2 =& c_{21}\dot{q}_1 + c_{22}\dot{q}_2 + g_2
\end{aligned}
$$

with centripetal and Coriolis terms,

$$
\begin{aligned}
c_{11} &= -\frac{L^2\, m\, \dot{q}_2}{{q_2}^4}\Big(2\, q_2 - 3\, \sin q_2 + q_2\, \cos q_2\Big) \\
c_{12} &= -\frac{L^2\, m}{{q_2}^5}\Big(4\dot{q}_2^2 + q_2^2\,(1 + \cos q_2\,)(\,\dot{q}_1 + \dot{q}_2) - 4\dot{q}_2 \cos q_2 - q_2 \sin q_2\,(3\dot{q}_1 + 4\dot{q}_2)\,\Big) \\
c_{21} &= \frac{L^2\, m\, \dot{q}_1}{{q_2}^4}\Big(2\, q_2 - 3\, \sin q_2 + q_2\, \cos q_2\Big) \\
c_{22} &= \frac{L^2\, m\, \dot{q}_2}{3\, {q_2}^3}\left(3\left(10 - q_2^2\right)\frac{\sin q_2}{q_2^3} - \frac{12 + 18\cos q_2}{q_2^2} - 1\right)
\end{aligned}
$$

and the gravity terms,

$$g_1 = -\frac{L\,g\,m\,\left(\sin\left(q_1\right) - \sin\left(q_2 + q_1\right) + q_2\,\cos\left(q_1\right)\right)}{{q_2}^2}$$
$$g_2 = \frac{2\,L\,g\,m\,\left(\cos\left(q_2 + q_1\right) - \cos\left(q_1\right) + q_2\,\sin\left(q_1\right)\right)}{{q_2}^3} + \frac{L\,g\,m\,\left(\sin\left(q_2 + q_1\right) - \sin\left(q_1\right)\right)}{{q_2}^2}.$$

The output function in this example is the COM angle of the soft link $y(t) = \psi(t)$. Obtaining ψ referring to Figure 8.3 will be briefly discussed. Due to the constant curvature assumption, the link can be considered as an arc. Hence, the COM, G, of the soft link with respect to the center of the curve O is located at a distance $OG = \frac{r\sin(q_2/2)}{(q_2/2)}$. Since the radius $r = L/q_2$ the COM distance is,

$$b := \frac{2L\sin(q_2/2)}{q_2^2}.$$

Now the angle α can be found as,

$$\alpha(q_2) = \text{atan}\left(\frac{b\sin(q_2/2)}{r - b\cos(q_2/2)}\right) = \text{atan}\left(\frac{1-\cos(q_2)}{q_2 - \sin(q_2)}\right).$$

Thus the angle of the COM w.r.t the vertical axis is,

$$\psi = (\pi/2 - \alpha) + q_1 = \pi/2 + \text{atan}\left(\frac{\cos(q_2) - 1}{q_2 - \sin(q_2)}\right) + q_1.$$

Thus, with the output defined as $y = \psi$, the Jacobians, in this case, will be

$$J_1 = 1, \qquad J_2(q_2) = \frac{2\,\cos(q_2) + q_2\,\sin(q_2) - 2}{2\,\cos(q_2) + 2\,q_2\,\sin(q_2) - {q_2}^2 - 2}.$$

Parameters used for the SIPR system are $m = 2$, $L = 1$, $D = 0.1$, $k = 0.5$, $\beta = 0.1$ and $g = 9.81$. The control gains $k_p = 2.5 \times 10^4$ and $k_d = 5 \times 10^2$ are used for the OPFL controller. For the LQR in (8.9) $Q = \mathbb{I}_{4\times 4}$ and $R = 1 \times 10^6$ are used and obtained the LQR gain

$$\mathbf{K} = [-1.09, -0.36, -3.62, -1.36].$$

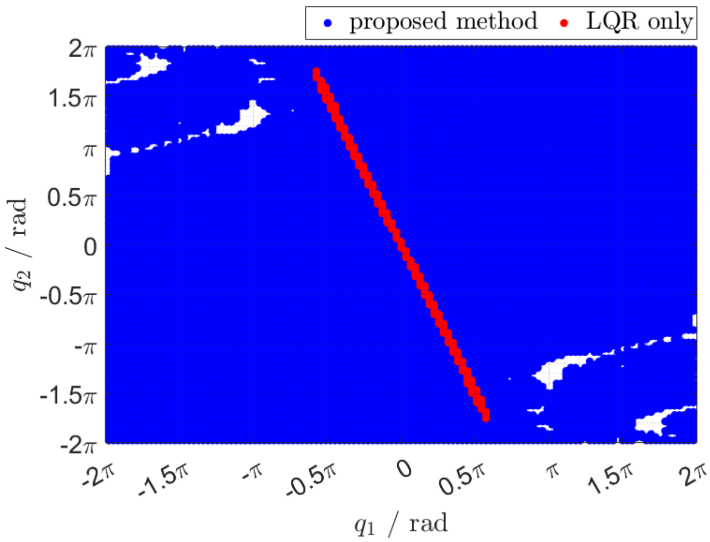
For comparison, the nominal LQR gains were computed using $Q = \mathbb{I}_{4\times 4}$ and $R = 1 \times 10^5$ and obtained

$$\mathbf{K}_{nom} = [-26.46, -8.82, -6.80, -2.50]$$

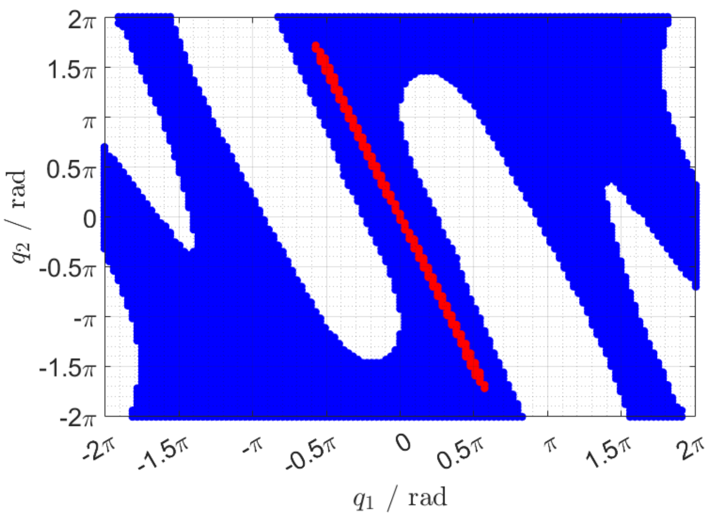
implementing the control as $\tau = -K_{nom}\mathbf{x}$ in (8.2).

The ROA slice at $q_1 - q_2$ plane was done considering the region $q_1, q_2 \in [-2\pi, 2\pi]$ with $\dot{q}_1 = \dot{q}_2 = 0$ to find the ROA numerically. The ROA for the unrestricted torque case is illustrated in Figure 8.4a, and the case with torque saturation at 50Nm is illustrated in Figure 8.4b. For the unrestricted case, observe that the ROA is increased by over 100-fold (10578.1%) using the proposed method, while for the torque saturated case, the proposed method enlarges the ROA by 60-fold (6377.3%). In the SIPR system, the ROA of the nominal LQR is extremely limited, and using the proposed method significantly enlarges the ROA for swing-up and stabilization.

An unrestricted torque simulation is illustrated in Figure 8.5 starting from initial conditions



a) Unrestricted torque.



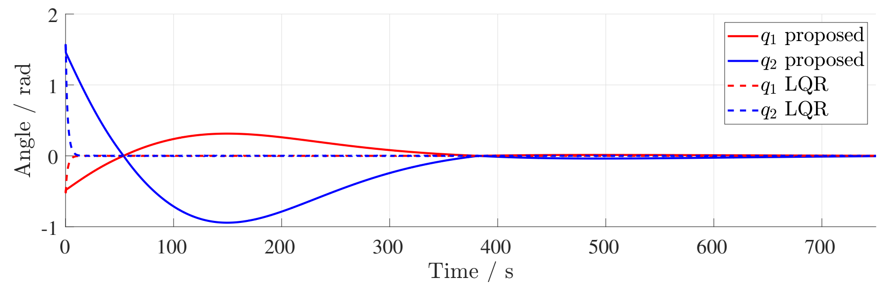
b) Torque saturated at 50Nm.

Figure 8.4: ROA slice for the SIPR for $q_1, q_2 \in [-2\pi, 2\pi]$ when $\dot{q}_1 = \dot{q}_2 = 0$.

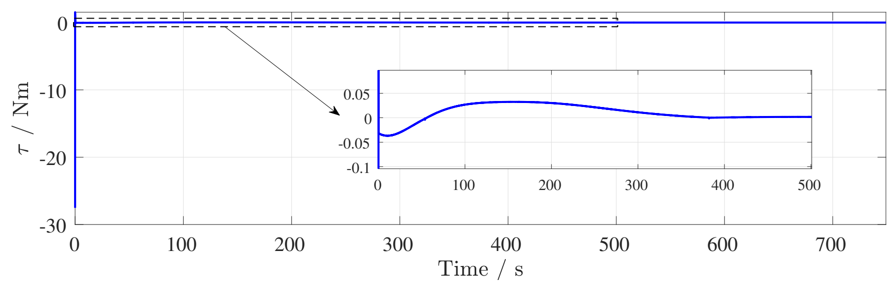
$(-\frac{\pi}{6}, \frac{\pi}{2}, 0, 0)$ which is within ROAs of both the controllers. Note that in this case, the torques are indeed within the saturation limits. The swing up and balancing of the SIPR initializing from the downward equilibrium point $(\pi, 0, 0, 0)$ with unrestricted torque is illustrated in Figure 8.6a-c. Note that this point is within the ROA of the proposed method but outside that of the nominal LQR controller. Thus, the LQR diverges, and unstable behavior is observed. The torque saturated behavior of the proposed method for this case is illustrated in Figure 8.6a,d.

8.5 Summary

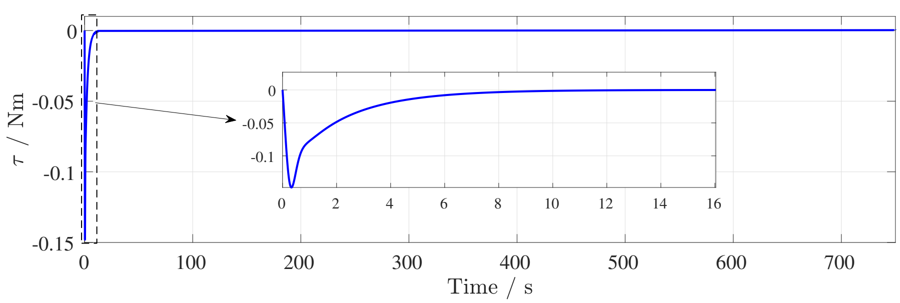
This chapter proposes a control method for the swing up and stabilization of the SIPR by developing the control architecture considering general *Class-I underactuated mechanical systems* by employing OPFL control and a linear control (LQR) simultaneously. Using numerical simulations on the SIPR system, it is shown that the ROA is increased significantly compared to a nominal linear controller without the OPFL input. In the case of unrestricted torque input, it is shown in numerical simulations that the considered systems can be swung up from the bottom-down equilibrium and stabilized at the upright equilibrium only using the proposed control method. However, the system's actuator has to provide a large impulsive input, as observed in the simulations. One way to achieve this is to initially load the actuator, enabling it to exert this sudden input. The case when the torque is saturated was also numerically investigated and observed that this limitation reduces the ROA; however, the observed ROA is still larger than that of a nominal linear controller without the OPFL step. Note that the proposed control approach could also be an alternative to the commonly used LQR method when transitioning to the stabilization phase after employing a dedicated swing-up controller.



a) Evolution of angles.

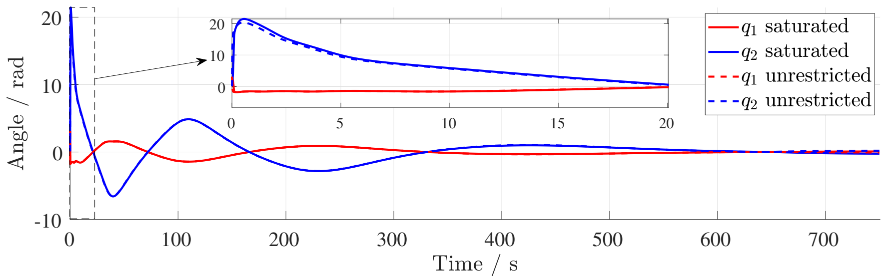


b) Torque evolution for the proposed method.

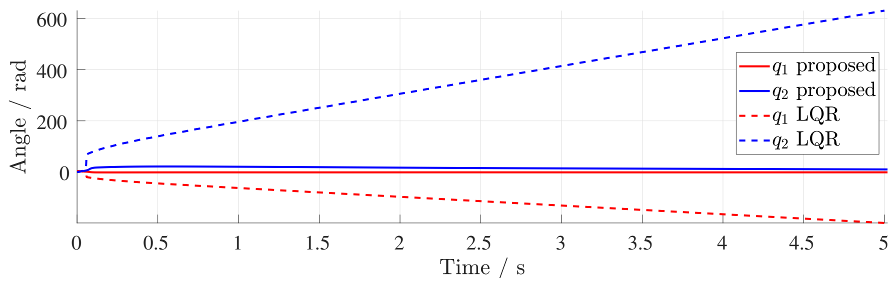


c) Torque evolution for the nominal LQR.

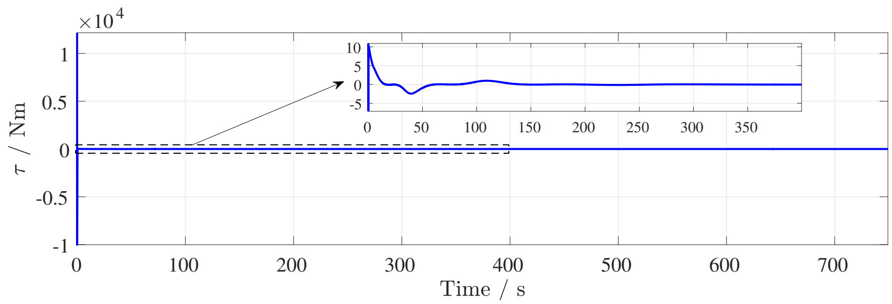
Figure 8.5: Simulation results for SIPR for initial conditions $(-\frac{\pi}{6}, \frac{\pi}{2}, 0, 0)$ which is inside both ROAs.



