State Estimation and Control of Continuum Robots with Applications in Force Sensing and Tissue Palpation

A Dissertation Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Joshua Gaston

May 2026

© by Joshua Gaston , 2026
All Rights Reserved.

Acknowledgments

I would like to express my sincere gratitude to my advisor, Dr. Caleb Rucker, for his guidance, mentorship, and continued support throughout this research. I also thank my committee members, Dr. Tayfun Ertop, Dr. Dustin Crouch, and Dr. Eric Barth, for their valuable feedback and thoughtful discussions that helped shape this work.

This research was supported in part by the National Science Foundation under Grant No. EFMA-1935278, the Advanced Research Projects Agency for Health (ARPA-H) under Award Number D24AC00415-00, as well as the Department of Mechanical and Aerospace Engineering at the University of Tennessee, Knoxville. The content here is solely the responsibility of the author and does not necessarily represent the official views of these organizations.

I am grateful to my colleagues in the REACH Lab for their collaboration and support, and for fostering a productive and engaging research environment.

Finally, I would like to thank my family and friends for their encouragement and support throughout this journey.

Abstract

Soft and continuum robots offer unique advantages for safe interaction, adaptability, and operation in confined environments due to their highly compliant structures. However, their underactuated nature and effectively infinite degrees of freedom pose significant challenges for modeling, control, and state estimation, particularly for large flexible robots. This dissertation addresses these challenges by developing mechanics-based modeling, control, and estimation frameworks that enable robust task-space control, full-state estimation from sparse measurements, and intrinsic force sensing without dedicated distal sensors.

First, a task-space control framework is developed for planar continuum robots using a highly underactuated discrete rod mechanics model formulated in maximal coordinates. This approach enables straightforward and computationally efficient control design. Input–output feedback linearization combined with sliding mode control achieves robust task-space tracking despite significant modeling uncertainty, inaccurate initial conditions, and output-only sensing. A model-based observer is introduced to estimate the full system state from sparse measurements, enabling closed-loop control with minimal sensing requirements.

This modeling and estimation framework is then extended to three-dimensional continuum robots using a symmetric constrained-Lagrangian formulation with quaternion-based orientation representation and constraint stabilization. A physically intuitive observer based on virtual forces and moments is developed to estimate the robot's dynamic state. Experimental validation on a large tendon-driven continuum robot demonstrates accurate state estimation during dynamic actuation using optical tracking.

Finally, the dissertation leverages the inherent compliance of continuum robots to enable intrinsic force sensing for surgical robotic palpation. Deflection-based force-sensing probes are designed and modeled using quasi-static Cosserat and Kirchhoff rod formulations,

enabling estimation of both bending and axial contact forces without dedicated tip force sensors. Experimental validation on a surgical continuum robot demonstrates accurate force estimation and supports applications such as tissue stiffness mapping and tumor localization.

Collectively, this work advances the modeling, control, estimation, and sensing capabilities of continuum robots, providing a unified, sensor-efficient foundation for robust autonomy in medical and unstructured environments.

Table of Contents

1 Introduction **1**

1.1 Overview and Objectives 2

2 Background on Modeling and Control of Continuum Robots **5**

2.1 Dynamic Modeling 5

2.2 Observers 7

2.3 Control 7

2.4 Contributions 8

3 2D Dynamics **10**

3.1 Modeling 10

3.2 Force-Based Observer 18

3.3 Controller 19

3.4 Experiments 22

4 3D Dynamics **27**

4.1 3D Modeling 27

4.2 Observer 41

4.3 Robot Design 42

4.4 Experiments 44

4.5 Results 47

4.5.1 Computational Speed 47

4.5.2 Open Loop Model Estimation 47

4.5.3 Force Based Observer Estimation 48

5 Background of Surgical Palpation via Continuum Robots 53

6 Force Sensing and Tissue Palpation 57

6.1 Hardware 58

6.2 Modeling 62

6.3 Force Estimation Algorithm 65

6.4 Initial Verification in Simulation 69

6.5 Experimental Validation 71

6.5.1 Validation Results 75

6.5.2 Simulation Environment 76

6.5.3 Stiffness Sensing in Simulation 77

6.5.4 Phantom Stiffness Mapping 79

6.6 Chapter Summary 81

7 Three Axis Force Sensing and Tissue Palpation 82

7.1 Methodology 83

7.1.1 Hardware 83

7.1.2 Tool Design 84

7.1.3 Kinematics Algorithm 88

7.1.4 Force Estimation Algorithm 93

7.2 Experiments 94

7.2.1 Simulation Experiments 94

7.2.2 Force Estimation Experiments 100

7.2.3 Stiffness Estimation Experiments 102

7.3 Chapter Summary 105

8 Conclusion 106

Bibliography 108

Vita 120

List of Tables

3.1 Simulated Model Parameters 23
3.2 Controller Parameters 23

6.1 Magnetic Tracking Accuracy (5DOF, 95% CI) 62
6.2 Example of the tip compliance matrix C. The inverse of each value corresponds to the principal stiffness along that mode. 68
6.3 Calibration results grouped by hole index (holes 0–2). 75
6.4 Validation results at unseen holes 3–4 using calibrated parameters. 76
6.5 Mean absolute error (MAE) by eigen-direction in the tip frame. 77

7.1 Magnetic Tracking Accuracy (5DOF, 95% CI) 87
7.2 Model Parameters 96

List of Figures

3.1 An arbitrary section of a rod from a to s along σ is shown on the left. The distributed forces and moments are represented as f and l respectively. The internal forces and moments are represented as $\mathbf{n}$ and ζ respectively. On the right, the internal forces $\mathbf{n}$ and moments ζ are indicated on the links and nodes on either side of the i^{th} node of a set of discrete links. The dots represent the nodes at position $\mathbf{p}$, and the dashed lines represent the "links" of the system. The tangent angle of each link is indicated by θ. 11

3.2 The overall structure of the control system. 19

3.3 Snapshots of the dynamic robot behavior during the circle task after the tracking error has converged to zero (after 0.2 seconds). 23

3.4 Snapshots of the dynamic robot behavior during the line following task after the tracking error has converged to zero (after 0.2 seconds). 24

3.5 Top: The desired reference trajectory is compared to the robot output for the circle tracking task. Bottom: The control torques indicate that significant inertial dynamics are being overcome. 25

3.6 Top: The desired reference trajectory is compared to the robot output for the cyclical line tracking task. Bottom: The control torques indicate that significant inertial dynamics and couplings between the actuated DOFs are being overcome. 26

4.1 An arbitrary section of a rod is indicated from c to s along σ. The distributed forces and moments are represented as $\mathbf{f}$ and $\mathbf{l}$ respectively. The internal forces and moments are represented as $\mathbf{n}$ and $\mathbf{m}$ respectively. Three positions shown by $\mathbf{p}$ are indicated along the center-line of the rod at the proximal end c, the distal end s, and an arbitrary point $\mathbf{p}_o$. 28

4.2 To approximate an arbitrary continuous rod segment, a series of links and nodes can be connected in series via internal forces $\mathbf{n}$ and moments $\mathbf{m}$ to formulate a discrete elastic rod model. The internal forces, moments, and curvatures $\mathbf{u}$ can naturally reside on the nodes indicated by black dots. Alternatively, the positions $\mathbf{p}$, orientation frames $\mathbf{R}$, and angular velocities ω can be indicated on the links. We can let the i^{th} link lie in between the i^{th} and the $i+1^{th}$ node. 29

4.3 The j^{th} actuator input can be seen acting on the backbone of the robot. The j^{th} actuator tendon passes along guide channels located $r_{i,j}$ away from the i^{th} link frame. At the i^{th} tendon guide channel, position of the j^{th} tendon is indicated as $\mathbf{p}_{i,j}$. The unit vector describing the direction of the tendon tension from $\mathbf{p}_{i,j}$ to $\mathbf{p}_{i-1,j}$ is described by $\mathbf{t}_i$. Likewise, the unit vector describing the direction of the tendon tension from $\mathbf{p}_{i,j}$ to $\mathbf{p}_{i+1,j}$ is described by $\mathbf{t}_{i+1}$. . . . 37

4.4 This figure shows a block diagram of the observer structure. The tendon length controller can use ether of the methods detailed in the "Tendon Actuation via Displacement Commands" subsection. $\mathbf{L}_{ref}$ is the reference tendon length, $\boldsymbol{\tau}$ is the actuation tension of the tendons, $\mathbf{q}$ is the state vector, and $\mathbf{g}$ is the observer forces and moments 38

4.5 This figure shows an illustration of the experimental continuum robot. The upper left side of the figure shows the threaded outserts. These are threaded on the outside of the helical bellows tube allowing for easily added tendon routing channels. The lower left side of the figure shows the inserts. These are threaded inside the tube and allow for a carbon fiber rod to be fixed in the center of the robot. The right side of the figure shows an illustration of the apparatus in its assembled state. 44

4.6 This figure shows the experimental continuum robot set up. The left side of the figure shows the robot in its undeformed shape. The right side of the figure shows the robot deforming under actuation. The optical tracking system, helical bellows backbone, 3D-printed outserts, Kevlar tendons and reflective markers are all indicated in the figure. 45

4.7 This figure shows the solution time of both MATLAB's mldivide() (indicated by the dashed line), and the Thomas block tri-diagonal algorithm (indicated by the solid line) as the number of links used to discretize the Kirchhoff rod increase. The mldivide() algorithm scales cubically but is more efficient for lower numbers of links. Alternatively, the Thomas algorithm scales linearly and is therefore more efficient for higher numbers of links. 46

4.8 This figure shows plots of the tip and center positions of the robot and the model as it follows a given trajectory. No Restorative forces or moments are acting on the model. Here X_{Est}, Y_{Est}, and Z_{Est} represent the estimated x, y, and z values. Similarly, X_{Meas}, Y_{Meas}, and Z_{Meas} represent the measured x, y, and z values. 49

4.9 This figure contains the open loop position error magnitudes of 6 equally spaced measured points along the robot arm as well as the tip orientation error. The nondimensionalized mean position errors across all measured points along the robot for the duration of the experiment are 0.0762 of the robot's length. Mean orientation errors about the x, y, and z axis (represented as θ_X, θ_Y, and θ_Z) are 11.60°, 3.15°, and 3.35° respectively. 50

4.10 This figure shows plots of the tip and center positions of the robot and the model as it follows a given trajectory. Restorative forces and moments act on the tip of the robot to compensate for unmodeled forces. Here X_{Est}, Y_{Est}, and Z_{Est} represent the estimated x, y, and z values. Similarly, X_{Meas}, Y_{Meas}, and Z_{Meas} represents the measured x, y, and z values. 51

4.11 This figure contains closed loop observer position error magnitudes of 6 equally spaced measured points along the robot arm as well as tip orientation error. The nondimensionalized mean position errors across all measured points along the robot for the duration of the experiment are 0.0191 of the robot's length. Mean tip orientation errors about the x, y, and z axis (represented as θ_X, θ_Y, and θ_Z) are 2.61°, 2.35°, and 1.43° respectively. 52

6.1 Deflection-based force sensing uses the known stiffness of the robot and the observed deflection from the commanded position to infer the tissue force. . . 58

6.2 This figure shows a top-down view of the robot system used in this work. The motor pack is highlighted in red; the surgical cartridge is highlighted in green, the endoscope is highlighted in purple, the NDI field generator is highlighted in blue, and the surgical camera (not used in this work) is highlighted in yellow. 59

6.3 CAD drawing of palpation tool with the base force sensor mounted at the rear. 60

6.4 Palpation sensors: (a) NDI tabletop field generator workspace. (b) NDI magnetic tracker sensor. (c) CAD drawing of palpation tool. 61

6.5 This figure shows a free body diagram of an arbitrary section of rod from a to b along s. 63

6.6 Simulation of deflection-based force estimation using only magnetic tracking sensor data. 70

6.7 Simulation of force estimation fusing magnetic tracking data and base force sensor data. 70

6.8 This figure shows a drawing of a sliced view of the proximal end of the robot when performing the first force transmission test. 72

6.9 This figure shows a drawing of a sliced view of the proximal end of the robot when performing the second force transmission test. 72