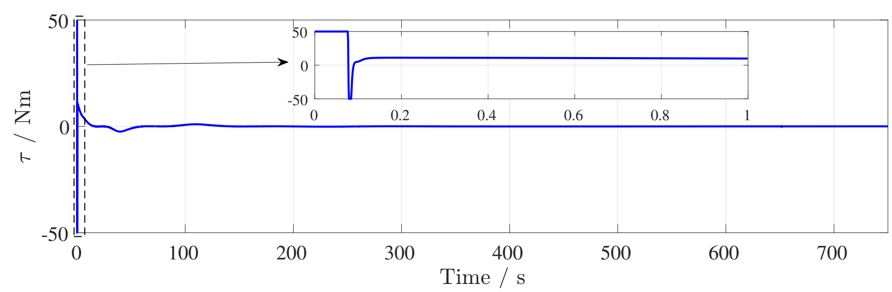
a) Evolution of angles for the proposed method.



b) Evolution of angles for the nominal LQR showing unstable behavior with unrestricted input.



c) Torque evolution for the proposed method with unrestricted input.



d) Torque evolution for the proposed method with saturated input.

Figure 8.6: Simulation results for SIPR with initial conditions $(\pi, 0, 0, 0)$.

Chapter 9: Conclusions

This dissertation discussed dynamic control of dexterous soft robotic systems addressing uncertainty mitigation and robust stabilization of soft robotic systems, utilization of stretchable soft sensors for closed-loop control of soft robots, dexterous and precise manipulation of hybrid rigid-soft robots, and swing-up and stabilization of underactuated soft robots. First, a summary of contributions of this dissertation is given below. Then, some significant dissertation limitations are pointed out and potential future research directions are noted.

9.1 Summary of contributions

Chapter 2 detailed the development of a passivity-based adaptive controller for achieving task space trajectory tracking in soft robots. Furthermore, it introduced an adaptive bilateral teleoperation framework designed for a system featuring a non-redundant rigid leader manipulator alongside a redundant PCC soft follower manipulator. These adaptive controllers were effective in ensuring task space tracking for the soft robot and synchronization between the robots, even in scenarios with time-invariant parameter uncertainties affecting either the leader or soft follower side. Specifically, the task space position errors were shown reaching the origin asymptotically. Within the bilateral teleoperation framework, the redundancy present in the soft follower manipulator was utilized to accomplish sub-task objectives, such as adhering to curvature limits while simultaneously tracking the position of the leader robot. Both simulation and experimental results substantiated strong performance in terms of task space tracking and task synchronization. Notably, the bilateral teleoperation framework revealed high immersion in the remote environment due to the projected force feedback. This work on utilizing passivity-based adaptive control approaches for soft robots were investigated for the first time, in this dissertation. The results on the adaptive task space bilateral teleoperation framework from this chapter was presented in part at [73].

In Chapter 3, an enhanced dynamics model for planar PCC soft robots was introduced using the Euler-Lagrange formulation. This model takes into explicit account the mass distribution of the soft robot, thus capturing the significant inertia effects often omitted in simplified models that treat the robot as a lumped mass system. Building upon this dynamics model, a passivity-based robust controller was developed for task space tracking proving the task space errors were uniformly ultimately bounded. The effectiveness of the proposed controller was demonstrated through simulations and physical experiments. This dissertation explored the passivity-based robust control for soft robot control for the first time.

In Chapter 4, the robust task space bilateral teleoperation of soft robots was explored, employing a rigid leader robot and a soft follower robot. To achieve this, passivity-based robust control techniques were utilized, which are particularly effective in the presence of unmodeled external disturbances, dynamics uncertainty, and constant asymmetric time delays. The focus is on creating task space bilateral teleoperators that remain stable under the influence of passive or non-passive external forces, whether they stem from a human operator or interactions with the environment. Specifically, the theoretical results in this chapter proves the system trajectories are uniformly ultimately bounded. Additionally, the redundancy in the soft follower robot was exploited via null space velocity tracking to enable collision avoidance as a sub-task within the teleoperation framework. The effectiveness of this approach is demonstrated through extensive simulations and physical experiments, highlighting its practical utility. This work on proposing a passivity-based robust bilateral teleoperation for task space synchronization of a dissimilar leader-follower system is a significant advancement in the controls literature.

In Chapter 5, the application of integrated sensing for dynamic control of soft robots, operating under the PCC modeling hypothesis was showcased. The innovative soft sensing skins introduced in this research are adaptable and can be potentially retrofitted to a wide array of soft robots. Furthermore, an LSTM network was employed to learn the estimation of the degree of curvature, relying on strain signals from the sensing skin and actuator inputs. Additionally, a passivity-based adaptive controller was designed to track a desired degree of curvature trajectory in the presence of parameter uncertainty. The adaptive controller ensures that the curvature errors reach the origin asymptotically. The favorable performance observed in the degree of curvature tracking for both low and high-frequency target trajectories attests to the robustness of the proposed soft skins in estimating curvature and holds promise for integration within a dynamic control framework. This was the first time such an integrated sensing and control framework was demonstrated. The results stemming from this chapter was presented in part at [94].

In Chapter 6, a novel robotic system known as a hybrid rigid-soft (HyRiSo) robot was introduced. This robot was comprised of a combination of rigid and soft links serially connected. These robots are particularly intriguing due to their enhanced dexterity, resulting from including soft links within the system. However, the hybrid nature of this robot introduces parameter uncertainties, mainly due to the heterogeneous actuation modes of the revolute joints and the soft links. Consequently, uncertain actuator mapping becomes critical in controlling such hybrid robots. In response to this challenge, two passivity-based controllers: adaptive and robust were devised. These controllers were designed to address the complexities of the hybrid robot's uncertain actuation system and other uncertainties. The effectiveness of the proposed hybrid robotic system, along with these controllers, was demonstrated through numerical examples that showcased the system's capability for task space trajectory tracking, even within challenging workspace environments. The results from this chapter was presented at [107].

In Chapter 7, a unique underactuated soft robotic system called the SIPR: soft inverted pendulum with a revolute base was introduced. This soft robot was designed with an actuated constant curvature soft body, while its base joint had the freedom to rotate freely. The system's dynamics were derived and a control strategy was devised based on switching techniques. For the swing-up phase, an energy-based control method was developed, and to stabilize the soft system at the unstable upright equilibrium, an LQR approach was applied. A thorough stability analysis of the proposed control methodology was conducted and simulation results were presented demonstrating this control framework's effectiveness. The results stemming from this chapter was presented in part at [112].

In Chapter 8, a continuous control method was developed for the swing-up and stabilization of the SIPR system without switching between controllers. This control architecture was devised with consideration for general *Class-I underactuated mechanical systems* by simultaneously applying an output partial feedback linearization control and a linear control method, such as LQR. It should be noted that the stability guarantees are only local as the control is derived considering a linearized system near the upright equilibrium. Through numerical simulations involving the SIPR system, a significant increase in the ROA compared to using a standard linear controller without the OPFL input was demonstrated. In scenarios where the torque input is not restricted, our simulations revealed that the SIPR could be effectively swung up from the bottom-down equilibrium and stabilized in the upright position solely through the proposed control approach. However, it's worth noting that in such cases, the actuator must provide a substantial impulsive input, as observed in our simulations. One approach to achieving this is initially loading the actuator, enabling it to deliver the required sudden input. Additionally, numerical investigations were conducted with torque saturation constraints, which showed that while this limitation does reduce the ROA, the observed ROA remains larger than that of a nominal linear controller without the OPFL component. Notably, the proposed control methodology could also serve as an alternative to the commonly used LQR method during the transition to the stabilization phase after employing a dedicated swing-up controller.

9.2 Limitations and future research directions

One obvious limitation in the work discussed in this dissertation is that all the controllers were designed based on dynamics models, which were developed assuming the soft robot to be PCC. While the PCC assumption holds for the systems considered here, especially for the planar cases without loading, it may fail upon significant external loading or self-weight. These limitations open up an intriguing opportunity to develop dynamics models for soft robots, relaxing the PCC hypothesis. Several researchers have already started exploring this avenue by proposing affine curvature models [85], polynomial curvature models [34], models based on Euler curves [140] to name a few. However, it is not well understood what level of fidelity should the dynamics model capture for it to be computationally amenable yet accurate enough to generate repeatable, precise motion control of soft robots. Moreover, due to the limitation of actuation, all the degrees of freedom/ curvature modes of these higher-order models can't be actuated. Therefore, a potential future research direction from this dissertation is investigating robust stabilization of higher-order highly underactuated soft robots.

Considering the utilization of integrated sensing for the control of soft robots, as investigated in Chapter 3, the current technologies only provide sensing for the simplified PCC model. However, to utilize integrated sensing in higher-order models, the soft sensors need to be able to measure the curvature at a higher degree, such as a polynomial curve or an Euler curve. To that end, another future research direction stemming from this dissertation is investigating a distributed sensing approach with a learning-based curvature estimation that could enhance the curvature measurements.

Considering the swing-up control of SIPR, as discussed in Chapter 5, the experimental validation was only limited to a simulation study. While the simulation results were exceptional for the assumed CC model, challenges were faced when experimentally evaluating the proposed controller in a physical system. Actuator saturation was observed, i.e., the actuation unit cannot provide the required torques due to the limits in the actuation unit. Further, the revolute joint at the base is not frictionless as assumed in the control derivation. Moreover, the CC assumption might fail due to torques induced by self-weight. The system parameters, which were found by system identification, might be uncertain. These limitations prompt exciting future research directions. Given the actuator saturation problem, one possible future direction is to pursue an alternate energy-based control approach for soft robots modeled as under-actuated systems inspired by the works in [111], which employs a partial feedback linearization step followed by passivity-based switching control that includes actuator saturation in the control design. However, the key here is to note that for the partial feedback linearization step [138], dynamic and kinematic uncertainty can destabilize the overall system. Hence, it is also essential to investigate robust and adaptive control of under-actuated systems to address dynamic and kinematic uncertainty in the system. To that end, a control strategy such as the adaptive task-space regulation of rigid-link flexible-joint robots with uncertain kinematics in [141] can be used to gain insight into developing an improved controller.

Revisiting the physical experimental investigations discussed in this dissertation, actuation delays, actuator saturation, and hysteresis effects were noted. These effects were overcome in this dissertation by carefully tuning a lower-order controller for the pneumatic actuator unit. However, this approach limits the motions of the soft robots; thus, the full potential of the manipulation capabilities can't be achieved. Therefore, future research addressing this limitation in actuation is paramount to realize the dexterous motions enabled by the designed controllers for the soft robotic systems.

Appendix A: Preliminaries and Supplementary Material

A.1 Stability

In this section the definitions are provided for the stability notions [74, 139] used in this dissertation .

Definition A.1 *Given the nonlinear system $\dot{x}(t) = f(x(t))$, suppose that $x = 0$ is an equilibrium. Then the solution $x(t) = 0$ is:*

- ***stable**: if and only if, for any $epsilon > 0$ there exist $\delta = \delta(\epsilon) > 0$ such that $||x(t_0)|| < \delta$ implies $||x(t)|| < \epsilon$ for all $t > t_0$.*
- ***asymptotically stable**: if $x = 0$ is stable and, in addition, $||x(t_0)|| < \delta$ implies $||x(t)|| \to 0$ as $t \to \infty$.*
- ***unstable**: if it is not stable.*

Definition A.2 *A solution $x(t)$ for the nonlinear system $\dot{x}(t) = f(x(t))$ is said to be **uniformly ultimately bounded** (u.u.b), if there exist constants $a, b > 0$ and a time $T = T(a, b)$ such that $||x(t_0)|| \leq a$ implies that $||x(t_0)|| \leq b$, for all $t \geq t_0 + T$. In this case the ultimate bound is b.*

Definition A.3 *The equilibrium $x = 0$ of the system $\dot{x}(t) = f(x(t))$ is **exponentially stable** if there are constants $\alpha, \gamma > 0$ such that $||x(t)|| \leq \alpha||x(t_0)||e^{-\gamma t}$ for all $t > 0$.*

A.2 Lagrangian systems

Important properties of the Lagrangian systems that has been utilized in the dissertation is discussed here. Consider the robot dynamics expressed in the general Lagrangian form as,

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + N(q,\dot{q}) = \tau \tag{A.1}$$

where $M(q)$ is the inertia matrix, $C(q,\dot{q})$ is the Coriolis/centripetal torque matrix and τ is the generalized force vector. $N(q,\dot{q})$ is the vector that collects gravity terms and other dynamic effects including the stiffness and damping terms.

The robot dynamics described as Lagrangian systems (A.1) posses the following fundamental properties [74, 142]:

Property A.1 *$M(q)$ is symmetric and positive definite and bounded- i.e there exist positive constants λ_a and λ_b such that $\lambda_a \leqslant M(q) \leqslant \lambda_b$*

Property A.2 *Under an appropriate definition of $C(q,\dot{q})\dot{q}$, such as defined using Christoffel symbols, the matrix $\dot{M}(q) - 2C(q,\dot{q})\dot{q}$ is skew symmetric.*

Property A.3 *For any differentiable vector $\gamma \in R^{\alpha}$ the Lagrangian dynamics are linearly parameterizable. Therefore there exists a regressor $Y(q,\dot{q},\gamma,\dot{\gamma}) \in R^{\alpha\times\beta}$ and a constant parameter vector $\Theta \in R^{\beta}$ such that $M(q)\dot{\gamma} + C(q,\dot{q})\gamma + N(q,\dot{q}) = Y(q,\dot{q},\gamma,\dot{\gamma})\Theta$*

Property A.4 *For $q,\dot{q},\gamma \in R^{n}$ there exists $k_c \in R^{+}$ such that the centrifugal and Coriolis forces/torques are bounded by $|C(q,\dot{q})\gamma| \leqslant k_c|\dot{q}||\gamma|$.*

A.3 2 DoF planar rigid elbow manipulator

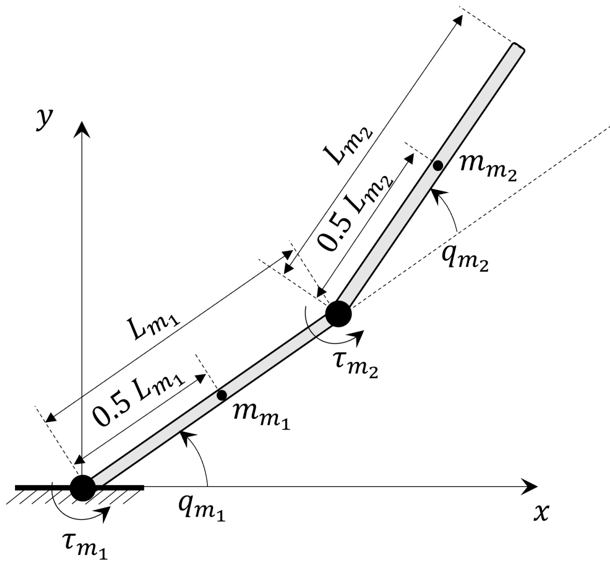


Figure A.1: The 2 DoF planar rigid elbow manipulator

Here the dynamics of the 2-DOF rigid robot used in the simulation study and in the experimental investigation of the bilateral teleoperation framework in Chapter 2 is discussed. It is assumed that the robot is on the horizontal plane, thus ignoring the gravity terms. The dynamics of the planar rigid manipulator in the Lagrangian form is,

$$\begin{bmatrix} M_{m_{11}} & M_{m_{12}} \\ M_{m_{21}} & M_{m_{22}} \end{bmatrix} \begin{bmatrix} \ddot{q}_{m_1} \\ \ddot{q}_{m_2} \end{bmatrix} + \begin{bmatrix} C_{m_{11}} & C_{m_{12}} \\ C_{m_{21}} & C_{m_{22}} \end{bmatrix} \begin{bmatrix} \dot{q}_{m_1} \\ \dot{q}_{m_2} \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} - J^\top F_{ext}$$

with the inertia matrix terms,

$$
\begin{aligned}
M_{m_{11}} &= 0.25 m_{m_1} L_{m_1}^2 + m_{m_2}\left(L_{m_1}^2 + 0.25 L_{m_2}^2 + L_{m_1} L_{m_2}\cos q_{m_2}\right) + I_{m_1} + I_{m_2} \\
M_{m_{12}} &= M_{m_{21}} = m_{m_2}\left(0.25 L_{m_2}^2 + L_{m_1} L_{m_2}\cos q_{m_2}\right) + I_{m_2} \\
M_{m_{22}} &= 0.25 m_{m_2} L_{m_2}^2 + I_{m_2}
\end{aligned}
$$

and the Coriolis /centripetal terms

$$
\begin{aligned}
C_{m_{11}} &= -0.5 m_{m_2} L_{m_1} L_{m_2} \sin q_{m_2} \dot{q_{m_2}} \\
C_{m_{12}} &= -0.5 m_{m_2} L_{m_1} L_{m_2} \sin q_{m_2} \left(\dot{q}_{m_1} + \dot{q}_{m_2}\right) \\
M_{m_{21}} &= 0.5 m_{m_2} L_{m_1} L_{m_2} \sin q_{m_2} \dot{q}_{m_1} \\
C_{m_{22}} &= 0
\end{aligned}
$$

Here, $J^\top$ is the Jacobian defined as,

$$
J^\top = \begin{bmatrix} -L_{m_1}\sin q_{m_1} - L_{m_2}\sin(q_{m_1}+q_{m_2}) & -L_{m_2}\sin(q_{m_1}+q_{m_2}) \\ L_{m_1}\cos q_{m_1} + L_{m_2}\cos(q_{m_1}+q_{m_2}) & L_{m_2}\cos(q_{m_1}+q_{m_2}) \end{bmatrix}
$$

and F_{ext} is the external force (human force).

Following the textbook [74], the inertia terms of the rigid robot is found as,

$$\Theta_1 = 0.25 m_{m_1} L_{m_1}^2 + m_{m_2}\left(L_{m_1}^2 + 0.25 L_{m_2}^2\right) + I_{m_1} + I_{m_2}$$

$$\Theta_2 = 0.5 m_{m_2} L_{m_1} L_{m_2}$$

$$\Theta_3 = 0.25 m_{m_2} L_{m_2}^2 + I_{m_2}$$

Now, using the linearity in parameters property (*Property A.3*), one can write the dynamics as,

$$Y(q_m, \dot{q}_m, \ddot{q}_m)\Theta = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix} - J^\top F_{ext}$$

where

$$Y(q_m, \dot{q}_m, \ddot{q}_m) = \begin{bmatrix} \ddot{q}_{m_1} & \cos(q_{m_2})(2\ddot{q}_{m_1} + \ddot{q}_{m_2}) - \sin(q_{m_2})(\dot{q}_{m_1}^2 + 2\dot{q}_{m_1}\dot{q}_{m_2}) & \ddot{q}_{m_2} \\ 0 & \cos(q_{m_2})\ddot{q}_{m_2} + \sin(q_{m_2})\dot{q}_{m_1}^2 & \ddot{q}_{m_1} + \ddot{q}_{m_2} \end{bmatrix}$$

and $\Theta = [\Theta_1, \Theta_2, \Theta_3]^\top$.

A.4 Sub-task control in null space

A given robot is redundant, if the robot's degrees of freedom (α) is greater than the dimension of the task space p. Thus, $null(J)$ has a minimum dimension of $(\alpha - p)$ which can be exploited to accomplish a desired sub-task control as the task space motion is not affected of the link velocity in the null space. This is done by designing an auxiliary function $\psi_s(t)$ appropriately. The sub-task error is defined as the null space velocity tracking error of the auxiliary function $\psi_s(t)$ [143],

$$e_N(t) = (\mathbb{I}_\alpha - J^+ J)(\dot{q} - \psi_s(t)).$$

It is observed that when the signal $s(t)$ (in (2.4), (6.3)) is projected on to the null space of J,

$$\begin{aligned}(\mathbb{I}_\alpha - J^+ J)s(t) =& (\mathbb{I}_\alpha - J^+ J)J^+ \Lambda e + (\mathbb{I}_\alpha - J^+ J)\dot{q} - (\mathbb{I}_\alpha - J^+ J)(\mathbb{I}_\alpha - J^+ J)\psi_s \\ =& (\mathbb{I}_\alpha - J^+ J)(\dot{q} - \psi_s(t)) \\ =& e_N(t).\end{aligned}$$

Here the properties $(\mathbb{I}_\alpha - J^+ J)J^+ = 0$ and $(\mathbb{I}_\alpha - J^+ J)(I_\alpha - J^+ J) = (\mathbb{I}_\alpha - J^+ J)$ of the pseudo inverse J^+ are utilized. Thus, observe that as $s \to 0$, $e_N \to 0$.

The auxiliary function can be designed in such a way that taking the negative gradient of a convex function $f(q_s)$ as,

$$\psi_s = -\frac{\partial}{\partial q} f(q), \tag{A.2}$$

whose minima leads to the desired state. When there are multiple sub-tasks, the auxiliary function is the summation of the negative gradients.

This dissertation considers the sub-tasks of achieving collision avoidance of the soft/HyRiSo robot and accounting for joint/curvature limits.

A.4.1 Collision avoidance

In this subtask, consider the location of an obstacle in the environment to be X_0. To avoid points $X_{s_j}, j \in \Omega$ on the robot (Ω is the set of points on the robot designed for collision avoidance) colliding with the obstacle, the convex function for the collision avoidance sub-task is defined as,

$$f^0_{\text{obs}_j}(q) = \left(\min\left\{0, \frac{d^2_{j0} - R^2}{d^2_{j0} - r^2}\right\}\right)^2$$

where $d_{j0} = ||X_{s_j} - X_0||$ is the distance between a point X_{s_j} on the robot and the obstacle X_0. Here R is the avoidance distance and r is the smallest safe distance of d_{j0}. The objective of this avoidance function is to guarantee that d_{j0} remains greater than the safe distance r by changing the configuration of the robot in the null space. Now, the auxiliary function is,

$$\psi^0_{s_j} = -\begin{bmatrix} \frac{\partial f^0_{\text{obs}_j}(q)}{\partial q_1} & \frac{\partial f^0_{\text{obs}_j}(q)}{\partial q_2} & \dots & \frac{\partial f^0_{\text{obs}_j}(q)}{\partial q_\alpha} \end{bmatrix}^\top \tag{A.3}$$

with

$$
\frac{\partial f^0_{\text{obs}_j}(q)}{\partial q_i} = \frac{\partial f^0_{\text{obs}_j}(q)}{\partial X_{s_j}} \frac{\partial X_{s_j}}{\partial q_i} \qquad (\text{for} \quad i = 1, 2, ..., \alpha)
$$

$$
= \begin{cases} 0 \quad \text{if} \quad d_{j0} \geq R \\ 4\left[\frac{(R^2-r^2)(d_{j0}^2-R^2)}{(d_{j0}^2-r^2)^3}\right]\left(X_{s_j} - X_0\right)^\top \frac{\partial X_{s_j}}{\partial q_i} \\ \qquad \text{if} \quad r < d_{j0} < R \\ \text{not defined} \quad \text{if} \quad d_{k0} = r \\ 0 \quad \text{if} \quad d_{k0} < r. \end{cases} \tag{A.4}
$$

In different applications there could be multiple points on the robot for collision avoidance (X_{s_j}, $j \in \Omega$) as well as multiple objects ($X_k,\ k = 1, 2, .., m$) in the environment. In that case, the auxiliary function for collision avoidance is the summation, $\psi_s = \sum_{k=1}^{m} \sum_{j\in\Omega} \psi^k_{s_j}$.

A.4.2 Joint angle/curvature limits

For the joint angle/curvature limit sub-task the convex function is defined as,

$$
f_{\text{joint}}(q) = \Pi_{j=1}^{\alpha} \left(\left(\frac{1}{q_j^{max} - q_j} \right) \left(\frac{1}{q_j - q_j^{min}} \right) \right)
$$

where q_j is the joint angle/degree of curvature of the j^{th} link with $j = 1, 2, ..., \alpha$. Here q_j^{max} and q_j^{min} denote the maximum and minimum allowable joint/degree of curvature limits of the j^{th} link. The corresponding auxiliary function for this sub-task can be easily found using (A.2).

A.5 HyRiSo robot example

In this appendix the details of the hybrid robot considered in the Example 6.1 is discussed briefly. Using the usual definitions, the dynamics of the robot in the form of (6.1) is given by,

$$\begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} + \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & d \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & k \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix}$$

with,

$$\begin{aligned} M_{11} &= m_1 \phi_1^M(L,q) + m_2 \phi_2^M(L,q) + I_1 \\ M_{21} &= M_{12} = m_2 \phi_3^M(L,q) \\ M_{22} &= m_2 \phi_4^M(L,q) \end{aligned}$$

where,

$$\begin{aligned} \phi_1^M(q,L) &= \frac{1}{4\,{q_2}^2} \left((L_2 \sin(q_1+q_2) - L_2 \sin(q_1) + 2\,L_1\,q_2 \cos(q_1))^2 + {L_2}^2 (\cos(q_1) - \cos(q_1+q_2) + 2\,q_2 \sin(q_1))^2 \right) \\ \phi_2^M(q,L) &= L_1^2 \\ \phi_3^M(q,L) &= \frac{L_2 \sin\left(\frac{q_2}{2}\right) \left(2L_2 \sin\left(\frac{q_2}{2}\right) + L_1\,q_2 \cos\left(\frac{q_2}{2}\right) + L_2\,q_2 \cos\left(\frac{q_2}{2}\right) + L_1\,q_2 \cos\left(2\,q_1 + \frac{q_2}{2}\right) - L_2\,q_2 \cos\left(2\,q_1 + \frac{q_2}{2}\right)\right)}{4\,{q_2}^2} \\ &\quad - \frac{L_2 \left(2 \sin\left(\frac{q_2}{2}\right) - q_2 \cos\left(\frac{q_2}{2}\right)\right) \left(L_1 \sin\left(q_1 + \frac{q_2}{2}\right) \cos(q_1) - L_2 \cos\left(q_1 + \frac{q_2}{2}\right) \sin(q_1)\right)}{2\,{q_2}^2} \\ \phi_4^M(q,L) &= -\frac{{L_2}^2 \left(2 \cos(q_2) + 2\,q_2 \sin(q_2) - {q_2}^2 - 2\right)}{4\,{q_2}^4}\,. \end{aligned}$$

The elements

$$
\begin{aligned}
C_{11} =& m_1\phi_1^C(L,q,\dot{q}) + m_2\phi_2^C(L,q,\dot{q}) \\
C_{12} =& m_2\phi_3^C(L,q,\dot{q}) \\
C_{21} =& m_2\phi_4^C(L,q,\dot{q}) \\
C_{22} =& m_2\phi_5^C(L,q,\dot{q})
\end{aligned}
$$

are defined using the Christoffel symbols. For $a_1, a_2 > 0$, let the transformation matrix,

$$A = \begin{bmatrix} a_1 & 0 \\ 0 & a_2 \end{bmatrix}.$$

Now the dynamics in the form of (6.2) is,

$$
\begin{bmatrix} \frac{M_{11}}{a_1} & \frac{M_{12}}{a_1} \\ \frac{M_{21}}{a_2} & \frac{M_{22}}{a_2} \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} + \begin{bmatrix} \frac{C_{11}}{a_1} & \frac{C_{12}}{a_1} \\ \frac{C_{21}}{a_2} & \frac{C_{22}}{a_2} \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix}
$$
$$
+ \begin{bmatrix} 0 & 0 \\ 0 & \frac{d}{a_2} \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} + \begin{bmatrix} 0 & 0 \\ 0 & \frac{k}{a_2} \end{bmatrix} \begin{bmatrix} q_1 \\ q_2 \end{bmatrix} = \begin{bmatrix} p_1 \\ p_2 \end{bmatrix}.
$$

A.6 Soft inverted pendulum dynamics

In this appendix, the explicit expressions for the inertia matrix $\mathbf{M}(q_0, \theta)$, Corriolis and centrifugal terms matrix $\mathbf{C}(q_0, \theta, \dot{q}_0, \dot{\theta})$ and gravity terms vector $\mathbf{G}(q_0, \theta)$ are provided in Section 7.2. Also, for what concerns to be division by q_0, the limiting case $q_0 \to 0$ for these expressions is provided explicitly.