6.10 This figure shows a picture of our robot during palpation. 73

6.11 This figure shows an illustration of the palpation plate. The red indicates our palpation tube when deformed during palpation. The blue shows our palpation rod before palpation. The holes are indicated in green and labeled from 0 to 4. The diameter of each hole is 4 mm. The distance between each hole is 5.5 mm. 74
6.12 Figure shows the estimated (green) and measured (red) palpation force vectors on each palpation hole. 76
6.13 Screenshots from the simulator showcasing CTR interaction with deformable tissue. The tissue block is 2.5 cm × 1.25 cm × 1.25 cm. 78
6.14 Tissue stiffness sensing via palpation in simulation. **Left:** Force–displacement curve for a representative palpation, where the slope defines local tissue stiffness. **Top Right:** Tissue block with two embedded stiff spherical inclusions. **Bottom Right:** Stiffness map obtained from vertical palpation on a 1 × 1 mm grid, revealing the embedded objects. 79
6.15 The top of figure shows the robot during palpation of the tissue phantom. The bottom panel presents a palpation-generated stiffness map of the phantom tissue containing an embedded nodule located 5 mm below the surface. The estimated tumor location is defined as the point of maximum stiffness and is indicated by a blue marker. The ground-truth tumor location is shown in red. Circles are overlaid on both to indicate the approximate tumor boundaries. 80

7.1 Palpation tool prototypes are shown. A) shows a design that translates along the outside of the inner tube of the Virtuoso robot. B) show a design that translates along the inside of the inner tube of the Virtuoso robot. 84
7.2 This figure shows a drawing of the laser patterned inner tube. 86
7.3 This figure shows a 3D drawing of the wire tensioner device. The Green shows the housing structure of the device, the orange shows the spring holder, the purple shows the piston, the blue shows the cover and the yellow shows the NDI wires. 88

7.4 This figure shows a diagram of the internal forces and moments along a section of the robot without the plunger at the tip. 89
7.5 This figure shows a diagram of the internal forces and moments along a section of the robot with the plunger at the tip. 92
7.6 This figure shows an illustration of how the tip forces ($\mathbf{F}_{\text{tip}}$) are applied in simulation. 96
7.7 This figure shows the percentage error in simulation of the estimated forces when the estimator ignores the component of the force in the stiff direction of the robot. The x axis shows the angle the forces are applied to the robot tip, with 0° being when the force is applied directly against the tangent vector of the tip. This figure shows three different robot configurations. IT stands for the inner tube translation. OT stands for outer tube translation. 97
7.8 This figure shows the percent error in simulation of the estimated forces when the estimator includes information from a base force sensor. The x axis shows the angle the forces are applied to the robot tip, with 0° being when the force is applied directly against the tangent vector of the tip. This figure shows three different robot configurations. IT stands for inner tube translation. OT stands for outer tube translation. 98
7.9 This figure shows the percentage error in simulation of the estimated forces when using our spring tip design. The x axis shows the angle the forces are applied to the robot tip, with 0° being when the force is applied directly against the tangent vector of the tip. This figure shows three different robot configurations. IT stands for inner tube translation. OT stands for outer tube translation. 99
7.10 This figure shows our force sensor apparatus fixed to the exterior of the robot endoscope. 100
7.11 Experimental setup used for automated palpation of the phantom using the robotic systems and EM tracking sensor. 102

7.12 Stiffness map obtained using grid-based robotic palpation. Tumor localization error: 3.5 mm. The red dot indicates the measured tumor position. The red circle indicates the outer edge of the tumor based on the "ground-truth" measured tumor location. The blue Dot indicates the estimated tumor position. The blue line indicates the outer edge of the tumor based the palpation generated stiffness map. 104

Chapter 1

Introduction

Soft and continuum robots, which can bend continuously due to their compliant nature [40], have been used in a variety of applications [84], [13], [39], and have demonstrated unique advantages over traditional rigid-linked robots. These benefits are evident across various types and scales of continuum robots, from large robotic systems to miniature medical devices. At both larger and small scales, soft and continuum robots offer safer human-robot interactions and enhanced adaptability due to their inherent compliance. This compliance allows them to navigate and manipulate objects in unstructured environments, where rigid robots may struggle. Additionally, their ability to distribute forces over their deformable structure reduces the risk of damage or injury, making them ideal for collaborative tasks alongside humans. Another key advantage of soft and continuum robots is their capacity to offload actuators, reducing the size and mass along the robot arm. This feature is particularly crucial in small-scale applications, such as surgical robotics. By minimizing the number of actuators at the operational site, these robots enable greater precision and maneuverability in confined spaces, improving outcomes in minimally invasive procedures. Furthermore, recent advances in actuation [23] and manufacturing [69], [91] have enhanced their capabilities. Overall, soft and continuum robots are highly versatile across a broad spectrum of applications.

Despite their advantages, continuum robots face significant challenges in control and state estimation due to their underactuated nature and theoretically infinite degrees of freedom.

These challenges are particularly pronounced in large-scale, highly flexible systems, where precise manipulation and real-time feedback are critical.

Accurate state estimation (determining the robot's shape and position) is essential for accurate simulation, control, design, and sensing. While some continuous shape sensors (e.g., fiber Bragg gratings or embedded strain sensors) enable direct state measurement, they are often expensive and limited by low refresh rates, making them impractical for many real-world applications. Consequently, alternative state estimation methods, relying on some type of mechanics-based model, are highly motivated, as reviewed in [92, 87, 10, 17].

Moreover, solving the state estimation problem can unlock additional advantages unique to continuum robots. For instance, the compliant structure of a continuum robot can be leveraged to estimate external contact forces without dedicated force sensors—enabling intrinsic force sensing for delicate tasks like minimally invasive surgery or safe human-robot interaction. This capability further distinguishes continuum robots from traditional rigid systems, which often require dedicated sensors for similar functionality.

Thus, advancing state estimation techniques is not only a prerequisite for overcoming control challenges but also a pathway to enhancing the autonomy, adaptability, and cost-effectiveness of continuum robots in practical applications. To this end, this study investigates solutions to the challenges of sensory estimation and control in continuum robots. The work is organized into the following chapters.

1.1 Overview and Objectives

The next three chapters present published research in [63][1] and [24] [2] that develops a force-based state observer and controller for highly flexible continuum robots in both planar (2D) and spatial (3D) cases.

[1]Rucker, C., Barth, E. J., Gaston, J., and Gallentine, J. C. (2022). Task-space control of continuum robots using underactuated discrete rod models. In 2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), pages 10967–10974. © 2022 IEEE

[2]Used with permission of American Society of Mechanical Engineers (ASME), from "A 3D Discrete Elastic Rod Model and Observer for Continuum Robots", Gaston, J. B., Kumar, N. S., Barth, E. J., and Rucker, C., 18, 2026; permission conveyed through Copyright Clearance Center, Inc.

Chapter 2 provides a background of related work and establishes motivation for the force-based state observer and controller detailed in chapters 3 and 4. Chapter 3 introduces a planar model-based observer and task-space controller, which is validated in simulation. Chapter 4 extends these results to the fully 3D case, with an emphasis on state estimation.

The remaining chapters present work that builds on the insights from the earlier chapters, focusing on the estimation of contact forces at the robot tip in surgical contexts. Chapter 5 reviews the relevant literature on force estimation. Chapter 6 presents a novel force-sensing approach using a surgical robot. This approach is enhanced by the development of a new tool described in Chapter 7 that addresses some of the limitations of the previous chapter and provides a more robust apparatus. Finally, a conclusion to this work is provided in Chapter 8.

These sections are described in more detail below.

Overview of Chapter 2- 4

Chapter 3 develops a task-space control approach for planar continuum robots that is robust to modeling errors and requires minimal sensor information. The controller is based on a highly underactuated discrete rod mechanics model expressed in maximal coordinates, eliminating the need to convert to a classical robot dynamics formulation. This modeling choice promotes straightforward control design, efficient implementation, and computational efficiency. We apply input–output feedback linearization to the model, augment it with sliding mode control to enhance robustness, and design an observer to estimate the full system state from sparse output measurements. Simulation results demonstrate that exact task-space reference tracking can be achieved even in the presence of significant modeling errors, inaccurate initial conditions, and output-only sensing.

Chapter 4 extends the maximal-coordinate modeling approach to 3D using a symmetric, constrained-Lagrangian formulation with quaternions and Baumgarte constraint stabilization. We also introduce a physically intuitive, model-based observer that estimates the full state of a continuum robot by incorporating virtual forces and moments. The model is then simulated to demonstrate computational speed, robustness to modeling error, and robustness to sensor noise. Both the open-loop model and the observer are then validated

through optical tracking experiments on a large tendon-driven continuum robot prototype, demonstrating accurate estimation of the robot's dynamic state during actuation.

Overview of Chapter 5- 7

Chapter 6 develops custom hardware to be used in a Virtuoso surgical robot in order to palpate tissue. Additionally, a mathematical model based on a Kerkhoff rod is used alongside a weighted least squares method to back calculate tip interaction forces acting on the robot. These estimated forces can be used to determine the stiffness properties when pushing against a surface, opening the door to a new method of detecting subsurface tumors with a robot. Simulations and physical experiments using an external force sensor are performed to verify our approach.

Chapter 7 details a continuation of the work in chapter 6 by addressing hardware limitations in determining forces along the stiff direction (near the tip tangent vector) of the palpation device. This is done by developing a new palpation tool composed of a spring plunger at the end of a laser-patterned tube, thus allowing for deformation in bending and along the tangent vector of the tip of the robot. Additionally, modifications of the model in chapter 6 are developed in order to fully utilize the hardware changes. Rigorous experimentation involving force sensors and phantom tissue with hard nodules embedded under the surface of the tissue is then used to verify this method's ability not only to sense forces, but to locate sub surface tumors and map the stiffness of a tissue surface.

Chapter 2

Background on Modeling and Control of Continuum Robots

In this chapter, we discuss the relevant literature surrounding the modeling, control, and state estimation of continuum robots to motivate Chapters 3 and 4. We break this chapter into three sections. First, we discuss common dynamic modeling strategies for continuum robots. Then we review approaches to designing an observer. Lastly, we discuss task space controllers.

2.1 Dynamic Modeling

When modeling a continuum robot, a widely used approach for modeling the geometry of the robot is the piecewise-constant curvature (PCC) model, which discretizes a continuous slender object into a series of constant curves [12, 94, 37, 89, 61, 18, 26]. This method is naturally suited for many continuum robots because of the serial segment structure that is typical in continuum robot actuation. The segments are often designed to attain constant curvature in steady-state when actuated. However, while PCC is a good approximation for certain continuum robots under negligible external loads and negligible inertial dynamics [93], its geometric assumption cannot capture higher-order vibrations and variations in curvature under dynamic conditions without finer discretization. A PCC model is one example of a representation in "minimal coordinates" (it defines the shape of the manipulator in the

least number of variables, i.e. the number of degrees of freedom (DOF) of the system). PCC coordinates are also "relative coordinates" in the sense that they define a relationship between absolute positions and orientations. In continuum robot dynamics, this always leads to a fully dense mass matrix, resulting in higher computational costs associated with solving for the coordinate accelerations, and the computation time scales cubically with the resolution (number of PCC segments used) [81, 28]. Other minimal-coordinate methods, such as actuator lengths [20], piecewise-constant strains [60, 59, 9, 86], parameterized curves [50], and virtual rigid-link joint angles [88] share in this limitation owing to the dense mass matrix that comes with minimal coordinate representation.

In contrast to the relative-coordinate, minimal-coordinate approach of PCC and similar frameworks, continuum robot geometry can be represented in absolute coordinates (positions and orientations) with appropriate constraints (representing mechanical assumptions like inextensibility, for example). This is generally called a model in "maximal coordinates" because there are more coordinates than degrees of freedom, and constraints on the coordinates must be explicitly resolved. The dynamic equations of maximal coordinate models can be obtained by discretizing the partial differential equations of continuous nonlinear elastic theories for slender objects, such as Kirchhoff rod theory and Cosserat rod theory [8, 7, 83, 57, 34, 45, 81, 44]. We show in the next chapter that certain spatial discretization methods for rod models are equivalent to models for a chain of rigid bodies connected by elastic joints.

Absolute-coordinate, maximal-coordinate approaches have many benefits (computational efficiency, generality, ease of programming, physically intuitive, ease of applying additional constraints), but they are often seen as difficult to use for control development, as they are not written in the standard minimal dynamics form familiar to control theory. In our work, we would like to promote the contrary view: underactuated maximal coordinate models obtained from discrete mechanics models or discretization of a continuous PDE *are* suitable for control design and efficient control implementation, particularly in task space.

2.2 Observers

Many control algorithms require full-state knowledge [25, 16], which is challenging to provide due to limited sensing and the infinite degrees of freedom of continuum robots. While continuous shape sensors [75] can measure the state vector directly, they are often expensive and lack the bandwidth for dynamic environments. This has driven research into state estimation strategies [98] and diverse modeling techniques [3, 27].