Mass matrix:

$$\mathbf{M}(q_0, \theta) = \begin{pmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{pmatrix}$$

where

$$M_{11} = \frac{m\left(72\,L^2\,q_0 + D^2\,{q_0}^5 + 12\,L^2\,{q_0}^3 - 144\,L^2\,\sin\left(q_0\right) + 72\,L^2\,q_0\,\cos\left(q_0\right)\right)}{36\,{q_0}^5}$$
$$M_{12} = M_{21} = \frac{m\left(24\,L^2 + D^2\,{q_0}^4 + 12\,L^2\,{q_0}^2 - 24\,L^2\,\cos\left(q_0\right) - 24\,L^2\,q_0\,\sin\left(q_0\right)\right)}{24\,{q_0}^4}$$
$$M_{22} = \frac{m\left(24\,L^2\,q_0 + D^2\,{q_0}^3 - 24\,L^2\,\sin\left(q_0\right)\right)}{12\,{q_0}^3}$$

with the limiting case,

$$\lim_{q_0 \to 0} \mathbf{M} = \begin{pmatrix} \frac{m\left(D^2 + \frac{9\,L^2}{5}\right)}{36} & \frac{m\left(D^2 + 3\,L^2\right)}{24} \\ \frac{m\left(D^2 + 3\,L^2\right)}{24} & \frac{m\left(D^2 + 4\,L^2\right)}{12} \end{pmatrix}.$$

Corriolis and centrifugal terms matrix:

$$\mathbf{C}(q_0, \theta, \dot{q}_0, \dot{\theta}) = \begin{pmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{pmatrix}$$

where

$$C_{11} = -\frac{mL^2\dot{q}_0\,\left(12\,q_0 - 30\,\sin\left(q_0\right) + 3\,{q_0}^2\,\sin\left(q_0\right) + 18\,q_0\,\cos\left(q_0\right) + {q_0}^3\right)}{3\,{q_0}^6}$$
$$C_{12} = \frac{mL^2\dot{\theta}\,\left(2\,q_0 - 3\,\sin\left(q_0\right) + q_0\,\cos\left(q_0\right)\right)}{{q_0}^4}$$
$$C_{21} = -\frac{mL^2\left(4\,\dot{q}_0 + {q_0}^2\,\dot{q}_0 + 2\,{q_0}^2\,\dot{\theta} - 4\,\dot{q}_0\,\cos(q_0) - 4\,q_0\,\dot{q}_0\,\sin(q_0) - 3\,q_0\,\dot{\theta}\,\sin(q_0) + {q_0}^2\,\dot{q}_0\,\cos(q_0) + {q_0}^2\,\dot{\theta}\,\cos(q_0)\right)}{{q_0}^5}$$
$$C_{22} = -\frac{mL^2\dot{q}_0\,\left(2\,q_0 - 3\,\sin\left(q_0\right) + q_0\,\cos\left(q_0\right)\right)}{{q_0}^4}$$

with the limiting case,

$$\lim_{q_0 \to 0} \mathbf{C} = \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}.$$

Gravity terms vector,

$$\mathbf{G}(q_0, \theta) = L\,g\,m\begin{pmatrix} \frac{\left(2\,\cos(\phi+q_0+\theta) - 2\,\cos(\phi+\theta) + q_0\,\sin(\phi+\theta) + q_0\,\sin(\phi+q_0+\theta)\right)}{{q_0}^3} \\ -\frac{\left(\sin(\phi+\theta) - \sin(\phi+q_0+\theta) + q_0\,\cos(\phi+\theta)\right)}{{q_0}^2} \end{pmatrix}$$

with the limiting case,

$$\lim_{q_0 \to 0} \mathbf{G} = \begin{pmatrix} -\frac{L\,g\,m\,\sin(\phi+\theta)}{6} \\ -\frac{L\,g\,m\,\sin(\phi+\theta)}{2} \end{pmatrix}.$$

A.7 Linear Quadratic Regulator (LQR) for SIPR

The LQR to stabilize the SIPR in the upright equilibrium is discussed here. Let the state vector, $\mathbf{x}(\mathbf{t}) = [q_0(t), \theta(t), \dot{q}_0(t), \dot{\theta}(t)]^\top$ and control input $u(t) = \tau(t)$. Then linearizing around the upright equilibrium $(q_o, \theta, \dot{q}_0, \dot{\theta}) = (0, 0, 0, 0)$, $\tau = 0$ one gets the linear time invariant system,

$$\dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(\mathbf{t}) + \mathbf{B}u(t)$$

where $\mathbf{A} \in \mathbb{R}^{4\times4}$ is the system matrix and $\mathbf{B} \in \mathbb{R}^{4\times1}$ is the input matrix. Using $\mathbf{A}$ and $\mathbf{B}$ matrices one can show that this system is controllable.

For stabilization at the vertical upright position, the LQR problem is formulated as,

$$\min_{u} \quad \int_0^T \frac{1}{2}\left(\mathbf{x}^\top \mathbf{Q}\mathbf{x} + \mathbf{R}u^2\right) dt$$

to find the control $u(t)$ which is of the form,

$$u(t) = -(\mathbf{R}^{-1}\mathbf{B}^\top\mathbf{P})\mathbf{x}(t) = -\mathbf{K}\mathbf{x}(t).$$

Here $\mathbf{R} > 0$ and $\mathbf{P}$ is the solution to the algebraic Riccati equation,

$$\mathbf{A}^\top\mathbf{P} + \mathbf{P}\mathbf{A} - \mathbf{P}\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^\top\mathbf{P} + \mathbf{Q} = 0$$

where $\mathbf{Q}$ is a positive semidefinite matrix. In this work, $\mathbf{Q} = \mathbb{I}_{4\times4}$ and $\mathbf{R} = 1$ was chosen and the gain $\mathbf{K}$ was found using MATLAB command `lqr(A,B,Q,R)`.

A.8 Additional numerical examples for OPFL based swing up and stabilization of underactuated systems

In this appendix, numerical examples for utilizing the proposed control architecture for swing up and stabilization introduced in Chaper 8 is presented for two other *Class-I underactuated systems*, 1) RWP system, and 2) Acrobot.

A.8.1 The Reaction Wheel Pendulum

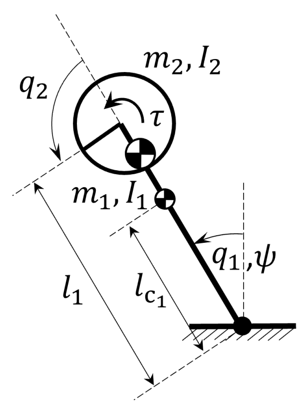


Figure A.2: The Reaction Wheel Pendulum (RWP).

The dynamics of the RWP as in FigureA.2 can be written as,

$$m_{11}\ddot{q}_1 + m_{12}\ddot{q}_2 + g_1(q_1) = 0$$

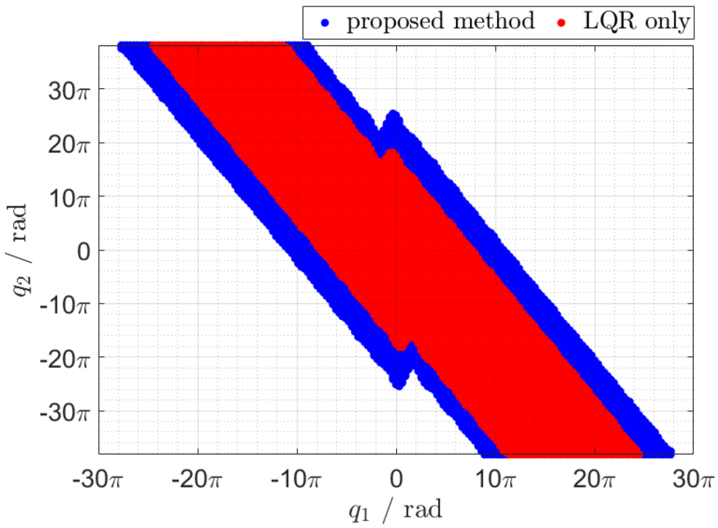
$$m_{21}\ddot{q}_1 + m_{22}\ddot{q}_2 = \tau$$

where, $m_{11} = m_1 l_{c1}^2 + I_1 + I_2$, $m_{12} = m_{21} = m_{22} = I_2$ and $g_1(q_1) = -(m_1 l_{c1} + m_2 l_2)\, g \sin(q_1)$. The output is defined as $y(t) = \psi(t) = q_1(t)$ considering the COM angle for the RWP. The Jacobians in this case will be $J_1 = 1, J_2 = 0$. The parameters were chosen as $m_1 = m_2 = 1$,

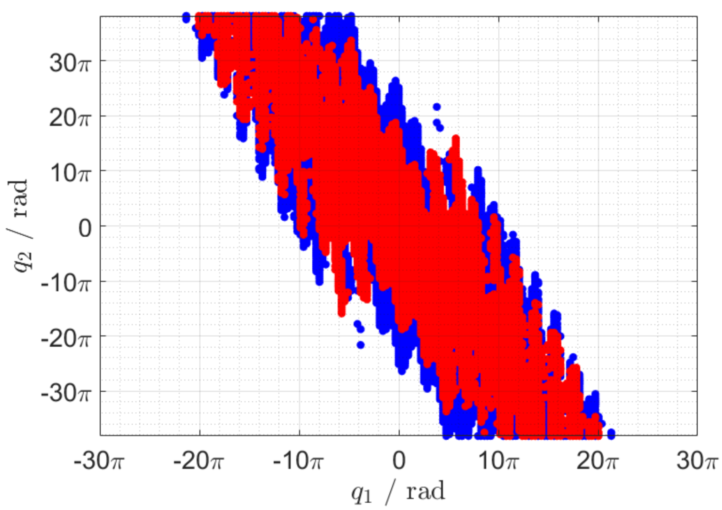
$I_1 = I_2 = 1$, $l_1 = 1$ and $l_{c_1} = 0.5$. Control gains $k_p = 250$ and $k_d = 100$ were used for the OPFL controller. For the LQR input, $\tau_q = \mathbf{K}\mathbf{x}$, as in (8.9) the gain $\mathbf{K} = [-433.71, -31.62, -171.26, -64.86]$ with $Q = \mathbb{I}_{4\times 4}$ and $R = 10^{-3}$ was calculated. For the nominal LQR $Q = \mathbb{I}_{4\times 4}$ and $R = 10^{-3}$ was used and obtained the gain $\mathbf{K}_{nom} = [-549.97, -31.62, -214.47, -57.45]$ implementing the control as $\tau = -\mathbf{K}_{nom}\mathbf{x}$ in (8.2).

For clarity in illustration, a slice of the ROA at $q_1 - q_2$ plane for the proposed method and the nominal LQR for trajectories starting from zero velocity positions is presented. Two cases were considered , i. when the applied torque τ is unrestricted (FigureA.3a) and ii. when the torque is saturated at 50Nm (FigureA.3b). While for practical initializations $q_1, q_2 \in [-2\pi, 2\pi]$ and $\dot{q}_1 = \dot{q}_2 = 0$, the ROAs are identical, it is seen that the proposed method has increased the ROA for the swing up and stabilization of the RWP in extreme cases. In the unrestricted torque case, the ROA increase is 45%, and in the saturated case, the ROA increase is 42.9% in the numerically analyzed range for the considered parameters.

FigureA.4a-c illustrates an unrestricted torque simulation for RWP starting from initial conditions $(-11\pi, 20\pi, 0, 0)$, which is inside both controllers' ROAs. In this case, the performance is similar for both the controllers. However, note the sudden high torque at the start of both controllers. The torque saturated simulation for the same is illustrated in FigureA.4d-e. Additionally a simulation when the system initialized at an extreme point $(-11\pi, 3\pi, 0, 0)$, which lies in the proposed method's ROA but outside the nominal LQR's ROA is also illustrated. The results are illustrated in FigureA.5a-c for the unrestricted case. It can be seen that the proposed method can swing up and stabilize the RWP at the origin while the nominal LQR diverges and stabilizes at a different equilibrium. The torque saturated simulation for this case using the proposed method is illustrated in FigureA.5d-e.

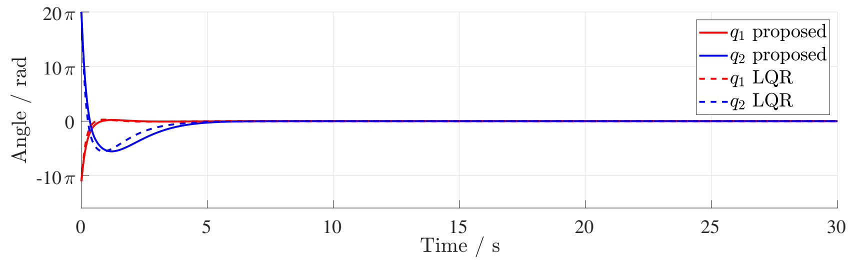


a) Unrestricted torque input.

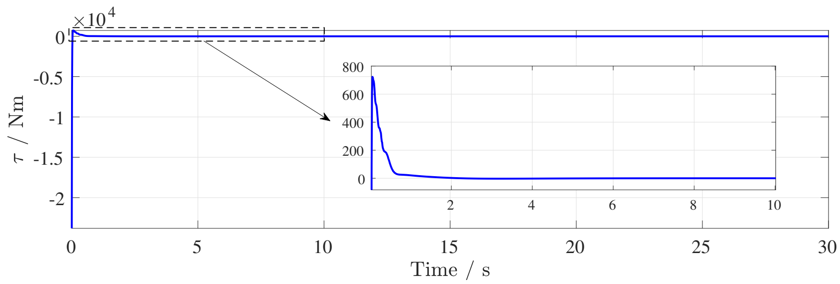


b) Saturated torque input at 50Nm.

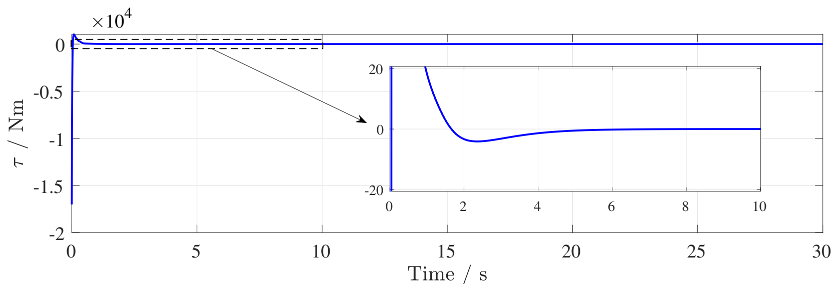
Figure A.3: ROA slice for RWP for $q_1, q_2 \in [-40\pi, 40\pi]$ when $\dot{q}_1 = \dot{q}_2 = 0$.



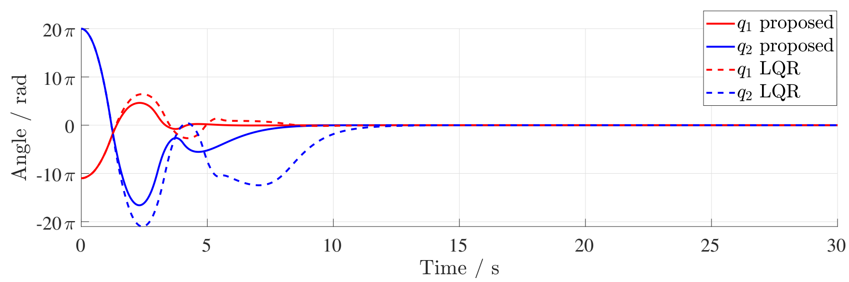
a) Evolution of angles with unrestricted input.



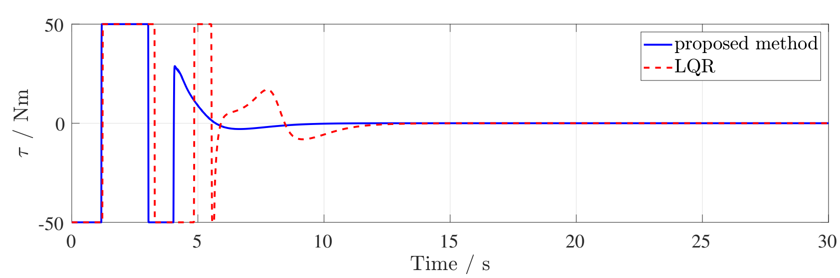
b) Torque evolution for the proposed method with unrestricted input.



c) Torque evolution for the nominal LQR with unrestricted input.

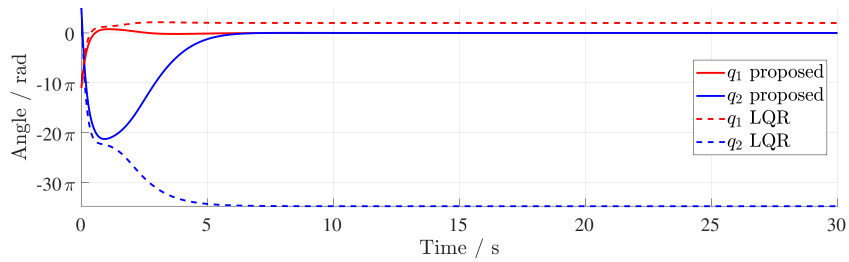


d) Evolution of angles with saturated input.

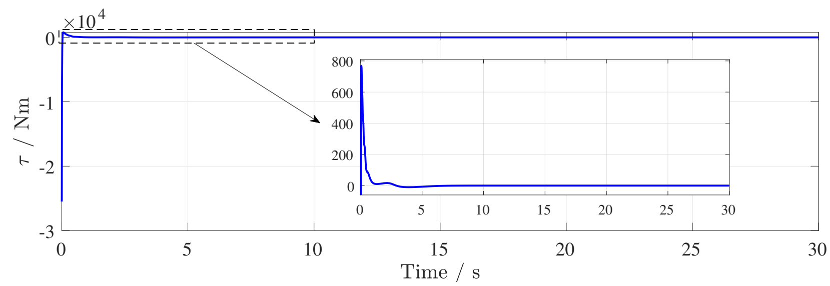


e) Torque evolution with saturated input.

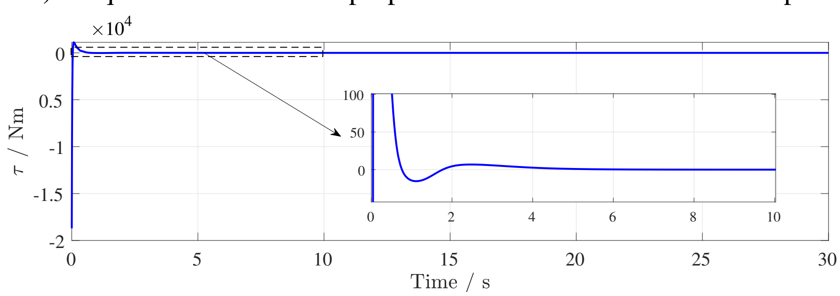
Figure A.4: Simulation results for RWP with initial conditions $(-11\pi, 20\pi, 0, 0)$.



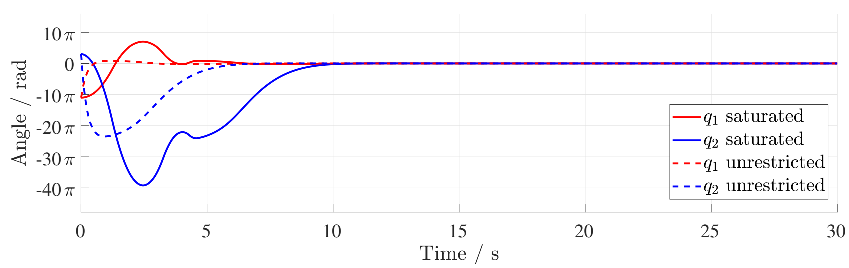
a) Evolution of angles with unrestricted input.



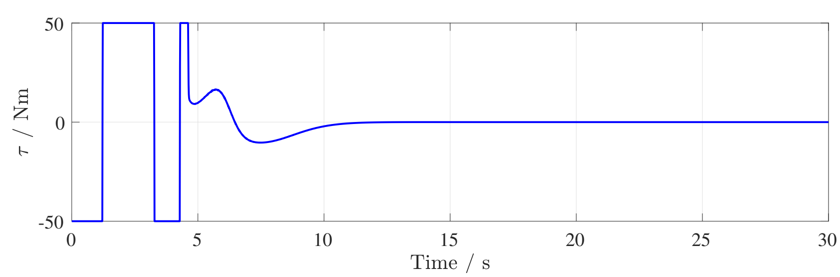
b) Torque evolution for the proposed method with unrestricted input.



c) Torque evolution for the nominal LQR with unrestricted input.



d) Evolution of angles for the proposed method with saturated input



e) Torque evolution for the proposed method with saturated input.

Figure A.5: Simulation results for RWP with the initial conditions $(-11\pi, 3\pi, 0, 0)$.

A.8.2 The Acrobot

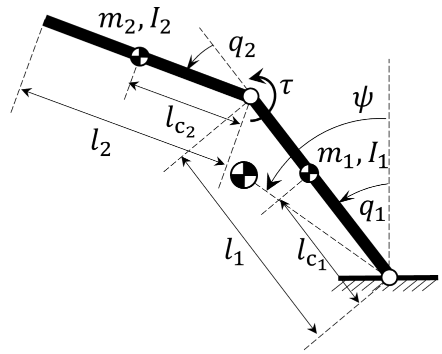


Figure A.6: The Acrobot.

The dynamics of the Acrobot as in FigureA.6 can be written as,

$$m_{11}\ddot{q}_1 + m_{12}\ddot{q}_2 + c_1 + g_1 = 0$$

$$m_{21}\ddot{q}_1 + m_{22}\ddot{q}_2 + c_2 + g_2 = \tau$$

with the mass terms,

$$m_{11} = m_1 l_{c1}^2 + m_2\left(l_1^2 + l_{c2}^2 + 2l_1 l_{c2}\cos q_2\right) + I_1 + I_2$$

$$m_{12} = m_{21} = m_2\left(l_{c2}^2 + l_1 l_{c2}\cos q_2\right) + I_2$$

$$m_{22} = m_2 l_{c2}^2 + I_2$$

centripetal and Coriolis terms,

$$c_1 = -m_2 l_1 l_{c2}\sin q_2\left(2\dot{q}_1\dot{q}_2 + \dot{q}_1^2\right), \qquad c_2 = m_2 l_1 l_{c2}\dot{q}_1^2$$

and gravity terms,

$$g_1 = -\left(m_1 l_{c1} + m_2 l_1\right) g \sin q_1 - m_2 l_{c2} g \sin\left(q_1 + q_2\right)$$

$$g_2 = -m_2 l_{c2} g \sin\left(q_1 + q_2\right).$$

The output function for the Acrobot is chosen as the COM angle from the vertical $y(t) = \psi(t)$. The COM $(\bar{x}, \bar{y})$ is found using composite parts method

$$\bar{x} = -\frac{m_1 l_{c1} \sin q_1 + m_2 \left(l_1 \sin q_1 + l_{c2} \sin(q_1 + q_2)\right)}{\left(m_1 + m_2\right)}$$
$$\bar{y} = \frac{m_1 l_{c1} \cos q_1 + m_2 \left(l_1 \cos q_1 + l_{c2} \cos(q_1 + q_2)\right)}{\left(m_1 + m_2\right)}$$

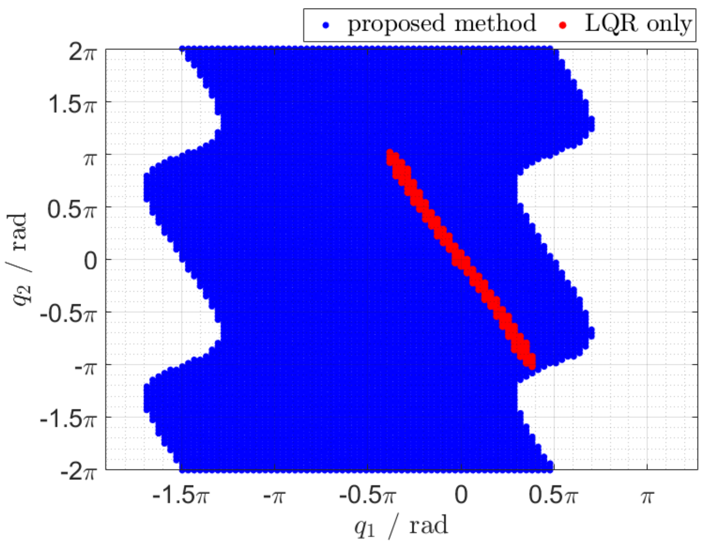
and the COM angle is found as $\psi(t) = \pi/2 - \text{atan}\left(\frac{\bar{y}}{\bar{x}}\right)$. Thus, with the defined output $y(q_1, q_2) = \psi$, the Jacobians are $J_1 = 1$ and

$$J_2 = \frac{m_2 l_{c2} \left(m_2 l_{c2} + m_2 l_1 \cos q_2 + m_1 l_{c1} \cos q2\right)}{\left(l_1 m_2 + l_{c1} m_1\right)^2 + 2 l_{c2} \left(l_1 m_2 + l_{c21} m_1\right) \cos q_2 + m_2^2 l_{c2}^2}$$

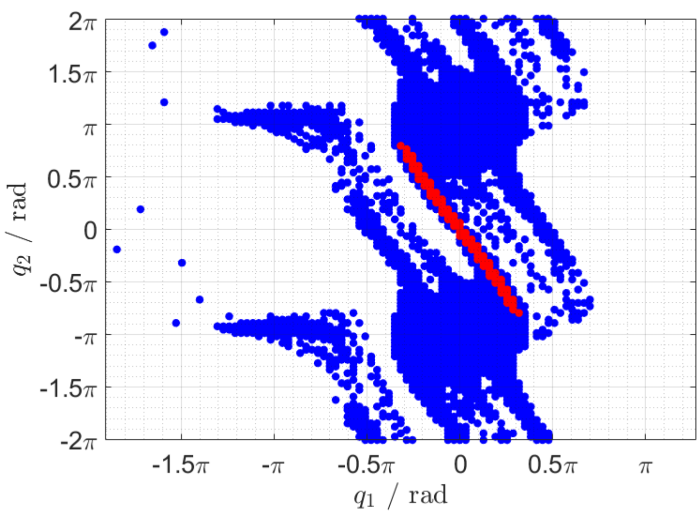
The parameters used were $m_1 = 1, m_2 = 2, l_1 = 1, l_2 = 1.5, l_{c1} = 0.5, l_{c2} = 0.75$, $I_1 = 0.2, I_2 = 1$ and $g = 9.81$. The control parameters $k_p = 2.5 \times 10^4$ and $k_d = 1 \times 10^3$ for the OPFL controller were chosen. For the LQR in (8.9), using $Q = \mathbb{I}_{4\times4}$ and $R = 1$e the LQR gain was calculated as, $\mathbf{K} = [-137.27, -52.55, -613.22, -293.44]$. For comparison, the nominal LQR gains were computed using $Q = \mathbb{I}_{4\times4}$ and $R = 10^{-3}$ and obtained the gain $\mathbf{K}_{nom} = [-399.53, -149.89, -173.75, -77.36]$.

The ROA slice at $q_1 - q_2$ plane is presented as done in the RWP example considering the initial conditions $q_1, q_2 \in [-2\pi, 2\pi]$ with $\dot{q}_1 = \dot{q}_2 = 0$. The ROA for the unrestricted torque case is illustrated in FigureA.7a, and the case with torque saturation at 50Nm is illustrated in FigureA.7b. It is observed that the proposed method significantly enlarges the ROA in the unrestricted case with an increase of over 50-fold (5151.3%). For the torque-saturated case, the ROA increase is almost 30-fold (2955.2%).