State estimation for continuum robots often relies on mathematical models that predict state trajectories. Sensor feedback is then incorporated to correct for discrepancies between the model and the physical system. This can be done through various methods. In robotics, types of Kalman filters (optimal observers) are commonly applied to mathematical models for state estimation [79]. An Extended Kalman Filter (EKF) approach has been popular for estimating states of dynamic continuum robots [4, 53, 67]. Robust [78, 46] and adaptive [41] Kalman filtering approaches have been applied to mitigate the effects of model uncertainties and sensor noise. Finite-time observers have also been implemented on continuum robots using sliding mode control [73] and Lyapunov functions [43]. Other researchers have explored using data-driven models in robot state estimation [71, 38].

The above estimation methods have been successful in their various applications, but the process of determining stable and effective filter gains is complex and not physically intuitive. Alternatively, our work presents an intuitive force-based observer that injects a virtual force and moment into the model (based on easy-to-tune, energetically passive PD control gains) at any point where position/orientation data can be sensed, even if data is only available at a single point (e.g. the tip). Variations of this method were also proposed in [99, 98] that treats it from a boundary-observer perspective on the continuous PDE. However, it was not verified physically in a fully 3D experiment.

2.3 Control

The majority of previous work on dynamic control of continuum manipulators assumes a fully actuated model (e.g. [71, 19, 88]) in minimal coordinates such as piecewise constant

curvatures. Some work has considered control with under-actuated models using parametric curves [50], or reduced order models generated from general 3D finite element simulations [80, 37, 42], while early continuum robot control work acknowledged and modeled the underactuated dynamics [29], formulating linear boundary control laws that sought to stabilize the system and reduce vibration. A closely related work to what we present in the next chapter is [70] which showed that m actuators are sufficient to control an underactuated soft robot in an m degree of freedom task space. This was done by leveraging the operational space dynamics formulation and projecting the required actuation torques into a lower dimensional space using muscle synergies.

We believe the input-output feedback linearization part of our controller might be numerically equivalent to the operational-space control in [70]. However, the formulation herein may facilitate a more straightforward implementation and efficient computation, since our constrained Lagrangian model (or equivalently the discretized PDE model) never needs to be explicitly expressed in either minimal coordinates or operational space, nor does it need to consider muscle synergies. In addition, robustness and state estimation were left to future work in [70]. Herein we contributed toward those goals by proposing a sliding mode outer-loop controller and a passive observer design. We additionally include simulations with parametric modeling error and limited sensing.

2.4 Contributions

In Chapters 3 and 4, we introduce a novel state estimator for soft robots, utilizing a model based on a first-principle PDE discretization of an elastic rod in maximal coordinates. This formulation enables efficient implementation for large numbers of links while providing an accurate and intuitive method for reconstructing unmeasured states from minimal measurements by modeling the system's internal dynamics and driving the model with sensor measurements. Furthermore, we also present an input-output feedback linearization control design methodology that is augmented by a sliding mode outer loop that provides robustness to parametric uncertainty. Over the next two chapters, we aim to verify the approach in

simulation, as well as demonstrate the observer in real time on a tendon-driven soft robotic prototype.

Chapter 3

2D Dynamics

This chapter reviews the work published in [63] and details the design of an observer and controller for a 2D continuum robot. Starting with a derivation of the mechanics based model, we will then discuss the use of this model to create a state estimator. Following this, we will derive a task space controller to be used in combination with the observer. Finally, this section concludes with a discussion of the experiments performed using this approach.

3.1 Modeling

PDE Perspective

We will start by describing the dynamics of a classical Kirchhoff rod in 2D, written as the following set of nonlinear partial differential equations [2]:

$$\begin{aligned} \mathbf{p}' &= \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} \\ \theta' &= u \\ \mathbf{n}' + \mathbf{f} &= \rho A \ddot{\mathbf{p}} \\ \zeta' + \mathbf{p}' \times \mathbf{n} + \ell &= \rho I \ddot{\theta} \end{aligned} \tag{3.1}$$

where $'$ denotes a derivative with respect to arc length, s, $\dot{}$ denotes a derivative with respect to time, t, $\mathbf{p}(s,t) \in \mathbb{R}^2$ is the position along the rod, $\theta(s,t) \in \mathbb{R}$ is the tangent angle

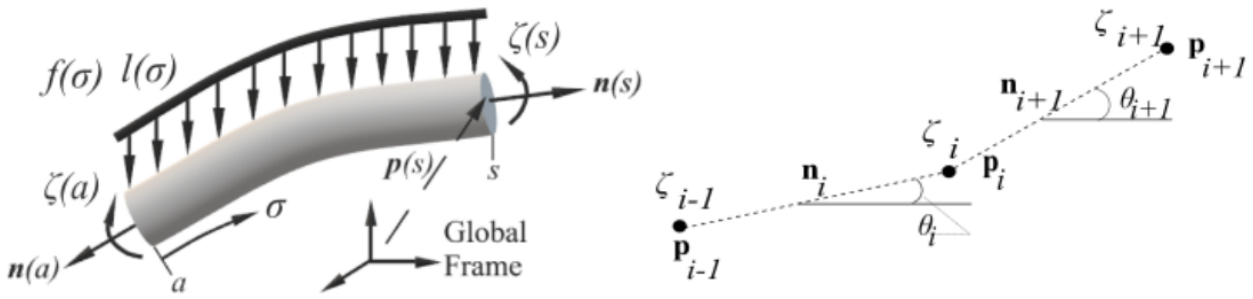


Figure 3.1: An arbitrary section of a rod from a to s along σ is shown on the left. The distributed forces and moments are represented as f and l respectively. The internal forces and moments are represented as $\mathbf{n}$ and ζ respectively. On the right, the internal forces $\mathbf{n}$ and moments ζ are indicated on the links and nodes on either side of the i^{th} node of a set of discrete links. The dots represent the nodes at position $\mathbf{p}$, and the dashed lines represent the "links" of the system. The tangent angle of each link is indicated by θ.

measured counterclockwise from the global x axis, $u(s,t) \in \mathbb{R}$ is the curvature, $\mathbf{n}(s,t) \in \mathbb{R}^2$ is the internal force, $\zeta \in \mathbb{R}$ is the internal moment about the out of the plane axis (counter-clockwise positive), ρ is the material density, A is the cross sectional area, I is the second area moment of the cross section, $\mathbf{f}(s,t) \in \mathbb{R}^2$ is an external distributed force, and $\ell(s,t) \in \mathbb{R}$ is an external distributed moment.

The equations in (3.1) are incomplete until a constitutive law is formulated to relate u and ζ.

The following viscoelastic model is commonly assumed:

$$\zeta = EIu + BI\dot{u} \tag{3.2}$$

where E is Young's modulus, I is the second area moment of the cross section, and B is a damping modulus. Substituting this into (3.1), we get a complete PDE set in terms of state variables $\mathbf{p}$, θ, and $\mathbf{n}$:

$$\begin{aligned} \mathbf{p}' &= \begin{bmatrix} \cos\theta \\ \sin\theta \end{bmatrix} \\ \mathbf{n}' + \mathbf{f} &= \rho A \ddot{\mathbf{p}} \\ EI\theta'' + BI\dot{\theta}'' + \begin{bmatrix} -\sin\theta \\ \cos\theta \end{bmatrix}^{\top} \mathbf{n} + \ell &= \rho I \ddot{\theta} \end{aligned} \tag{3.3}$$

An intuitive and symmetric discretization strategy for these PDE's is to assume a set of $N+1$ ordered equally-spaced nodes in arc length, $\{s_0\ s_1\ s_2\ \ldots\ s_N\}$ (over the total length L) where positions $\mathbf{p}_i$ and moments ζ_i naturally live at each node, while forces $\mathbf{n}_i$ and angles θ_i naturally live on the N edges (or links) halfway between nodes $i-1$ and i as shown in Figure 3.1 (Right).

Assuming a constant step size $h = s_i - s_{i-1}$, centered approximations for the first and second derivatives in arc length can then be applied, resulting in

$$\begin{aligned}
&(\mathbf{p}_i - \mathbf{p}_{i-1})/h = \begin{bmatrix}\cos\theta_i \\ \sin\theta_i\end{bmatrix} \\
&(\mathbf{n}_{i+1} - \mathbf{n}_i)/h + \mathbf{f}(s_i) = \rho A \ddot{\mathbf{p}}_i \\
&EI(\theta_{i-1} - 2\theta_i + \theta_{i+1})/h^2 + \\
&BI(\dot{\theta}_{i-1} - 2\dot{\theta}_i + \dot{\theta}_{i+1})/h^2 + \\
&\begin{bmatrix}-\sin\theta_i \\ \cos\theta_i\end{bmatrix}^\top \mathbf{n}_i + \ell(s_i) = \rho I \ddot{\theta}_i
\end{aligned} \tag{3.4}$$

The system in (3.4) is a set of differential algebraic equations with index 3 (the number of differentiations required to obtain differential equations for all state variables). There are no derivatives for the $\mathbf{n}_i$ variables, and the first equation in the list is an algebraic constraint. One can reduce the index by writing the derivatives of the constraint equations:

$$\begin{aligned}
\mathbf{p}_i - \mathbf{p}_{i-1} &= h \begin{bmatrix}\cos\theta_i \\ \sin\theta_i\end{bmatrix} \\
\dot{\mathbf{p}}_i - \dot{\mathbf{p}}_{i-1} &= h \begin{bmatrix}-\sin\theta_i \\ \cos\theta_i\end{bmatrix} \dot{\theta}_i \\
\ddot{\mathbf{p}}_i - \ddot{\mathbf{p}}_{i-1} &= h \begin{bmatrix}-\cos\theta_i \\ -\sin\theta_i\end{bmatrix} \dot{\theta}_i^2 + h \begin{bmatrix}-\sin\theta_i \\ \cos\theta_i\end{bmatrix} \ddot{\theta}_i
\end{aligned} \tag{3.5}$$

Then enforcing the second derivative of the constraints along with the original differential equations in the DAE set, one obtains an index-1 DAE system which can be written as:

$$
\begin{aligned}
\ddot{\mathbf{p}}_{i-1} - \ddot{\mathbf{p}}_i + h \begin{bmatrix} -\sin\theta_i \\ \cos\theta_i \end{bmatrix} \ddot{\theta}_i &= h \begin{bmatrix} \cos\theta_i \\ \sin\theta_i \end{bmatrix} \dot{\theta}_i^2 \\
\mathbf{n}_{i+1} - \mathbf{n}_i + \mathbf{f}_i &= m\ddot{\mathbf{p}}_i \\
k(\theta_{i-1} - 2\theta_i + \theta_{i+1}) + & \\
b(\dot{\theta}_{i-1} - 2\dot{\theta}_i + \dot{\theta}_{i+1}) + & \\
h \begin{bmatrix} -\sin\theta_i \\ \cos\theta_i \end{bmatrix}^\top \mathbf{n}_i + \ell_i &= J\ddot{\theta}_i
\end{aligned}
\tag{3.6}
$$

where $m = h\rho A$ is the mass associated with each node, $J = h\rho I$ is the density weighted second area moment integrated over the discrete length h, $k = EI/h$ and $b = BI/h$ are equivalent rotational spring and damping constants associated with the joints, $\mathbf{f}_i = h\mathbf{f}(s_i)$ is the effective external force applied at each node, and $\ell_i = h\ell(s_i)$ is the effective external moment applied at each link.

The equations in (3.6) are applicable for $i = 1...N-1$. If the robot has a fixed base at $s = 0$ we may take $\mathbf{p}_0$, $\dot{\mathbf{p}}_0$, $\ddot{\mathbf{p}}_0$, θ_0, and $\dot{\theta}_0$ to be prescribed values. If the robot has a free distal end, we may prescribe

$$
\begin{aligned}
-\mathbf{n}_N + \mathbf{f}_N &= \frac{m}{2}\ddot{\mathbf{p}}_N \\
k(\theta_{N-1} - \theta_N) + b(\dot{\theta}_{N-1} - \dot{\theta}_N) + & \\
h \begin{bmatrix} -\sin\theta_N \\ \cos\theta_N \end{bmatrix}^\top \mathbf{n}_N + \ell_N &= J\ddot{\theta}_N,
\end{aligned}
\tag{3.7}
$$

since there is no material beyond s_n to provide mass or exert internal force or moment on the proximal material.