An unrestricted torque simulation is illustrated in Figure A.8a-c starting from initial conditions $(-\frac{\pi}{7}, \frac{3\pi}{8}, 0, 0)$ which is within ROAs of both the controllers. The torque-saturated simulation for the proposed method is illustrated in Figure A.8d-e. In Figure A.9, the simulation results for the Acrobot initialized from the downward equilibrium point $(-\pi, 0, 0, 0)$ with unrestricted torque is illustrated. Note that this point is within the ROA of the proposed method but outside that of the nominal LQR controller. However, this point is not within the ROA of the proposed method if torque is saturated at 50Nm. To illustrate the torque saturated behavior of the proposed method, a simulation initialized from $(-\frac{\pi}{2}, 0, 0, 0)$ is considered, which is within the ROA of the proposed method and it is illustrated in FigureA.10.

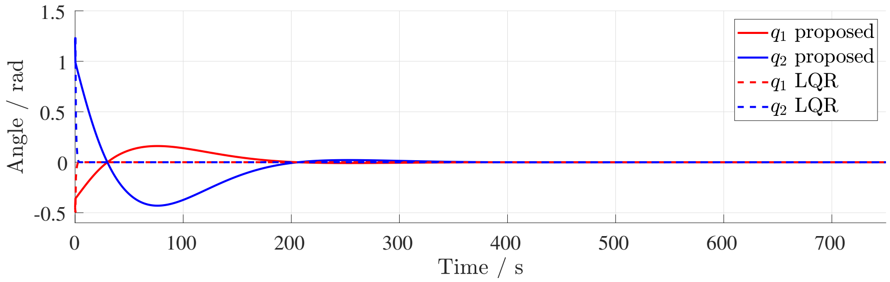


a) Unrestricted torque input

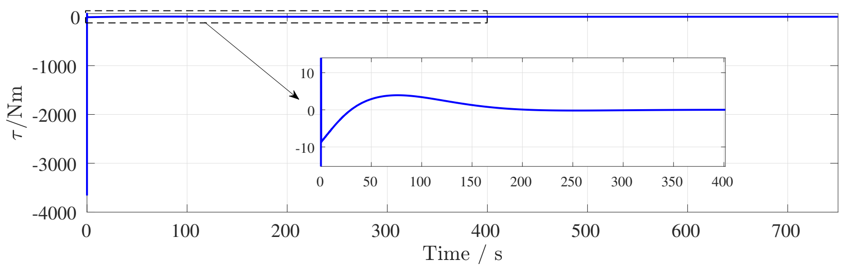


b) Saturated torque input at 50Nm

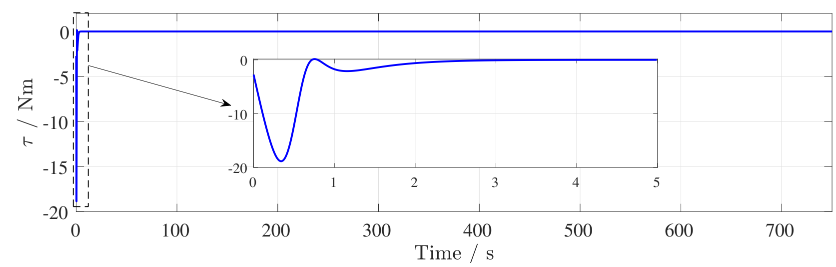
Figure A.7: ROA slice for Acrobot for $q_1, q_2 \in [-2\pi, 2\pi]$ with $\dot{q}_1 = \dot{q}_2 = 0$.



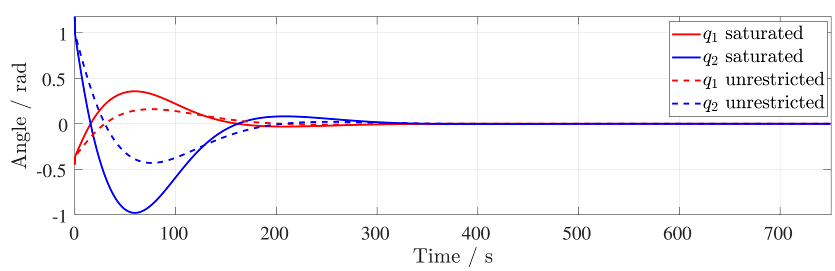
a) Evolution of angles with unrestricted input.



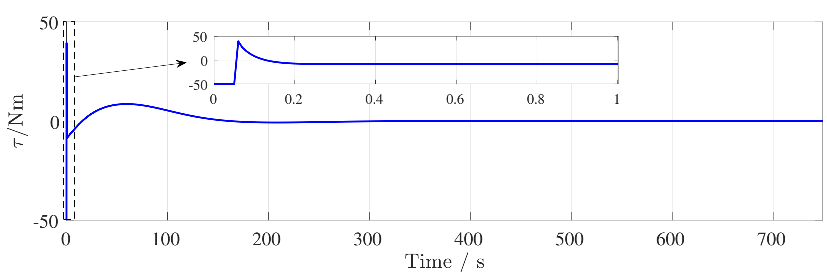
b) Torque Evolution for the proposed method with unrestricted input.



c) Torque Evolution for the nominal LQR with unrestricted torque input.

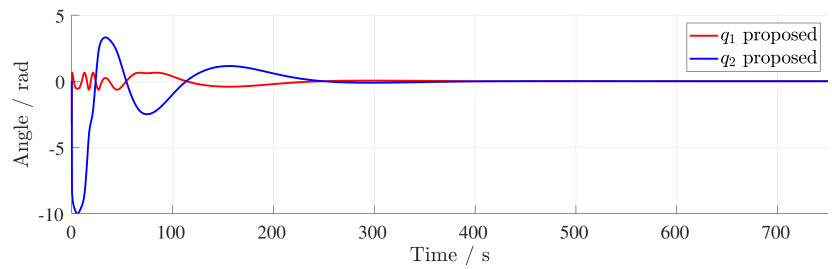


d) Evolution of angles with saturated torque input.

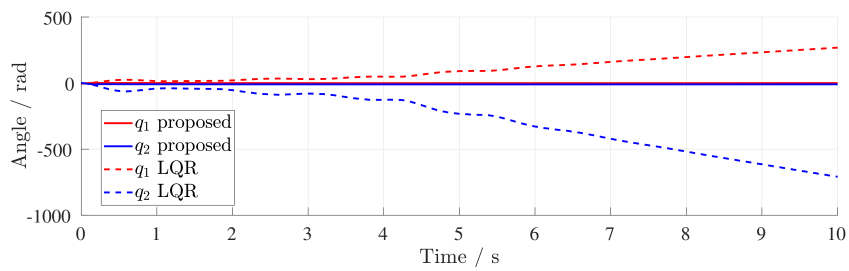


e) Torque Evolution for the proposed method with saturated torque input.

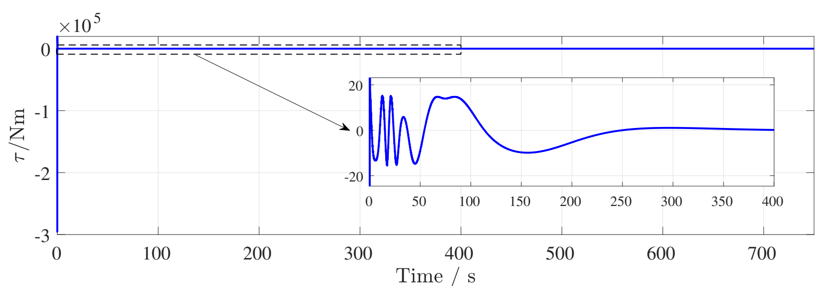
Figure A.8: Simulation results for Acrobot with initial conditions $(-\frac{\pi}{7}, \frac{3\pi}{8}, 0, 0)$ which is inside both ROAs.



a) Evolution of angles for the proposed method.

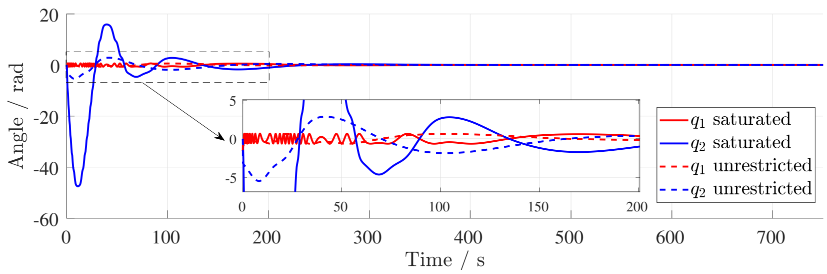


b) Evolution of angles for LQR showing unstable behavior.

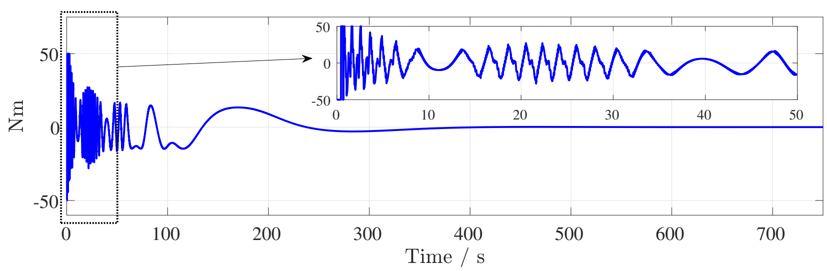


c) Torque evolution for the proposed method.

Figure A.9: Simulation results for Acrobot with unrestricted torque input for initial conditions $(-\pi, 0, 0, 0)$.



a) Evolution of angles for the proposed method.



b) Torque evolution for the proposed method with saturated input.

Figure A.10: Simulation results for Acrobot with initial conditions $(-\frac{\pi}{2}, 0, 0, 0)$ when torque is saturated at 50Nm.

Bibliography

[1] Oncay Yasa, Yasunori Toshimitsu, Mike Y Michelis, Lewis S Jones, Miriam Filippi, Thomas Buchner, and Robert K Katzschmann. An overview of soft robotics. *Annual Review of Control, Robotics, and Autonomous Systems*, 6:1–29, 2023.

[2] Daniela Rus and Michael T Tolley. Design, fabrication and control of soft robots. *Nature*, 521(7553):467–475, 2015.

[3] Cosimo Della Santina, Christian Duriez, and Daniela Rus. Model-based control of soft robots: A survey of the state of the art and open challenges. *IEEE Control Systems Magazine*, 43(3):30–65, 2023.

[4] Carmel Majidi. Soft robotics: a perspective—current trends and prospects for the future. *Soft Robotics*, 1(1):5–11, 2014.

[5] Thomas George Thuruthel, Yasmin Ansari, Egidio Falotico, and Cecilia Laschi. Control strategies for soft robotic manipulators: A survey. *Soft robotics*, 5(2):149–163, 2018.

[6] Cosimo Della Santina, Manuel G Catalano, and Antonio Bicchi. Soft robots. *Encyclopedia of Robotics, Springer*, 2020.

[7] Helmut Hauser. Soft robotics: An introduction to the special issue. *IEEE Control Systems Magazine*, 43(3):28–29, 2023.

[8] Matteo Cianchetti, Tommaso Ranzani, Giada Gerboni, Thrishantha Nanayakkara, Kaspar Althoefer, Prokar Dasgupta, and Arianna Menciassi. Soft robotics technologies to address shortcomings in today's minimally invasive surgery: the stiff-flop approach. *Soft robotics*, 1(2):122–131, 2014.

[9] Matteo Cianchetti, Tommaso Ranzani, Giada Gerboni, Iris De Falco, Cecilia Laschi, and Arianna Menciassi. Stiff-flop surgical manipulator: Mechanical design and experimental characterization of the single module. In *2013 IEEE/RSJ international conference on intelligent robots and systems*, pages 3576–3581. IEEE, 2013.

[10] Damien Chablat, Swaminath Venkateswaran, and Frédéric Boyer. Dynamic model of a bio-inspired robot for piping inspection. In *ROMANSY 22–Robot Design, Dynamics and Control*, pages 42–51. Springer, 2019.

[11] Jinglin Li, Zhou Teng, and Jing Xiao. Can a continuum manipulator fetch an object in an unknown cluttered space? *IEEE Robotics and Automation Letters*, 2(1):2–9, 2016.

[12] T TolleyMichael, F ShepherdRobert, C GallowayKevin, J WoodRobert, M WhitesidesGeorge, et al. A resilient, untethered soft robot. *Soft robotics*, 2014.

[13] Marcello Calisti, Giacomo Picardi, and Cecilia Laschi. Fundamentals of soft robot locomotion. *Journal of The Royal Society Interface*, 14(130):20170101, 2017.

[14] Naveen Kumar Uppalapati, Benjamin Walt, Aaron Havens, Armeen Mahdian, Girish Chowdhary, and Girish Krishnan. A berry picking robot with a hybrid soft-rigid arm: Design and task space control. *Proceedings of Robotics: Science and Systems, Corvalis, Oregon, USA*, page 95, 2020.

[15] Matteo Cianchetti. Embodied intelligence in soft robotics through hardware multifunctionality. *Frontiers in Robotics and AI*, 8:724056, 2021.

[16] III Robert J. Webster and Bryan A. Jones. Design and kinematic modeling of constant curvature continuum robots: A review. *The International Journal of Robotics Research*, 29(13):1661–1683, 2010.

[17] B. A. Jones and I. D. Walker. Kinematics for multisection continuum robots. *IEEE Transactions on Robotics*, 22(1):43–55, Feb 2006.

[18] Michael W. Hannan and Ian D. Walker. Kinematics and the implementation of an elephant's trunk manipulator and other continuum style robots. *Journal of robotic systems*, 20 2:45–63, 2003.

[19] Enver Tatlicioglu, Ian D Walker, and Darren M Dawson. Dynamic modelling for planar extensible continuum robot manipulators. In *Proceedings 2007 IEEE International Conference on Robotics and Automation*, pages 1357–1362. IEEE, 2007.