The discretized and index-reduced PDE is essentially equivalent to an ODE in the positions and angles because the system can be directly solved for their second derivatives by eliminating the internal forces. Since the reduced index system only enforces the second derivative of the constraints, direct integration of all the coordinates in time can result in constraint drift during numerical simulation. There are numerous strategies available to

reduce such drift (such as Baumgarte stabilization), but these are unnecessary for this model. The $\mathbf{p}_i$ and $\dot{\mathbf{p}}_i$ which satisfy the constraints can be determined as a function of the θ_i and $\dot{\theta}_i$ recursively from (3.5). Thus, for simulation purposes, we may use $\theta_1 \ \ldots \ \theta_N$ and $\dot{\theta}_1 \ \ldots \ \dot{\theta}_N$ as a minimal set of state variables to be integrated in time, and simply extract $\ddot{\theta}_1 \ \ldots \ \ddot{\theta}_N$ from the full solution of the maximal coordinate model (3.6). This strategy combines the advantages of maximal coordinates (ease of expression, programming, and computational efficiency) with the advantages of minimal coordinates (impossibility of constraint drift). We aim to show in the controller sub-section that such maximal coordinate models are also convenient for formulating dynamic controllers directly in task space.

Constrained Lagrangian Perspective

The above dynamic model formulated from the original Kirchhoff rod PDEs is equivalent to a multi-body dynamics model that can be generated using either a Newton-Euler approach or a constrained Lagrangian approach. Writing down the equivalent constrained Lagrangian model will help us express the equations in a compact well-known form that will facilitate straightforward programming, efficient numerical solution, and controller design.

Consider a dynamic system in $\mathbf{q}$ subject to constraints $\boldsymbol{\phi}(\boldsymbol{q}) = 0$. Its associated Lagrangian is

$$L(\mathbf{q}, \dot{\mathbf{q}}, \boldsymbol{\lambda}) = T - V + \boldsymbol{\phi}(\mathbf{q})^\top \boldsymbol{\lambda}$$

where T is kinetic energy, and V is potential energy, and $\boldsymbol{\lambda}$ are the Lagrange multipliers. The equations of motion are then

$$\frac{d}{dt}\left(\frac{\partial L}{\partial \dot{\mathbf{q}}}\right) - \frac{\partial L}{\partial \mathbf{q}} + \mathbf{A}^\top \boldsymbol{\lambda} = \mathbf{Q}$$

and

$$\boldsymbol{\phi}(\boldsymbol{q}) = 0$$

where $\mathbf{A} = \frac{\partial \boldsymbol{\phi}(\mathbf{q})}{\partial \mathbf{q}}$, and $\mathbf{Q}$ collects any non-conservative generalized forces that do not arise from the constraints. This forms an index-3 DAE of a form equivalent to (3.4). Taking the

first and second derivatives of the constraint equations, we get

$$\mathbf{A}\dot{\mathbf{q}} = 0, \qquad \mathbf{A}\ddot{\mathbf{q}} = \boldsymbol{\gamma}$$

where $\boldsymbol{\gamma} = -\dot{\mathbf{A}}\dot{\mathbf{q}}$. Then the equivalent index-1 dynamic system can be written as

$$\begin{bmatrix} \mathbf{M} & \mathbf{A}^\top \\ \mathbf{A} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{F} \\ \boldsymbol{\gamma} \end{bmatrix} \tag{3.8}$$

where $\mathbf{F}$ contains all the unconstrained generalized forces. This helps realize and organize the structure of the large system formed by (3.6).

By inspection, we can see that the equations in (3.6) have exactly this form. The generalized coordinates are $\mathbf{q} = \begin{bmatrix} \mathbf{p}_1^\top & \theta_1 & ... & \mathbf{p}_N^\top & \theta_N \end{bmatrix}^\top$. The Lagrangian corresponding to our spatially discretized PDE system is

$$\begin{aligned} L = & \frac{1}{2}\sum_{i=1}^{N-1} m\dot{\mathbf{p}}_i^\top \dot{\mathbf{p}}_i + \frac{m}{4}\dot{\mathbf{p}}_N^\top \dot{\mathbf{p}}_N + \frac{1}{2}\sum_{i=1}^{N} J\dot{\theta}_i^2 \\ & -\frac{1}{2}\sum_{i=1}^{N} k\left(\theta_i - \theta_{i-1}\right)^2 + \boldsymbol{\phi}(\mathbf{q})^\top \boldsymbol{\lambda}. \end{aligned} \tag{3.9}$$

The Lagrange multiplier vector $\boldsymbol{\lambda} = \begin{bmatrix} \mathbf{n}_1^\top & \dots & \mathbf{n}_N^\top \end{bmatrix}^\top$ is the set of internal link forces, and the constraint equations $\boldsymbol{\phi}(\mathbf{q}) = \begin{bmatrix} \boldsymbol{\phi}_1^\top & \dots & \boldsymbol{\phi}_N^\top \end{bmatrix}^\top = \mathbf{0}$ are given by

$$\boldsymbol{\phi}_i = h \begin{bmatrix} \cos\theta_i \\ \sin\theta_i \end{bmatrix} - \mathbf{p}_i + \mathbf{p}_{i-1}$$

for $i = 1...n$. The generalized forces $\mathbf{Q}$ consist of damping moments and externally applied forces and moments. These terms manifest in $\mathbf{F}$ as detailed below.

The $n \times n$ matrix $\mathbf{M}$ is diagonal and given by the Euler-Lagrange equation as

$$\mathbf{M} = \text{diag}(m, m, J, m, m, J, ..., m/2, m/2, J)$$

The matrix $\mathbf{A}$ is sparse and defined by the constraint equations. It can be built row by row by specifying only the nonzero entries. The submatrix of $\mathbf{A}$ consisting of rows $2i-1$ to $2i$ and columns $3i-5$ to $3i$ is given as:

$$\begin{bmatrix} 1 & 0 & 0 & -1 & 0 & -h\sin\theta_i \\ 0 & 1 & 0 & 0 & -1 & h\cos\theta_i \end{bmatrix}$$

for $i = 2...n$, and the corresponding rows $2i-1$ to $2i$ of the vector $\boldsymbol{\gamma}$ are

$$h\dot{\theta}_i^2 \begin{bmatrix} \cos\theta_i \\ \sin\theta_i \end{bmatrix}.$$

If the rod is cantilevered with a prescribed $\mathbf{p}_0(t)$, this implies a prescribed $\ddot{\mathbf{p}}_0(t)$, and the upper left 2×3 of $\mathbf{A}$ is

$$\begin{bmatrix} -1 & 0 & -h\sin\theta_1 \\ 0 & -1 & h\cos\theta_1 \end{bmatrix}$$

while the first 2 rows of $\boldsymbol{\gamma}$ are

$$h\dot{\theta}_1^2 \begin{bmatrix} \cos\theta_1 \\ \sin\theta_1 \end{bmatrix} - \ddot{\mathbf{p}}_0$$

Finally, rows $3i-2$ to $3i$ of the vector $\mathbf{F}$ are specified as

$$\begin{bmatrix} \mathbf{f}_i \\ k(\theta_{i-1} - 2\theta_i + \theta_{i+1}) + b(\dot{\theta}_{i-1} - 2\dot{\theta}_i + \dot{\theta}_{i+1}) + \ell_i \end{bmatrix}$$

for $i = 1...n-1$ (letting θ_0 have a prescribed value), while the last two rows of $\mathbf{F}$ corresponding to the distal end are

$$\begin{bmatrix} \mathbf{f}_N \\ k(\theta_{N-1} - \theta_N) + b(\dot{\theta}_{N-1} - \dot{\theta}_N) + \ell_i \end{bmatrix}.$$

This completes all the blocks defining the constrained Lagrangian form of the model in (3.8), consistent with the discretized Kirchhoff rod PDE in (3.4). The linear system (3.8) is sparse with a structure that can be efficiently solved in $O(n)$ time using standard linear solvers, such as MATLAB's mldivide( ). Without optimization the link accelerations are computed

on the order of 1 millisecond for $N = 40$ in MATLAB. As discussed before, for simulation purposes, the link angle accelerations are extracted from this solution, integrated in time, and used to recursively compute the positions and velocities. This avoids the constraint drift problem typically associated with maximal coordinate formulations.

Robot Actuation

It is straightforward to include typical soft robot actuation methods into the model above. In a tendon actuated robot, the k^{th} tendon exerts tension T_k at the point were it is distally attached (suppose at node i), offset from the elastic center of stiffness by some distance r_k. This creates a torque $\tau_k = T_k r_k$ applied as an external moment ℓ_i applied at node i, and added into the appropriate row of $\mathbf{F}$. As discussed in [68], the tendon also creates an applied force in the tangential direction at the attachment point, as well as a distributed load along the length, always orthogonal to the tangent and proportional to the curvature. However, *in the planar case*, the combined effect of these point and distributed tendon forces only influences the local axial component of internal force. They do not affect the local transverse component of internal force or the internal moment along the length, and thus do not affect curvature or dynamic shape. Thus, it is common (e.g. in [71]) to treat the actuation as consisting only of a point moment at the attachment point for simplicity, while understanding that the true internal force is simply the model predicted internal force minus the collective tendon tension in the axial direction. We note that in 3D, this simplification no longer holds [68], but that fact does not affect the applicability of the approach in this Chapter.

In a planar fluid-powered soft robot, a pressurized chamber exerts the same set of forces and moments as a tendon, but in the opposite direction. (the two models are slightly different in 3D, however, as detailed in [81].) Thus, for either a planar fluid-powered robot or a planar tendon-actuated robot, we can model the actuation of a pressurized chamber or a tensioned tendon as a point moment τ_k applied at its distal termination point along the robot, as is commonly done.

Full Model for Control

Considering then an arbitrary set of m actuation torques $\mathbf{u} \in \mathbb{R}^m$, we find that the full model for control takes the form

$$\begin{bmatrix} \mathbf{M} & \mathbf{A}^\top \\ \mathbf{A} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \ddot{\mathbf{q}} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{F} \\ \boldsymbol{\gamma} \end{bmatrix} + \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix} \mathbf{u} \tag{3.10}$$

where $\mathbf{B} \in \mathbb{R}^{3N \times m}$ is a sparse matrix with a one placed in element $(3i, k)$ if actuator k applies a torque at node i, and $\mathbf{u} \in \mathbb{R}^m$ is the vector of actuator torque inputs. We note that it is possible to eliminate $\boldsymbol{\lambda}$ from (3.10) and arrange it in a form similar to the classical rigid liked robot dynamics form, with a different right hand side due to the under actuation. However, the resulting analytical expressions are highly inconvenient for large N, and task-space controller design is easily facilitated in the form (3.10).

3.2 Force-Based Observer

Many controller designs require knowledge of the full robot state $\mathbf{q}$ and $\dot{\mathbf{q}}$. While high-resolution shape sensors can provide this information, typical low-cost continuum robots have much more limited sparse sensing. Suppose that we can only sense the position $\mathbf{p}$ and velocity $\dot{\mathbf{p}}$ at a subset of the model nodes (say k of them), and designate the set of sensed positions as $\mathbf{z} = [\dots \mathbf{p}_i^\top \dots]^\top \in \mathbb{R}^{2k}$ for all nodes i where measurements are available. A conceptually intuitive observer design simply augments a forward dynamics simulation with external forces that are controlled to drive the estimated positions to their measured values. Thus the observer takes the form

$$\begin{bmatrix} \mathbf{M} & \mathbf{A}(\tilde{\mathbf{q}})^\top \\ \mathbf{A}(\tilde{\mathbf{q}}) & \mathbf{0} \end{bmatrix} \begin{bmatrix} \ddot{\tilde{\mathbf{q}}} \\ \tilde{\boldsymbol{\lambda}} \end{bmatrix} = \begin{bmatrix} \mathbf{F}_0(\tilde{\mathbf{q}}, \dot{\tilde{\mathbf{q}}}) + \mathbf{B}\mathbf{u} + \mathbf{g} \\ \boldsymbol{\gamma}(\tilde{\mathbf{q}}, \dot{\tilde{\mathbf{q}}}) \end{bmatrix} \tag{3.11}$$

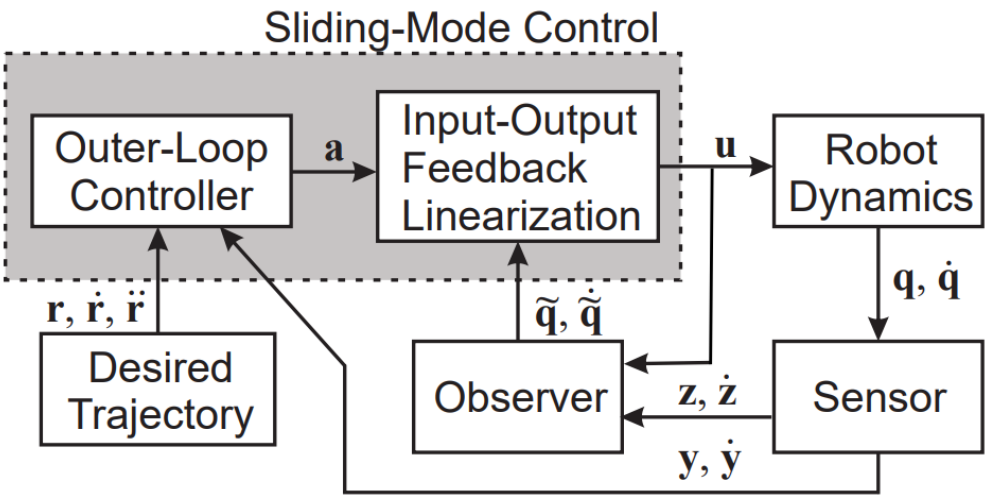


Figure 3.2: The overall structure of the control system.

where $\mathbf{g} \in \mathbb{R}^{3n}$ is a sparse vector containing a force $\mathbf{g}_i \in \mathbb{R}^2$ in elements $(3i-2:3i-1)$ whenever $\mathbf{p}_i$ and $\dot{\mathbf{p}}_i$ are measured by a sensor:

$$\mathbf{g}_i = L_d(\dot{\mathbf{p}}_i - \dot{\tilde{\mathbf{p}}}_i) + L_p\,(\mathbf{p}_i - \tilde{\mathbf{p}}_i)$$

with positive observer gains L_p, and L_d. This simple law is difficult to destabilize because it is passive, implementing a virtual spring and damper tied between the measured and estimated positions.