[20] Valentin Falkenhahn, Alexander Hildebrandt, Rüdiger Neumann, and Oliver Sawodny. Model-based feedforward position control of constant curvature continuum robots using feedback linearization. In *IEEE International Conference on Robotics and Automation*, pages 762–767, 2015.

[21] C. Della Santina, R. K. Katzschmann, A. Biechi, and D. Rus. Dynamic control of soft robots interacting with the environment. In *2018 IEEE International Conference on Soft Robotics*, pages 46–53, April 2018.

[22] Cosimo Della Santina, Robert K Katzschmann, Antonio Bicchi, and Daniela Rus. Model-based dynamic feedback control of a planar soft robot: trajectory tracking and interaction with the environment. *The International Journal of Robotics Research*, 39(4):490–513, 2020.

[23] Stuart S Antman. Problems in nonlinear elasticity. *Nonlinear Problems of Elasticity*, pages 513–584, 2005.

[24] Frédéric Boyer and Mathieu Porez. Multibody system dynamics for bio-inspired locomotion: from geometric structures to computational aspects. *Bioinspiration & biomimetics*, 10(2):025007, 2015.

[25] Federico Renda, Vito Cacucciolo, Jorge Dias, and Lakmal Seneviratne. Discrete cosserat approach for soft robot dynamics: A new piece-wise constant strain model with torsion and shears. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5495–5502. IEEE, 2016.

[26] Pierre E Dupont, Jesse Lock, and Brandon Itkowitz. Real-time position control of concentric tube robots. In *2010 IEEE International Conference on Robotics and Automation*, pages 562–568. IEEE, 2010.

[27] Deepak Trivedi, Christopher D Rahn, William M Kier, and Ian D Walker. Soft robotics: Biological inspiration, state of the art, and future research. *Applied bionics and biomechanics*, 5(3):99–117, 2008.

[28] Federico Renda, Frédéric Boyer, Jorge Dias, and Lakmal Seneviratne. Discrete cosserat approach for multisection soft manipulator dynamics. *IEEE Transactions on Robotics*, 34(6):1518–1533, 2018.

[29] Panagiotis Polygerinos, Zheng Wang, Johannes TB Overvelde, Kevin C Galloway, Robert J Wood, Katia Bertoldi, and Conor J Walsh. Modeling of soft fiber-reinforced bending actuators. *IEEE Transactions on Robotics*, 31(3):778–789, 2015.

[30] Frederick Largilliere, Valerian Verona, Eulalie Coevoet, Mario Sanz-Lopez, Jeremie Dequidt, and Christian Duriez. Real-time control of soft-robots using asynchronous finite element modeling. In *2015 IEEE International Conference on Robotics and Automation (ICRA)*, pages 2550–2555. IEEE, 2015.

[31] Maxime Thieffry, Alexandre Kruszewski, Olivier Goury, Thierry-Marie Guerra, and Christian Duriez. Dynamic control of soft robots. In *IFAC World congress*, 2017.

[32] SM Hadi Sadati, S Elnaz Naghibi, Ian D Walker, Kaspar Althoefer, and Thrishantha Nanayakkara. Control space reduction and real-time accurate modeling of continuum manipulators using ritz and ritz–galerkin methods. *IEEE Robotics and Automation Letters*, 3(1):328–335, 2017.

[33] Ryan S Penning and Michael R Zinn. A combined modal-joint space control approach for continuum manipulators. *Advanced Robotics*, 28(16):1091–1108, 2014.

[34] Cosimo Della Santina and Daniela Rus. Control oriented modeling of soft robots: the polynomial curvature case. *IEEE Robotics and Automation Letters*, 5(2):290–298, 2019.

[35] Cosimo Della Santina. The soft inverted pendulum with affine curvature. In *2020 59th IEEE Conference on Decision and Control (CDC)*, pages 4135–4142. IEEE, 2020.

[36] H. Wang, B. Yang, Y. Liu, W. Chen, X. Liang, and R. Pfeifer. Visual servoing of soft robot manipulator in constrained environments with an adaptive controller. *IEEE/ASME Transactions on Mechatronics*, 22(1):41–50, Feb 2017.

[37] Zhongkai Zhang, Jeremie Dequidt, Alexandre Kruszewski, Frederick Largilliere, and Christian Duriez. Kinematic modeling and observer based control of soft robot using real-time finite element method. In *2016 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 5509–5514. IEEE, 2016.

[38] Andrew D Marchese and Daniela Rus. Design, kinematics, and control of a soft spatial fluidic elastomer manipulator. *The International Journal of Robotics Research*, 35(7):840–869, 2016.

[39] Andrew D Marchese, Russ Tedrake, and Daniela Rus. Dynamics and trajectory optimization for a soft spatial fluidic elastomer manipulator. *The International Journal of Robotics Research*, 35(8):1000–1019, 2016.

[40] Robert K Katzschmann, Cosimo Della Santina, Yasunori Toshimitsu, Antonio Bicchi, and Daniela Rus. Dynamic motion control of multi-segment soft robots using piecewise constant curvature matched with an augmented rigid body model. In *2019 2nd IEEE International Conference on Soft Robotics (RoboSoft)*, pages 454–461. IEEE, 2019.

[41] Apoorva D Kapadia, Katelyn E Fry, and Ian D Walker. Empirical investigation of closed-loop control of extensible continuum manipulators. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 329–335. IEEE, 2014.

[42] Apoorva D Kapadia, Ian D Walker, Darren M Dawson, and Enver Tatlicioglu. A model-based sliding mode controller for extensible continuum robots. In *Proceedings of the 9th WSEAS international conference on Signal processing, robotics and automation*, pages 113–120. World Scientific and Engineering Academy and Society (WSEAS), 2010.

[43] Amirhossein Kazemipour, Oliver Fischer, Yasunori Toshimitsu, Ki Wan Wong, and Robert K Katzschmann. Adaptive dynamic sliding mode control of soft continuum manipulators. *arXiv preprint arXiv:2109.11388*, 2021.

[44] Maja Trumić, Cosimo Della Santina, Kosta Jovanović, and Adriano Fagiolini. Adaptive control of soft robots based on an enhanced 3d augmented rigid robot matching. In *2021 American Control Conference (ACC)*, pages 4991–4996. IEEE, 2021.

[45] Florian Hisch, Andrea Giusti, and Matthias Althoff. Robust control of continuum robots using interval arithmetic. *IFAC-PapersOnLine*, 50(1):5660–5665, 2017.

[46] Maja Trumić, Kosta Jovanović, and Adriano Fagiolini. Decoupled nonlinear adaptive control of position and stiffness for pneumatic soft robots. *The International Journal of Robotics Research*, 40(1):277–295, 2021.

[47] Hongbo Wang, Massimo Totaro, and Lucia Beccai. Toward perceptive soft robots: Progress and challenges. *Advanced Science*, 5(9):1800541, 2018.

[48] Giada Gerboni, Alessandro Diodato, Gastone Ciuti, Matteo Cianchetti, and Arianna Menciassi. Feedback control of soft robot actuators via commercial flex bend sensors. *IEEE/ASME Transactions on Mechatronics*, 22(4):1881–1888, 2017.

[49] Khaled Elgeneidy, Niels Lohse, and Michael Jackson. Bending angle prediction and control of soft pneumatic actuators with embedded flex sensors–a data-driven approach. *Mechatronics*, 50:234–247, 2018.

[50] Zhi Qiang Tang, Ho Lam Heung, Kai Yu Tong, and Zheng Li. A novel iterative learning model predictive control method for soft bending actuators. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 4004–4010. IEEE, 2019.

[51] Zhi Qiang Tang, Ho Lam Heung, Kai Yu Tong, and Li. Model-based online learning and adaptive control for a human-wearable soft robot integrated system. *The International Journal of Robotics Research*, 2019.

[52] Thomas George Thuruthel, Benjamin Shih, Cecilia Laschi, and Michael Thomas Tolley. Soft robot perception using embedded soft sensors and recurrent neural networks. *Science Robotics*, 4(26), 2019.

[53] Yang Yang and Yonghua Chen. Innovative design of embedded pressure and position sensors for soft actuators. *IEEE Robotics and Automation Letters*, 3(2):656–663, 2017.

[54] Michelle C Yuen, Rebecca Kramer-Bottiglio, and Jamie Paik. Strain sensor-embedded soft pneumatic actuators for extension and bending feedback. In *2018 IEEE International Conference on Soft Robotics (RoboSoft)*, pages 202–207. IEEE, 2018.

[55] Lucas O Tiziani, Thomas W Cahoon, and Frank L Hammond. Sensorized pneumatic muscle for force and stiffness control. In *2017 IEEE International Conference on Robotics and Automation (ICRA)*, pages 5545–5552. IEEE, 2017.

[56] Ali Zolfagharian, MA Parvez Mahmud, Saleh Gharaie, Mahdi Bodaghi, Abbas Z Kouzani, and Akif Kaynak. 3d/4d-printed bending-type soft pneumatic actuators: fabrication, modelling, and control. *Virtual and Physical Prototyping*, pages 1–30, 2020.

[57] Benjamin Shih, Dylan Drotman, Caleb Christianson, Zhaoyuan Huo, Ruffin White, Henrik I Christensen, and Michael T Tolley. Custom soft robotic gripper sensor skins for haptic object visualization. In *2017 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pages 494–501. IEEE, 2017.

[58] Ryan L Truby, Cosimo Della Santina, and Daniela Rus. Distributed proprioception of 3d configuration in soft, sensorized robots via deep learning. *IEEE Robotics and Automation Letters*, 5(2):3299–3306, 2020.

[59] Cosimo Della Santina, Ryan Landon Truby, and Daniela Rus. Data–driven disturbance observers for estimating external forces on soft robots. *IEEE Robotics and Automation Letters*, 5(4):5717–5724, 2020.

[60] Gregory S Chirikjian. Hyper-redundant manipulator dynamics: A continuum approximation. *Advanced Robotics*, 9(3):217–243, 1994.

[61] J. . E. Slotine and Li Weiping. Adaptive manipulator control: A case study. *IEEE Transactions on Automatic Control*, 33(11):995–1003, Nov 1988.

[62] E. H. Skorina, M. Luo, W. Tao, F. Chen, J. Fu, and C. D. Onal. Adapting to flexibility: Model reference adaptive control of soft bending actuators. *IEEE Robotics and Automation Letters*, 2(2):964–970, April 2017.

[63] Apoorva D. Kapadia, Ian D. Walker, Darren M. Dawson, and Enver Tatlicioglu. A model-based sliding mode controller for extensible continuum robots. In *Proceedings of the 9th WSEAS International Conference on Signal Processing, Robotics and Automation*, ISPRA'10, pages 113–120, Wisconsin, USA, 2010.

[64] M. Li, R. Kang, D. T. Branson, and J. S. Dai. Model-free control for continuum robots based on an adaptive kalman filter. *IEEE/ASME Transactions on Mechatronics*, 23(1):286–297, Feb 2018.

[65] Chase G Frazelle, Apoorva D Kapadia, Katelyn E Fry, and Ian D Walker. Teleoperation mappings from rigid link robots to their extensible continuum counterparts. In *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pages 4093–4100. IEEE, 2016.

[66] Yimeng Dong and Nikhil Chopra. Passivity-based bilateral tele-driving system with parametric uncertainty and communication delays. *IEEE Control Systems Letters*, 3(2):350–355, 2018.

[67] Yen-Chen Liu and Nikhil Chopra. Control of semi-autonomous teleoperation system with time delays. *Automatica*, 49(6):1553–1565, 2013.

[68] Yongqiang Ye, Ya-Jun Pan, and Trent Hilliard. Bilateral teleoperation with time-varying delay: A communication channel passification approach. *IEEE/ASME Transactions on Mechatronics*, 18(4):1431–1434, 2013.

[69] Nikhil Chopra, Mark W Spong, and Rogelio Lozano. Synchronization of bilateral teleoperators with time delay. *Automatica*, 44(8):2142–2148, 2008.

[70] Jessica Burgner, D Caleb Rucker, Hunter B Gilbert, Philip J Swaney, Paul T Russell, Kyle D Weaver, and Robert J Webster. A telerobotic system for transnasal surgery. *IEEE/ASME Transactions on Mechatronics*, 19(3):996–1006, 2013.

[71] Apoorva D Kapadia, Ian D Walker, and Enver Tatlicioglu. Teleoperation control of a redundant continuum manipulator using a non-redundant rigid-link master. In *2012 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3105–3110. IEEE, 2012.

[72] C. G. Frazelle, A. D. Kapadia, and I. D. Walker. A nonlinear control strategy for extensible continuum robots. In *IEEE International Conference on Robotics and Automation*, pages 7727–7734, 2018.

[73] L. Weerakoon and N. Chopra. Bilateral teleoperation of soft robots under piecewise constant curvature hypothesis: An experimental investigation. In *2020 American Control Conference (ACC)*, pages 2124–2129, 2020.

[74] Mark W Spong, Seth Hutchinson, and Mathukumalli Vidyasagar. *Robot modeling and control*. John Wiley & Sons, 2020.

[75] Andrew D. Marchese, Robert K. Katzschmann, and Daniela Rus. A recipe for soft fluidic elastomer robots. In *Soft robotics*, 2015.

[76] Fluidic control board. Last accessed on 03/16/2020.

[77] Amirhossein Kazemipour, Oliver Fischer, Yasunori Toshimitsu, Ki Wan Wong, and Robert K Katzschmann. Adaptive dynamic sliding mode control of soft continuum manipulators. In *2022 International Conference on Robotics and Automation (ICRA)*, pages 3259–3265. IEEE, 2022.

[78] Milad Azizkhani, Isuru S Godage, and Yue Chen. Dynamic control of soft robotic arm: A simulation study. *IEEE Robotics and Automation Letters*, 7(2):3584–3591, 2022.

[79] Romeo Ortega and Mark W Spong. Adaptive motion control of rigid robots: A tutorial. *Automatica*, 25(6):877–888, 1989.

[80] M.W. Spong. On the robust control of robot manipulators. *IEEE Transactions on Automatic Control*, 37(11):1782–1786, 1992.