3.3 Controller

In this section we formulate an output control design based on the model of (3.10). Consider that we want to control an output $\mathbf{y} \in \mathbb{R}^m$ which is some linear function of $\mathbf{q}$, such that $\mathbf{y} = \mathbf{C}\mathbf{q}$ for some matrix $\mathbf{C} \in \mathbb{R}^{m\times 3N}$. Task space outputs are easily described in this way because the pose at the end-effector (or any other point) is included in $\mathbf{q}$.

Using the Lagrangian dynamics form, the second derivative of the output can then be expressed as a linear function of the actuation torques as follows:

$$\ddot{\mathbf{y}} = [\mathbf{C} \;\; \mathbf{0}] \begin{bmatrix} \ddot{\mathbf{q}} \\ \lambda \end{bmatrix} = \ddot{\mathbf{y}}_0 + \mathbf{J}\mathbf{u}$$

where

$$\ddot{\mathbf{y}}_0 = [\mathbf{C} \;\; \mathbf{0}] \begin{bmatrix} \mathbf{M} & \mathbf{A}^\top \\ \mathbf{A} & \mathbf{0} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{F} \\ \boldsymbol{\gamma} \end{bmatrix}$$

is the output acceleration that would occur in the absence of any actuation input, and

$$\mathbf{J} = [\mathbf{C} \;\; \mathbf{0}] \begin{bmatrix} \mathbf{M} & \mathbf{A}^\top \\ \mathbf{A} & \mathbf{0} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix}$$

is a $m \times m$ dynamic Jacobian matrix relating actuator torque to output acceleration. Note that computation of $\ddot{\mathbf{y}}_0$ and $\mathbf{J}$ does not require inversion of the large system matrix, but can be done with a much less expensive linear solve. The term

$$[\mathbf{C} \;\; \mathbf{0}] \begin{bmatrix} \mathbf{M} & \mathbf{A}^\top \\ \mathbf{A} & \mathbf{0} \end{bmatrix}^{-1}$$

can be efficiently computed with MATLAB's right matrix divide, or similar function, and is merely solving m sparse linear systems.

Feedback linearization of the output is then accomplished by the inner loop control law:

$$\mathbf{u} = \mathbf{J}^{-1}(\mathbf{a} - \ddot{\mathbf{y}}_0) \tag{3.12}$$

where $\mathbf{a}$ is symbolic of the desired output acceleration and serves as the new control input that will be specified by an outer loop control policy. Thus the feedback linearization transforms the nonlinear input-output dynamics into a set of decoupled double integrators.

Note that such feedback linearization can be done only if there is *strong inertial coupling* [77] between the actuation and the output. This corresponds to $\mathbf{J}$ being nominally nonsingular. Strong inertial coupling in this model arises from the inextensibility constraints of the Kirchhoff rod PDE, although $\mathbf{J}$ may become singular in certain configurations due to the robot's kinematic singularities. If $\mathbf{J}$ does loose rank in a particular configuration, we can replace the inverse in (3.13) with the Moore-Penrose pseudo-inverse. More generally, we can use a damped pseudo-inverse that will prevent large commanded actuation torques and

guard against ill-conditioning around any singularities:

$$\mathbf{u} = \left(\mathbf{J}^\top \mathbf{J} + \beta \mathbf{I}\right)^{-1} \mathbf{J}^\top (\mathbf{a} - \ddot{\mathbf{y}}_0) \tag{3.13}$$

where β is a small damping parameter and can be chosen to balance tracking accuracy against mitigation of singularities.

If $\mathbf{r}(t)$ is the reference we desire the output to track, then a outer-loop linear control law with feedforward acceleration:

$$\mathbf{a} = \ddot{\mathbf{r}} - k_d \dot{\mathbf{e}} - k_p \mathbf{e} \tag{3.14}$$

will drive the output error $\mathbf{e} = \mathbf{h} - \mathbf{r}$ asymptotically to zero, with second order behavior according to our choice of PD gains k_p and k_d. However, classical feedback linearization, when combined with such a linear outer control loop, is known to suffer from poor robustness to model parameter uncertainty [30]. Our preliminary simulations showed that with this approach, even very small changes to the modelled Young's modulus caused the controlled continuum robot system to become unstable.

A sliding mode control approach can greatly improve robustness while preserving the basic structure and advantages of the input-output feedback linearization above. We define a sliding variable $\mathbf{s}$ in terms of the output error as follows:

$$\mathbf{s} = \dot{\mathbf{e}} + \alpha \mathbf{e}$$

for $\alpha > 0$. Thus, driving $\mathbf{s}$ to zero eventually enforces stable first-order output error dynamics with time constant $1/\alpha$. To derive a control which robustly drives $\mathbf{s}$ to zero, consider the Lyapunov function

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

Its derivative

$$V = \mathbf{s}^\top \dot{\mathbf{s}}$$

is forced to be negative definite (even under some finite disturbance or parametric error in the model) if we can choose the control input to achieve $\dot{\mathbf{s}} = -k \operatorname{sgn} \mathbf{s}$, where k is a gain

related to the expected amount of disturbance, and sgn takes the element-wise sign of a vector. So sliding mode control should enforce

$$\dot{\mathbf{s}} = \ddot{\mathbf{y}} - \ddot{\mathbf{r}} + \alpha\dot{\mathbf{e}} = -k\,\mathrm{sgn}\left(\dot{\mathbf{e}} + \alpha\mathbf{e}\right)$$

Since we can dictate $\ddot{\mathbf{y}} = \mathbf{a}$ using the input-output feedback linearization above, the sliding mode controller is then equivalent to replacing the linear outer loop control (3.14) with the nonlinear control law

$$\mathbf{a} = \ddot{\mathbf{r}} - \alpha\dot{\mathbf{e}} - k\,\mathrm{sgn}\left(\dot{\mathbf{e}} + \alpha\mathbf{e}\right) \tag{3.15}$$

The well-studied "chattering" phenomenon sometimes caused by the discontinuity in the $\mathrm{sgn}\left(\dot{\mathbf{e}} + \alpha\mathbf{e}\right)$ function can be eliminated by various means (conventionally deadband, boundary layer, etc.). In our implementation we simply replace sgn with a continuous approximation, such as the Gauss error function $\mathrm{erf}\left(c(\dot{\mathbf{e}} + \alpha\mathbf{e})\right)$ where c encodes the steepness of the transition from -1 to 1.

Since the open loop system is passive, consisting of only of inertial, capacitive, and dissipative elements, the system's zero dynamics are stable. The transient oscillations of the zero dynamics will decay at their natural rate, and any steady-state oscillations will simply consist of the passive system's response to the frequency content of the reference trajectory, and the controller additionally stabilizes the output error.

The overall structure of the control approach can be combined in the following manner illustrated in the block diagram in Figure 3.2. The sliding mode control consists of the nonlinear outer loop controller (3.15) with the input-output feedback linearization (3.13). The feedback linearization relies on the full state estimate from the observer (4.19) based on all of the sensed data $\mathbf{z}$ while the outer loop controller relies only on sensed output $\mathbf{y}$

3.4 Experiments

We tested our nonlinear controller/observer combination numerically in simulation using a model of a soft robot. In our model, the actuation manifests as a torque τ_1 applied at the distal tip, and another τ_2 applied at the midpoint. We identify realistic parameters for a

Table 3.1: Simulated Model Parameters

EI (Nm^2)	BI (Nm^{2}s)	ρA (kg/m)	ρI (kg m)	L (m)	N
0.14	0.003	0.8	0	0.275	20

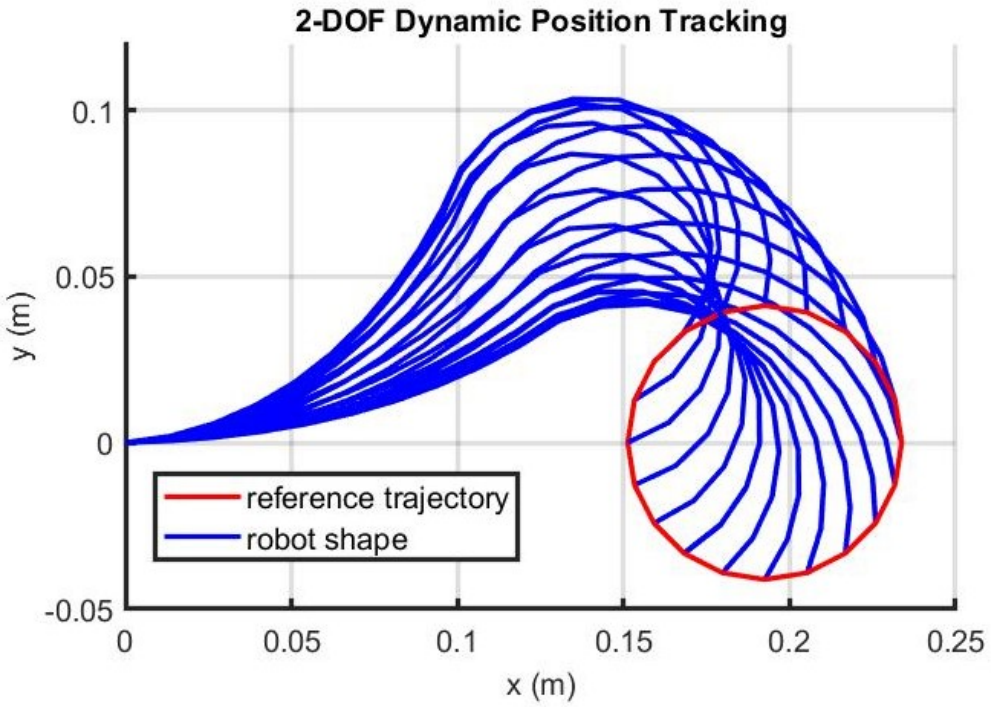


Figure 3.3: Snapshots of the dynamic robot behavior during the circle task after the tracking error has converged to zero (after 0.2 seconds).

soft robot made from Ninjaflex TPU. The parameters are listed in Table I. Note that while ρI is not really zero, it is small for most slender robots, and the model can be calibrated accurately enough assuming $\rho I = 0$. (Note that this causes no problem in the model. Even though $\mathbf{M}$ is singular, the larger system matrix in (3.10) is still full rank).

We implemented our controller/observer approach in Simulink, and the parameters used in the outer-loop controller and observer are listed in Table II. While the feedback linearization block and the observer assumed the model parameters in Table I, the simulated robot plant model used different parameters to simulate parametric modeling error and

Table 3.2: Controller Parameters

β	α	k	c	L_p	L_d
0.1	40	80	40	0.5	0.1

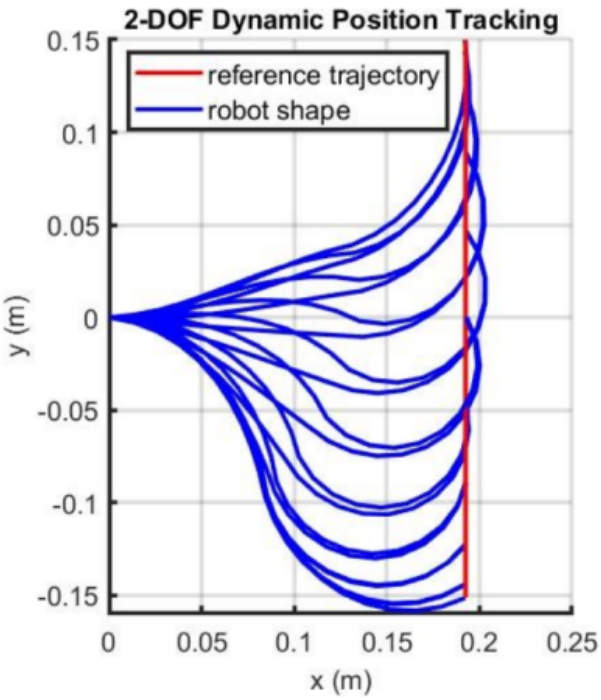


Figure 3.4: Snapshots of the dynamic robot behavior during the line following task after the tracking error has converged to zero (after 0.2 seconds).

assess controller robustness. The simulated robot had a 10% greater density and 10% lower stiffness and damping constants than the controller and observer assumed. The only output measurement used in the controller and observer was the tip position and velocity (i.e. $\mathbf{z} = \mathbf{y} = \mathbf{p}_N$), representing a very minimal sensing scenario, and the observer initial conditions were set differently than the robot model. Thus these numerical experiments simulate a practical scenario with limited sensor information, unknown initial conditions, and non-negligible model inaccuracy.

We performed two reference tracking experiments where the robot is commanded to follow a tip trajectory in space. The reference trajectories were designed to be dynamically significant by containing frequency content in the range of the first two modal frequencies of the underactuated robot dynamics (approx. 3.5 Hz and 19 Hz). The first trajectory of tracking a point moving around the red circle shown in Figure 3.3 at a frequency of 5 Hz. The second task is to follow a point moving sinusoidally back and forth along the red line (again at 5 Hz) as shown in Figure 3.4.

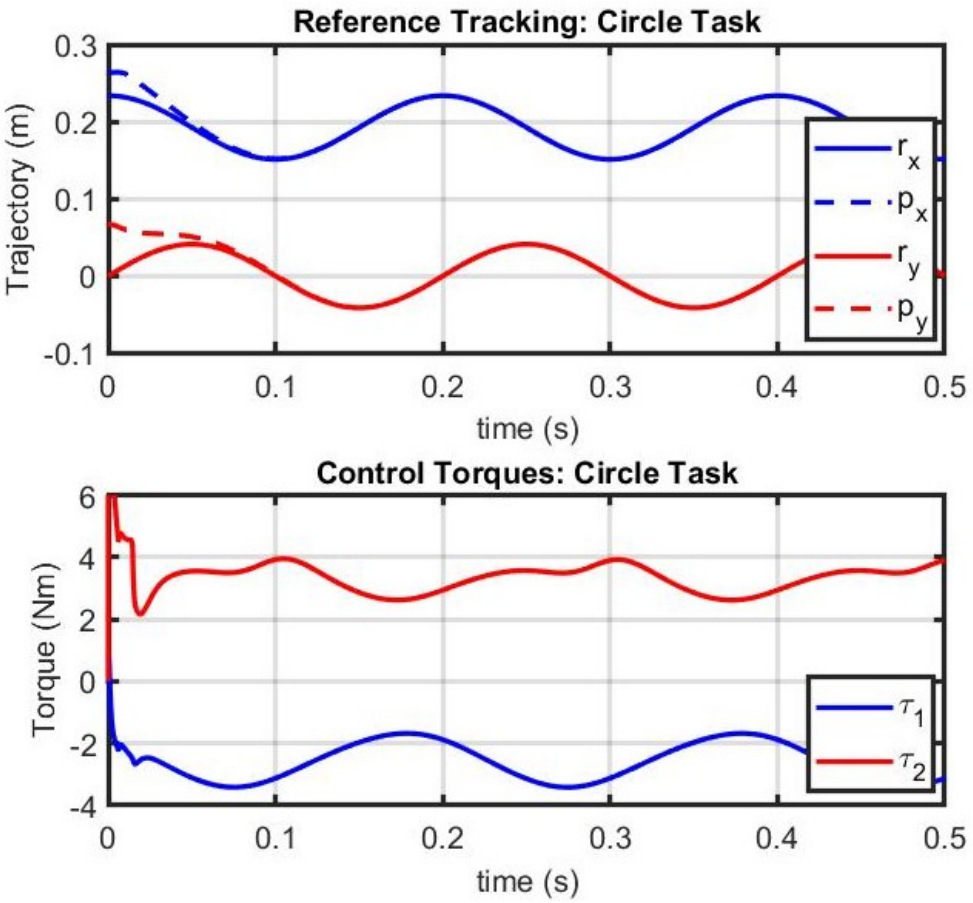


Figure 3.5: Top: The desired reference trajectory is compared to the robot output for the circle tracking task. Bottom: The control torques indicate that significant inertial dynamics are being overcome.

Results

Figures 3.3 and 3.4[1] show the dynamic robot behavior executing each task once the tracking error has converged to zero (i.e. after about 0.2 seconds). Figures 3.5 and 3.6 detail the error convergence behavior and the actuator torques commanded by the sliding mode controller. The state estimation error does not converge to zero because of the model differences between the observer and the robot, but the reference tracking error goes to zero in spite of this due to the robustness of the sliding mode controller.

The snapshots in Figure 3.4 are timed to show the same task-space points on subsequent passes moving left and right. The fact that there are several pairs of snapshots with the

[1] © 2022 IEEE

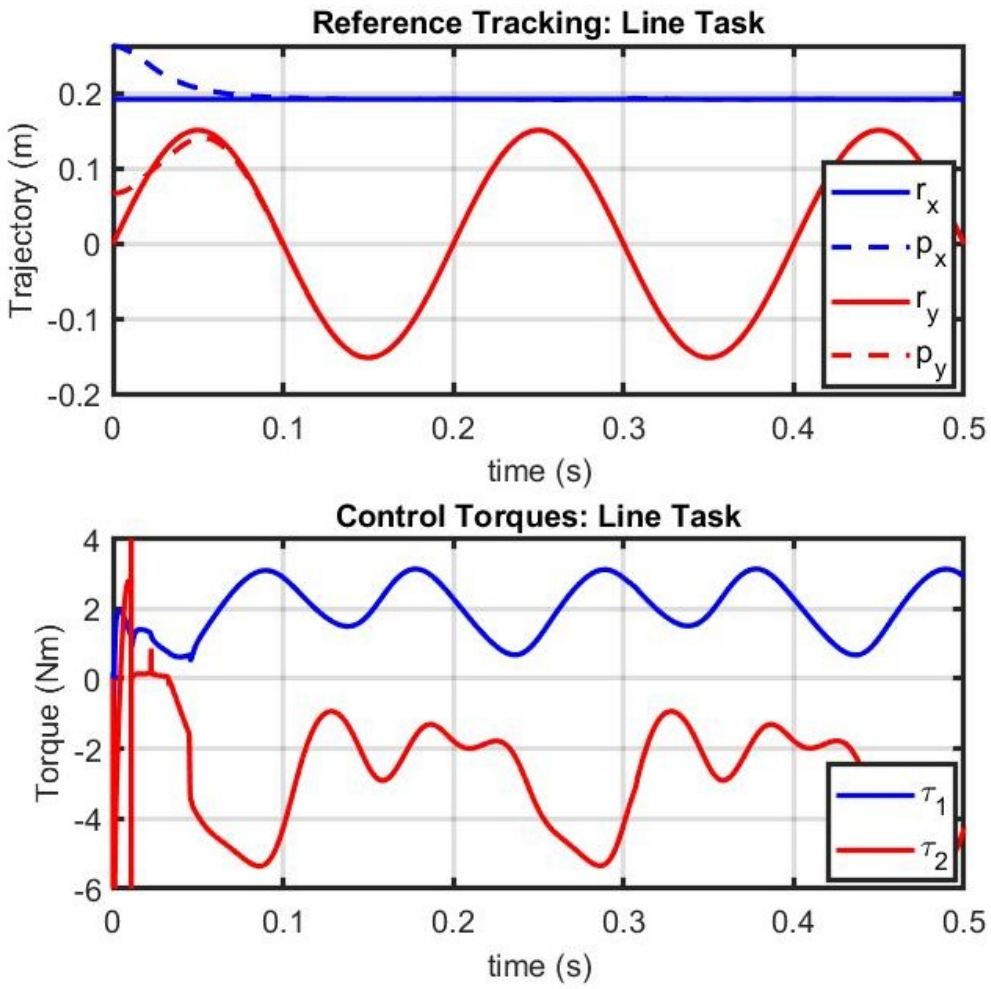


Figure 3.6: Top: The desired reference trajectory is compared to the robot output for the cyclical line tracking task. Bottom: The control torques indicate that significant inertial dynamics and couplings between the actuated DOFs are being overcome.

same end-effector location but different body deformations indicates the significance of the inertial dynamics in this task. The presence of non-sinusoidal torque signals in Figures 3.5 and 3.6 also indicate that significant inertial dynamics and couplings between the actuated DOFs are being overcome.

Chapter 4

3D Dynamics

In this chapter we build upon the work shown in chapter 3 by expanding the model and observer to 3D. The work in this chapter is published in [24]. This section is a review of that work and details the formulation and validation of a force based state estimator. We start by describing the continuous model used in this approach.

4.1 3D Modeling

The dynamic equilibrium of a Cosserat rod is classically described by a set of partial differential equations (PDEs) expressing dynamic equilibrium in a differential form. Following [2], this model is straightforward to derive based on a Newtonian equilibrium analysis on a finite section of rod (i.e. the sum of forces is equal to the time derivative of linear momentum, and the sum of moments is equal to the time derivative of angular momentum). Considering, as in Figure 4.1, a section from arc length c to s, and taking the moments and angular

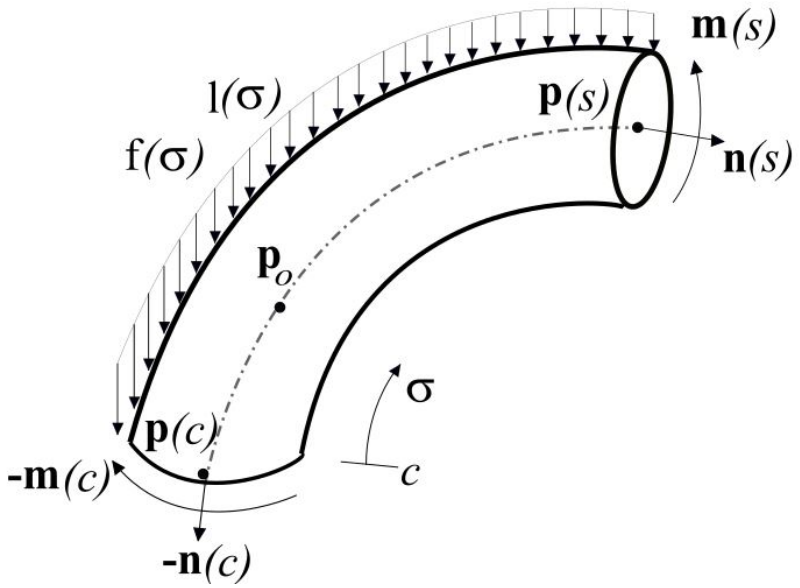


Figure 4.1: An arbitrary section of a rod is indicated from c to s along σ. The distributed forces and moments are represented as $\mathbf{f}$ and $\mathbf{l}$ respectively. The internal forces and moments are represented as $\mathbf{n}$ and $\mathbf{m}$ respectively. Three positions shown by $\mathbf{p}$ are indicated along the center-line of the rod at the proximal end c, the distal end s, and an arbitrary point $\mathbf{p}_o$.

momentum about an arbitrary point $\mathbf{p}_o$ we have

$$\mathbf{n}(s) - \mathbf{n}(c) + \int_c^s \mathbf{f}(\sigma) d\sigma = \frac{d}{dt} \int_c^s \rho A \dot{\mathbf{p}}(\sigma) d\sigma$$

$$\begin{aligned} \mathbf{m}(s) - \mathbf{m}(c) + (\mathbf{p}(s) - \mathbf{p}_o) \times \mathbf{n}(s) & \\ - (\mathbf{p}(c) - \mathbf{p}_o) \times \mathbf{n}(c) + \int_c^s \mathbf{l}(\sigma) d\sigma & \\ = \frac{d}{dt} \int_c^s \left(\rho A \left(\mathbf{p}(\sigma) - \mathbf{p}_o \right) \times \dot{\mathbf{p}}(\sigma) + \rho \mathbf{J}^g \mathbf{w} \right) d\sigma & \end{aligned} \tag{4.1}$$

where $\mathbf{n}$ is the internal force, $\mathbf{m}$ is the internal moment, $\mathbf{p}$ is the position, $\mathbf{w}$ is the angular velocity, $\mathbf{f}$ and $\mathbf{l}$ are external distributed force and moment vectors (per unit length), ρ is the material density, A is the cross section area, and $\mathbf{J}^g = \mathbf{R}\mathbf{J}\mathbf{R}^\top$ is the second moment of area matrix of the cross section (in the global frame), computed from $\mathbf{J}$, the (constant) second moment of area matrix in body frame and $\mathbf{R}$ the material orientation. The gravitational terms are contained in $\mathbf{f}$. All vectors are expressed in the inertial global frame and are functions of arc length s and time t. Derivatives with respect to time are denoted by $\dot{}$.