[81] Chaouki Abdallah, Darren M Dawson, Peter Dorato, and Mohammad Jamshidi. Survey of robust control for rigid robots. *IEEE Control Systems Magazine*, 11(2):24–30, 1991.

[82] Isuru S Godage, Gustavo A Medrano-Cerda, David T Branson, Emanuele Guglielmino, and Darwin G Caldwell. Dynamics for variable length multisection continuum arms. *The International Journal of Robotics Research*, 35(6):695–722, 2016.

[83] Yen-Chen Liu, Phuong Nam Dao, and Kai Yong Zhao. On robust control of nonlinear teleoperators under dynamic uncertainties with variable time delays and without relative velocity. *IEEE Transactions on Industrial Informatics*, 16(2):1272–1280, 2020.

[84] Robert K. Katzschmann, Cosimo Della Santina, Yasunori Toshimitsu, Antonio Bicchi, and Daniela Rus. Dynamic motion control of multi-segment soft robots using piecewise constant curvature matched with an augmented rigid body model. In *2019 2nd IEEE International Conference on Soft Robotics (RoboSoft)*, pages 454–461, 2019.

[85] Francesco Stella, Qinghua Guan, Cosimo Della Santina, and Josie Hughes. Piecewise affine curvature model: a reduced-order model for soft robot-environment interaction beyond pcc. In *2023 IEEE International Conference on Soft Robotics (RoboSoft)*, pages 1–7, 2023.

[86] Yasunori Toshimitsu, Ki Wan Wong, Thomas Buchner, and Robert Katzschmann. Sopra: Fabrication & dynamical modeling of a scalable soft continuum robotic arm with integrated proprioceptive sensing. In *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 653–660. IEEE, 2021.

[87] Chang-Chun Hua and Xiaoping P Liu. Delay-dependent stability criteria of teleoperation systems with asymmetric time-varying delays. *IEEE Transactions on Robotics*, 26(5):925–932, 2010.

[88] Takeshi Hatanaka, Nikhil Chopra, Masayuki Fujita, and Mark W Spong. *Passivity-based control and estimation in networked robotics*. Springer, 2015.

[89] Cosimo Della Santina, Ryan Landon Truby, and Daniela Rus. Data–driven disturbance observers for estimating external forces on soft robots. *IEEE Robotics and automation letters*, 5(4):5717–5724, 2020.

[90] Ryan L Truby, Michael Wehner, Abigail K Grosskopf, Daniel M Vogt, Sebastien GM Uzel, Robert J Wood, and Jennifer A Lewis. Soft somatosensitive actuators via embedded 3d printing. *Advanced Materials*, 30(15):1706383, 2018.

[91] Ying Chen, Miao Yu, Hugh A Bruck, and Elisabeth Smela. Compliant multi-layer tactile sensing for enhanced identification of human touch. *Smart Materials and Structures*, 27(12):125009, 2018.

[92] Eli M Barnett, Julian J Lofton, Miao Yu, Hugh A Bruck, and Elisabeth Smela. Targeted feature recognition using mechanical spatial filtering with a low-cost compliant strain sensor. *Scientific reports*, 7(1):1–14, 2017.

[93] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural computation*, 9(8):1735–1780, 1997.

[94] Lasitha Weerakoon, Zepeng Ye, Rahul Subramonian Bama, Elisabeth Smela, Miao Yu, and Nikhil Chopra. Adaptive tracking control of soft robots using integrated sensing skins and recurrent neural networks. In *2021 IEEE International Conference on Robotics and Automation (ICRA)*, pages 12170–12176. IEEE, 2021.

[95] Tao Wang, Yunce Zhang, Zheng Chen, and Shiqiang Zhu. Parameter identification and model-based nonlinear robust control of fluidic soft bending actuators. *IEEE/ASME Transactions on Mechatronics*, 24(3):1346–1355, 2019.

[96] Mahmoud H Mohamed, Soha H Wagdy, Mostafa A Atalla, Aliaa Rehan Youssef, and Shady A Maged. A proposed soft pneumatic actuator control based on angle estimation from data-driven model. *Proceedings of the Institution of Mechanical Engineers, Part H: Journal of Engineering in Medicine*, page 0954411920911277, 2020.

[97] Sang Yup Kim, Youngwoo Choo, R Adam Bilodeau, Michelle C Yuen, Gilad Kaufman, Dylan S Shah, Chinedum O Osuji, and Rebecca Kramer-Bottiglio. Sustainable manufacturing of sensors onto soft systems using self-coagulating conductive pickering emulsions. *Science Robotics*, 5(39), 2020.

[98] Dong Zhou, Yingxiang Liu, Xintian Tang, and Jiaoguo Zhao. Differential sensing method for multidimensional soft angle measurement using coiled conductive polymer fiber. *IEEE Transactions on Industrial Electronics*, 2020.

[99] Chao-Xuan Liu and Jin-Woo Choi. Patterning conductive pdms nanocomposite in an elastomer using microcontact printing. *Journal of Micromechanics and Microengineering*, 19(8):085019, 2009.

[100] Ying Chen, Miao Yu, Hugh A Bruck, and Elisabeth Smela. Stretchable touch-sensing skin over padding for co-robots. *Smart Materials and Structures*, 25(5):055006, 2016.

[101] Sreeshankar Satheeshbabu, Naveen Kumar Uppalapati, Girish Chowdhary, and Girish Krishnan. Open loop position control of soft continuum arm using deep reinforcement learning. In *2019 International Conference on Robotics and Automation (ICRA)*, pages 5133–5139. IEEE, 2019.

[102] Cecilia Laschi, Barbara Mazzolai, and Matteo Cianchetti. Soft robotics: Technologies and systems pushing the boundaries of robot abilities. *Science Robotics*, 1(1):eaah3690, 2016.

[103] Jean-Jacques E Slotine and Weiping Li. Composite adaptive control of robot manipulators. *Automatica*, 25(4):509–519, 1989.

[104] Hanlei Wang. Adaptive control of robot manipulators with uncertain kinematics and dynamics. *IEEE Transactions on Automatic Control*, 62(2):948–954, 2016.

[105] Chien-Chern Cheah, Chao Liu, and Jean-Jacques E Slotine. Adaptive jacobian tracking control of robots with uncertainties in kinematic, dynamic and actuator models. *IEEE transactions on automatic control*, 51(6):1024–1029, 2006.

[106] Thummaros Rugthum and Gang Tao. Adaptive actuator failure compensation for cooperative robotic manipulators with parameter uncertainties. *International Journal of Adaptive Control and Signal Processing*, 35(9):1916–1940, 2021.

[107] Lasitha Weerakoon and Nikhil Chopra. Passivity-based task space control of hybrid rigid-soft (hyriso) robots with parametric uncertainty. In *2022 IEEE 61st Conference on Decision and Control (CDC)*, pages 771–776, 2022.

[108] Valentin Falkenhahn, Tobias Mahl, Alexander Hildebrandt, Rüdiger Neumann, and Oliver Sawodny. Dynamic modeling of constant curvature continuum robots using the euler-lagrange formalism. In *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 2428–2433, 2014.

[109] Giovanni Tonietti and Antonio Bicchi. Adaptive simultaneous position and stiffness control for a soft robot arm. *IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2:1992–1997 vol.2, 2002.

[110] Travis Burch, John P Lathrop, William L Scott, and Derek A Paley. Feedback control of a soft swinging appendage. In *2020 3rd IEEE International Conference on Soft Robotics (RoboSoft)*, pages 1–6. IEEE, 2020.

[111] Mark W Spong. The swing up control problem for the acrobot. *IEEE control systems magazine*, 15(1):49–55, 1995.

[112] Lasitha Weerakoon and Nikhil Chopra. Swing up control of a soft inverted pendulum with revolute base. In *2021 60th IEEE Conference on Decision and Control (CDC)*, pages 685–690. IEEE, 2021.

[113] I Fantoni, R Lozano, F Mazenc, and A Annaswamy. Stabilization of a two-link robot using an energy approach. In *1999 European Control Conference (ECC)*, pages 2886–2891. Karlsruhe, Germany, 1999.

[114] Javier Moreno-Valenzuela and Carlos Aguilar-Avelar. *Motion control of underactuated mechanical systems*, volume 1. Springer, 2018.

[115] Soukaina Krafes, Zakaria Chalh, and Abdelmjid Saka. A review on the control of second order underactuated mechanical systems. *Complexity*, 2018, 2018.

[116] Isabelle Fantoni, Rogelio Lozano, and Mark W Spong. Energy based control of the pendubot. *IEEE Transactions on Automatic Control*, 45(4):725–729, 2000.

[117] Rogelio Lozano, Isabelle Fantoni, and Dan J. Block. Stabilization of the inverted pendulum around its homoclinic orbit. *Systems & Control Letters*, 40(3):197–204, 2000.

[118] JW Grizzle, ER Westervelt, and C Canudas-de Wit. Event-based pi control of an underactuated biped walker. In *42nd IEEE International Conference on Decision and Control*, volume 3, pages 3091–3096, 2003.

[119] Enrico Franco and Arnau Garriga-Casanovas. Energy-shaping control of soft continuum manipulators with in-plane disturbances. *The International Journal of Robotics Research*, 0(0):0278364920907679, 0.

[120] M. W. Spong. Swing up control of the acrobot. In *Proceedings of the 1994 IEEE International Conference on Robotics and Automation*, pages 2356–2361 vol.3, 1994.

[121] Mark W. Spong and Laurent Praly. Control of underactuated mechanical systems using switching and saturation. In A. Stephen Morse, editor, *Control Using Logic-Based Switching*, pages 162–172, Berlin, Heidelberg, 1997. Springer Berlin Heidelberg.

[122] C. Della Santina, M. Bianchi, G. Grioli, F. Angelini, M. Catalano, M. Garabini, and A. Bicchi. Controlling soft robots: Balancing feedback and feedforward elements. *IEEE Robotics Automation Magazine*, 24(3):75–83, 2017.

[123] Romeo Ortega, Mark W Spong, Fabio Gómez-Estern, and Guido Blankenstein. Stabilization of a class of underactuated mechanical systems via interconnection and damping assignment. *IEEE transactions on automatic control*, 47(8):1218–1233, 2002.

[124] Romeo Ortega and Eloisa Garcia-Canseco. Interconnection and damping assignment passivity-based control: A survey. *European Journal of control*, 10(5):432–450, 2004.

[125] A. Donaire, R. Mehra, R. Ortega, S. Satpute, J. G. Romero, F. Kazi, and N. M. Singh. Shaping the energy of mechanical systems without solving partial differential equations. *IEEE Transactions on Automatic Control*, 61(4):1051–1056, 2016.

[126] M. W. Spong and D. J. Block. The pendubot: a mechatronic system for control research and education. In *Proceedings of 1995 34th IEEE Conference on Decision and Control*, volume 1, pages 555–556 vol.1, 1995.

[127] Bara J Emran and Homayoun Najjaran. A review of quadrotor: An underactuated mechanical system. *Annual Reviews in Control*, 46:165–180, 2018.

[128] John Hauser, Shankar Sastry, and Petar Kokotovic. Nonlinear control via approximate input-output linearization: The ball and beam example. *IEEE trans. on automatic control*, 37(3):392–398, 1992.

[129] Yang Liu and Hongnian Yu. A survey of underactuated mechanical systems. *IET Control Theory & Applications*, 7(7):921–935, 2013.

[130] Mark W Spong and Laurent Praly. Control of underactuated mechanical systems using switching and saturation. *Control using logic-based switching*, pages 162–172, 2005.

[131] Reza Olfati-Saber. *Nonlinear control of underactuated mechanical systems with application to robotics and aerospace vehicles*. PhD thesis, Massachusetts Institute of Technology, 2001.

[132] Takamasa Horibe and Noboru Sakamoto. Nonlinear optimal control for swing up and stabilization of the acrobot via stable manifold approach: Theory and experiment. *IEEE Transactions on Control Systems Technology*, 27(6):2374–2387, 2018.

[133] Sudarsandhari Shibani Willson, Philippe Mullhaupt, and Dominique Bonvin. Quotient method for controlling the acrobot. In *Proceedings of the 48h IEEE Conference on Decision and Control (CDC) held jointly with 2009 28th Chinese Control Conference*, pages 1770–1775. IEEE, 2009.

[134] Ancai Zhang, Jinhua She, Xuzhi Lai, and Min Wu. Global stabilization control of acrobot based on equivalent-input-disturbance approach. *IFAC Proceedings Volumes*, 44(1):14596–14601, 2011.

[135] J Alfredo Rosas-Flores, Jaime Alvarez-Gallegos, and Rafael Castro-Linares. Stabilization of a class of underactuated systems. In *Proceedings of the 39th IEEE Conference on Decision and Control (Cat. No. 00CH37187)*, volume 3, pages 2168–2173. IEEE, 2000.

[136] Mark W Spong, Peter Corke, and Rogelio Lozano. Nonlinear control of the reaction wheel pendulum. *Automatica*, 37(11):1845–1851, 2001.

[137] Robert T Bupp, Dennis S Bernstein, and Vincent T Coppola. A benchmark problem for nonlinear control design. *International Journal of Robust and Nonlinear Control: IFAC-Affiliated Journal*, 8(4-5):307–310, 1998.

[138] Mark W Spong. Partial feedback linearization of underactuated mechanical systems. In *Proceedings of IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS'94)*, volume 1, pages 314–321. IEEE, 1994.

[139] Hassan K Khalil. Noninear systems. *Prentice-Hall, New Jersey*, 3, 1996.

[140] Priyanka Rao, Quentin Peyron, and Jessica Burgner-Kahrs. Shape representation and modeling of tendon-driven continuum robots using euler arc splines. *IEEE Robotics and Automation Letters*, 7(3):8114–8121, 2022.

[141] Chao Liu, Chien Chern Cheah, and Jean-Jacques E Slotine. Adaptive task-space regulation of rigid-link flexible-joint robots with uncertain kinematics. *Automatica*, 44(7):1806–1814, 2008.

[142] Richard M Murray. *A mathematical introduction to robotic manipulation*. CRC press, 2017.

[143] Ping Hsu, John Mauser, and Shankar Sastry. Dynamic control of redundant manipulators. *Journal of Robotic Systems*, 6(2):133–148, 1989.