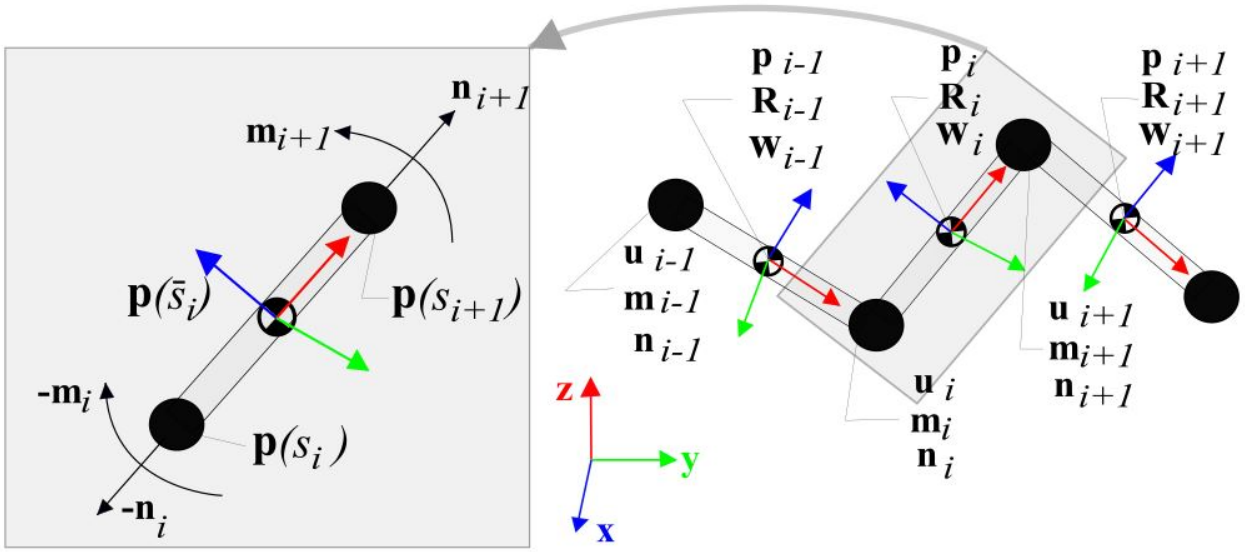


Figure 4.2: To approximate an arbitrary continuous rod segment, a series of links and nodes can be connected in series via internal forces $\mathbf{n}$ and moments $\mathbf{m}$ to formulate a discrete elastic rod model. The internal forces, moments, and curvatures $\mathbf{u}$ can naturally reside on the nodes indicated by black dots. Alternatively, the positions $\mathbf{p}$, orientation frames $\mathbf{R}$, and angular velocities ω can be indicated on the links. We can let the i^{th} link lie in between the i^{th} and the $i+1^{th}$ node.

Taking the derivative of (4.1) with respect to arc length s (denoted with $'$), the result can be algebraically reduced to the classical differential form below:

$$\begin{aligned} \mathbf{n}' + \mathbf{f} &= \rho A \ddot{\mathbf{p}} \\ \mathbf{m}' + \widehat{\mathbf{p}}'\mathbf{n} + \mathbf{l} &= \frac{d}{dt}(\rho \mathbf{J}^g \mathbf{w}) \end{aligned} \tag{4.2}$$

where following [52], throughout the chapter we employ $\widehat{\ }$ or $\wedge$ to denote the skew-symmetric matrix form of a vector, such that $\mathbf{a} \times \mathbf{b} = \widehat{\mathbf{a}}\mathbf{b} = \mathbf{a}^{\wedge}\mathbf{b}$. We also use $\vee$ to denote the inverse of $\widehat{\ }$, i.e. $\widehat{\mathbf{a}}^{\vee} = \mathbf{a}$

The global frame angular velocity $\mathbf{w}$ defines the time derivative of the rotation as

$$\dot{\mathbf{R}} = \widehat{\mathbf{w}}\mathbf{R} \tag{4.3}$$

whereas the evolution of $\mathbf{R}$ along s is described by the curvature $\mathbf{u}$ as

$$\mathbf{R}' = \mathbf{R}\hat{\mathbf{u}} \tag{4.4}$$

For notational convenience, we refer to the first, second, and third columns of $\mathbf{R}$ as $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{z}$, respectively.

The internal force and moment must be related to material deformation by a constitutive law. We employ a linear visco-elastic law of the form

$$\mathbf{m} = \mathbf{R}\left(\mathbf{K}\mathbf{u} + \mathbf{D}\dot{\mathbf{u}}\right) \tag{4.5}$$

where $\mathbf{K} \in \mathbb{R}^{3\times3}$ is the elastic stiffness matrix whose elements are composed of the elastic stiffness about $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{z}$. We choose the stiffness matrix to be diagonal. By doing this, we assume there are no coupling effects between the stiffness of different degrees of freedom. Likewise, $\mathbf{D} \in \mathbb{R}^{3\times3}$ is a viscous damping matrix whose elements represent the viscus damping about $\mathbf{x}$, $\mathbf{y}$, and $\mathbf{z}$. $\mathbf{D}$ is similarly chosen to be a diagonal matrix.

A consistent representation of the curvature derivative, $\dot{\mathbf{u}}$, can be found by taking the time derivative of (4.4). Isolating for $\dot{\hat{\mathbf{u}}}$ gives:

$$\dot{\hat{\mathbf{u}}} = \dot{\mathbf{R}}^T\mathbf{R}' + \mathbf{R}^T\dot{\mathbf{R}}' \tag{4.6}$$

By substituting (4.3) and (4.4) into (4.6), we get the following relationship between curvature and angular velocity:

$$\dot{\mathbf{u}} = \mathbf{R}^\top\mathbf{w}' \tag{4.7}$$

In the Kirchhoff case, we have a kinematic constraint on the evolution of $\mathbf{p}$ along the arc length:

$$\mathbf{p}' = \mathbf{R}\mathbf{e}_3 = \mathbf{z}$$

where $\mathbf{e}_3 = [0 \;\; 0 \;\; 1]^\top$. From a constrained dynamics perspective, the internal force vector $\mathbf{n}$ functions as a Lagrange multiplier responsible for maintaining this constraint.

Discretization

We would like to employ a discretization of the above equations in the arc length dimension and obtain a set of index-3 DAE's. One common strategy is to discretize the spatial derivatives in (4.2) with symmetric first-order discrete approximations. However, this approach does not correctly reproduce the dynamic equilibrium of a finite section of rod. To remedy this, we turn back to (4.1), the integral form of the model. As we show below, discretizing (4.1) itself is a better strategy because the discretized equations will retain the higher order inertial term that correctly accounts for the distribution of mass over a finite length, whereas that term is not recoverable by direct finite-difference approximation of (4.2), which effectively lumps mass into a single point on the interval. To perform discretization, we simply assume a constant orientation $\mathbf{R}_i$ over a segment with finite length h. Without loss of generalization, the arc lengths at the left, right, and midpoint of this segment are labeled s_i, s_{i+1}, and $\bar{s}_i$, and we take the moment reference point $\mathbf{p}_o$ to be the segment midpoint, labeled $\mathbf{p}(\bar{s}_i) = \mathbf{p}_i$. Thus, the position over the segment is

$$\mathbf{p}(\sigma) = \mathbf{p}_i + \mathbf{R}_i \mathbf{e}_3 (\sigma - \bar{s})$$

Substituting this segment shape assumption to the integral form of the Cosserat dynamic equilibrium above ultimately yields the following discrete equations:

$$\begin{aligned} \mathbf{n}_{i+1} - \mathbf{n}_i + h\mathbf{f}_i &= \mu_i \ddot{\mathbf{p}}_i \\ \mathbf{m}_{i+1} - \mathbf{m}_i + \frac{h}{2}\widehat{\mathbf{z}}_i (\mathbf{n}_i + \mathbf{n}_{i+1}) + h\mathbf{l}_i &= \frac{d}{dt}(\mathbf{I}_i \mathbf{w}_i) \end{aligned} \tag{4.8}$$

where the mass $\mu_i = \rho A h$, and the (global frame) rotational inertia is given by

$$\mathbf{I}_i = \mathbf{R}_i \left(\rho h \mathbf{J} + \frac{1}{12} \rho A h^3 \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \right) \mathbf{R}_i^\top$$

$$\mathbf{I}_i = \mathbf{R}_i \mathbf{I}_i^b \mathbf{R}_i^\top$$

where $\mathbf{I}_i^b$ is the (constant) body frame rotational inertia of the discrete segment. Observe that the rotational inertia term $\frac{1}{12}\rho A h^3$ arises from the mass distribution along the segment

length. This term would not have appeared if we had discretized (4.2) directly using a first-order difference formula because it is third-order in h. Comparing the inertia terms, we see that this term is a significant portion of the total rotational inertia of a segment about its midpoint. For a rod with solid circular cross section, the term accounts for over half of the local x, y rotational inertia when the discretization is composed of segments with length twice the size of the radius ($h = 2r$) (which is very fine), and its portion grows rapidly larger with h. Thus, while the effect is less for exceedingly small h, this term may have a significant effect on natural frequencies.

Expanding the right hand side derivative of the moment equilibrium then gives:

$$\frac{d}{dt}(\mathbf{I}_i\mathbf{w}_i) = \widehat{\mathbf{w}}_i\mathbf{I}_i\mathbf{w}_i + \mathbf{I}_i\dot{\mathbf{w}}_i \tag{4.9}$$

The Kirchhoff kinematic constraint is then discretized symmetrically and multiplied by h to obtain the i^{th} discrete constraint at arc length s_i:

$$\mathbf{p}_i - \mathbf{p}_{i-1} = \frac{h}{2}\left(\mathbf{z}_{i-1} + \mathbf{z}_i\right) \tag{4.10}$$

The geometric interpretation of this symmetric discretization is that internal force and moment quantities with subscript i are located at arc length s_i, while kinematic quantities ($\mathbf{p}$, $\mathbf{R}$, $\mathbf{w}$), external loads, and inertias with subscript i are all associated with arc length $s_i + h/2$. This is depicted in the diagram in Figure 4.2. We also note that this particular discretization produces dynamic equations and constraints that equivalently represent a chain of n rigid-body links connected by joints that carry forces and moments. In this interpretation, we see that (4.10) represents exactly the joint constraints of the multibody system.

The internal moments are represented using a visco-elastic constitutive law in (4.5). When symmetrically discretized we obtain:

$$\mathbf{m}_i = \frac{1}{2}\left(\mathbf{R}_{i-1} + \mathbf{R}_i\right)\left(\mathbf{K}_i\mathbf{u}_i + \mathbf{D}_i\dot{\mathbf{u}}_i\right)$$

A discretized representation of curvature can be found by the change in orientation over the segment length h. For a symmetric discretization, this can be computed by considering the

rotation matrix that brings frame $i-1$ to frame i. Thus, the curvature represented at the i^{th} node is

$$\mathbf{u}_i = \frac{\log(\mathbf{R}_{i-1}^\top \mathbf{R}_i)^\vee}{h}$$

Because $\mathbf{R}_{i-1}^\top \mathbf{R}_i$ belongs to $SO(3)$, the matrix logarithmic expression can be computed with common approaches such as Rodrigues' formula. Alternatively, a more computationally efficient solution can be obtained by approximating the matrix logarithm. Consider that the logarithm of an orthonormal matrix can be expressed as

$$\mathbf{u}_i = \begin{cases} 0 & \theta_i = 0 \\ (\frac{\theta_i}{h \sin(\theta_i)} \mathbf{Q}_i)^\vee & \theta_i \neq 0 \end{cases}$$

where θ_i is the angle about the rotational axis and $\mathbf{Q}_i = \frac{1}{2}\left(\mathbf{R}_{i-1}^\top \mathbf{R}_i - \mathbf{R}_i^\top \mathbf{R}_{i-1}\right)$. Given that large deformations are often distributed across numerous links making up the robot model (resulting in small orientational differences between neighboring frames), a small angle approximation is valid for most circumstances. Using this knowledge, we can arrive at the following computationally efficient curvature approximation:

$$\mathbf{u}_i \approx \frac{1}{2h}\left(\mathbf{R}_{i-1}^\top \mathbf{R}_i - \mathbf{R}_i^\top \mathbf{R}_{i-1}\right)^\vee \tag{4.11}$$

Applying our symmetric discretization approach to (4.7), the discretized expression for $\dot{\mathbf{u}}_i$ can be written as:

$$\dot{\mathbf{u}}_i = \left(\frac{\mathbf{R}_i + \mathbf{R}_{i-1}}{2}\right)^\top \left(\frac{\mathbf{w}_i - \mathbf{w}_{i-1}}{h}\right)$$

The calculation for curvature and its time derivative is dependent on two adjacent orientation frames ($\mathbf{R}_i$ and $\mathbf{R}_{i-1}$) indicated in Figure 4.2. When $i = 1$, we can assign a base link frame $\mathbf{R}_{i-1} = \mathbf{R}_0$ located at $\mathbf{p}_0$. When simulating fixed base conditions, both $\mathbf{R}_0$ and $\mathbf{p}_0$ are constants. Similarly, when calculating the moment at the n^{th} node, we can set $\mathbf{m}_{i+1}$ equal to zero, as there is no moment resulting from a $n+1$ link.

Kirchhoff Constraints and Index Reduction

The equations above form a set of index-3 ODEs. We reduce the index to 1 by taking two time derivatives of the discretized Kirchhoff kinematic constraints as follows:

$$\begin{aligned}\dot{\mathbf{p}}_i - \dot{\mathbf{p}}_{i-1} =& \frac{h}{2}\left(\widehat{\mathbf{w}}_{i-1}\mathbf{z}_{i-1} + \widehat{\mathbf{w}}_i\mathbf{z}_i\right) \\ \ddot{\mathbf{p}}_i - \ddot{\mathbf{p}}_{i-1} =& \frac{h}{2}\big(\widehat{\mathbf{w}}_{i-1}^2\mathbf{z}_{i-1} + \widehat{\mathbf{w}}_i^2\mathbf{z}_i \\ & - \widehat{\mathbf{z}}_{i-1}\dot{\mathbf{w}}_{i-1} - \widehat{\mathbf{z}}_i\dot{\mathbf{w}}_i\big)\end{aligned} \tag{4.12}$$

The equation representing the constraint of the i^{th} joint can then be represented in the form

$$\mathbf{A}_i\dot{\mathbf{v}} = \boldsymbol{\gamma}_i$$

for $i = 2...n$ where $\mathbf{v} = [\dot{\mathbf{p}}_1\ \mathbf{w}_1\ \dot{\mathbf{p}}_2\ \mathbf{w}_2\ \ldots]^\top$ and its time derivative $\dot{\mathbf{v}} = [\ddot{\mathbf{p}}_1\ \dot{\mathbf{w}}_1\ \ddot{\mathbf{p}}_2\ \dot{\mathbf{w}}_2\ \ldots]^\top$ is the vector containing all acceleration variables. $\mathbf{A}_i$ is a $3\times 6n$ matrix with zeros everywhere except columns $6i-11$ to $6i$ for $i = 2...n$ which are

$$\begin{bmatrix} -\mathbb{I} & \frac{h}{2}\widehat{\mathbf{z}}_{i-1} & \mathbb{I} & \frac{h}{2}\widehat{\mathbf{z}}_i \end{bmatrix}$$

(where $\mathbb{I}$ is the 3×3 identity matrix) and the right hand side is

$$\boldsymbol{\gamma}_i = \frac{h}{2}\big(\widehat{\mathbf{w}}_{i-1}^2\mathbf{z}_{i-1} + \widehat{\mathbf{w}}_i^2\mathbf{z}_i\big)$$

Stacking all the constraint equations together gives

$$\mathbf{A}\dot{\mathbf{v}} = \boldsymbol{\gamma}$$

where

$$\mathbf{A} = \begin{bmatrix}\mathbf{A}_1 \\ \mathbf{A}_2 \\ \vdots\end{bmatrix}, \quad \boldsymbol{\gamma} = \begin{bmatrix}\boldsymbol{\gamma}_1 \\ \boldsymbol{\gamma}_2 \\ \vdots\end{bmatrix}$$

So far, the above equations represent only the "joint constraints" between discrete links. If other constraints are present, such as connection to a fixed or moving base, or prescribed tip position, these are treated the same way by taking the second derivative, writing it in the above form, and concatenating an additional block of rows into $\mathbf{A}$ and $\boldsymbol{\gamma}$ (at either the top or the bottom). For a general constraint equation $\boldsymbol{\phi}_i(\mathbf{p}_1, \mathbf{R}_1, \mathbf{p}_2, \mathbf{R}_2, ...) = \mathbf{0}$, $\mathbf{A}_i$ simply contains the coefficients of any of the accelerations in $\dot{\mathbf{v}}$ that happen to be present in $\boldsymbol{\phi}_i$, while $\boldsymbol{\gamma}_i$ contains all other terms remaining on the right hand side. For example, under fixed base conditions columns 1 to 3 of $\mathbf{A}_1$ are $\begin{bmatrix} \mathbb{I} & \frac{h}{2}\widehat{\mathbf{z}}_i \end{bmatrix}$. Every other element of $\mathbf{A}_1$ is zero. The right hand side of the equation when $i = 1$ is $\boldsymbol{\gamma}_1 = \frac{h}{2}\left(\widehat{\mathbf{w}}_i^2 \mathbf{z}_i\right)$. There are a minimum of $n - 1$ constraint equations (representing the "joint" connections between the n "links"), plus any context-dependent constraints.

Baumgarte Stabilization:

The model's constraints act on acceleration, which can lead to constraint drift due to numerical integration. To enforce constraints, Baumgarte's stabilization [6] adds corrective terms to $\boldsymbol{\gamma}$.

We begin by expressing the position and velocity constraints ($\boldsymbol{\phi}$ and $\dot{\boldsymbol{\phi}}$). The i^{th} constraint in $\boldsymbol{\phi}$ is:

$$\boldsymbol{\phi}_i = \mathbf{p}_i - \mathbf{p}_{i-1} - \frac{h}{2}\left(\mathbf{z}_{i-1} + \mathbf{z}_i\right)$$

The time derivative of $\boldsymbol{\phi}$ is:

$$\dot{\boldsymbol{\phi}} = A\mathbf{v}$$

When fully enforced, both $\boldsymbol{\phi}$ and $\dot{\boldsymbol{\phi}}$ are zero. Thus, these terms represent errors in constraint enforcement. The modified constraint equation is:

$$A\dot{\mathbf{v}} = \underbrace{-\dot{A}\mathbf{v} - 2\zeta\omega_N A\mathbf{v} - \omega_N^2 \boldsymbol{\phi}}_{=\boldsymbol{\gamma}}. \tag{4.13}$$

Here, ω_N and ζ control the convergence rate and damping ratio, respectively. These gains are selected to ensure stability and fast convergence, typically chosen to be much larger than the system's dynamics. In our case, we choose them to be $\zeta = 2$ and $\omega_N = 150$.

Symmetric Form

By a detailed inspection we find that $\mathbf{A}$ also appears in the set of dynamic equations as particularly written in (4.8). This is unsurprising, as the model could have been derived from a constrained Lagrangian perspective, which is known to have this form. Concatenating all link dynamic equations together we get

$$\mathbf{M}\dot{\mathbf{v}} + \mathbf{A}^\top \boldsymbol{\lambda} = \mathbf{F} \tag{4.14}$$

Where $\mathbf{M}$ is symmetric and block diagonal:

$$\mathbf{M} = \operatorname{diag}\left(\mu_1 \mathbb{I}, \mathbf{I}_1, \mu_2 \mathbb{I}, \mathbf{I}_2, \dots\right)$$

$\mathbf{F}$ is a vector containing all forces and moments not arising from constraints:

$$\mathbf{F} = \begin{bmatrix} h\mathbf{f}_1 \\ h\mathbf{l}_1 + \mathbf{m}_2 - \mathbf{m}_1 - \widehat{\mathbf{w}}_1 \mathbf{I}_1 \mathbf{w}_1 \\ h\mathbf{f}_2 \\ h\mathbf{l}_2 + \mathbf{m}_3 - \mathbf{m}_2 - \widehat{\mathbf{w}}_2 \mathbf{I}_2 \mathbf{w}_2 \\ \vdots \end{bmatrix}$$

and $\boldsymbol{\lambda}$ is the vector of constraint forces, containing (at least) all joint forces $\mathbf{n}_2$ through $\mathbf{n}_n$, plus any forces or moments associated with context-dependent constraints that may be represented in $\mathbf{A}$. In total, this way of writing the equations can now be seen as a symmetric system where the unknowns are all accelerations and constraint forces:

$$\begin{bmatrix} \mathbf{M} & \mathbf{A}^\top \\ \mathbf{A} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{v}} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{F} \\ \boldsymbol{\gamma} \end{bmatrix} \tag{4.15}$$

Writing it this way is convenient from a programming perspective, reduces opportunities for errors, and can be leveraged for computational efficiency due to the maturity of solvers for symmetric linear systems, including a block tri diagonal matrix solver that scales linearly

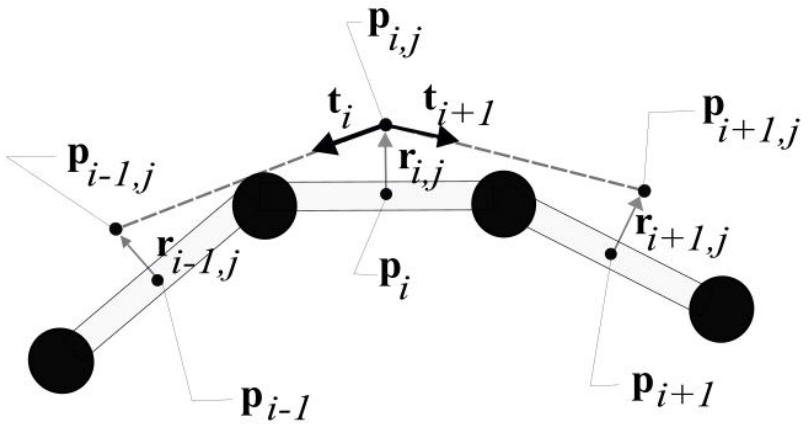


Figure 4.3: The j^{th} actuator input can be seen acting on the backbone of the robot. The j^{th} actuator tendon passes along guide channels located $r_{i,j}$ away from the i^{th} link frame. At the i^{th} tendon guide channel, position of the j^{th} tendon is indicated as $\mathbf{p}_{i,j}$. The unit vector describing the direction of the tendon tension from $\mathbf{p}_{i,j}$ to $\mathbf{p}_{i-1,j}$ is described by $\mathbf{t}_i$. Likewise, the unit vector describing the direction of the tendon tension from $\mathbf{p}_{i,j}$ to $\mathbf{p}_{i+1,j}$ is described by $\mathbf{t}_{i+1}$.

with the amount of links [58]. We note that this symmetric form is exactly what would have arisen if we had taken a constrained Lagrangian approach to the derivation as shown in [63].

Actuation modeling

We use a similar approach to tendon modeling as described in [68, 18]. To model tendon forces, we consider n_τ actuator inputs, where the j^{th} actuator corresponds to a tendon running through a guiding channel defined by $\mathbf{r}_{i,j}$ relative to the i^{th} link frame, as shown in figure 4.3. The tendon's position at the i^{th} link is given by:

$$\mathbf{p}_{i,j} = \mathbf{p}_i + \mathbf{R}_i \mathbf{r}_{i,j}$$

Since tendon forces only act axially, the unit vector describing the tendon's direction from the i^{th} link to the $(i+1)^{th}$ link is:

$$\mathbf{t}_{i+1,j} = \frac{\mathbf{p}_{i+1,j} - \mathbf{p}_{i,j}}{||\mathbf{p}_{i+1,j} - \mathbf{p}_{i,j}||}$$

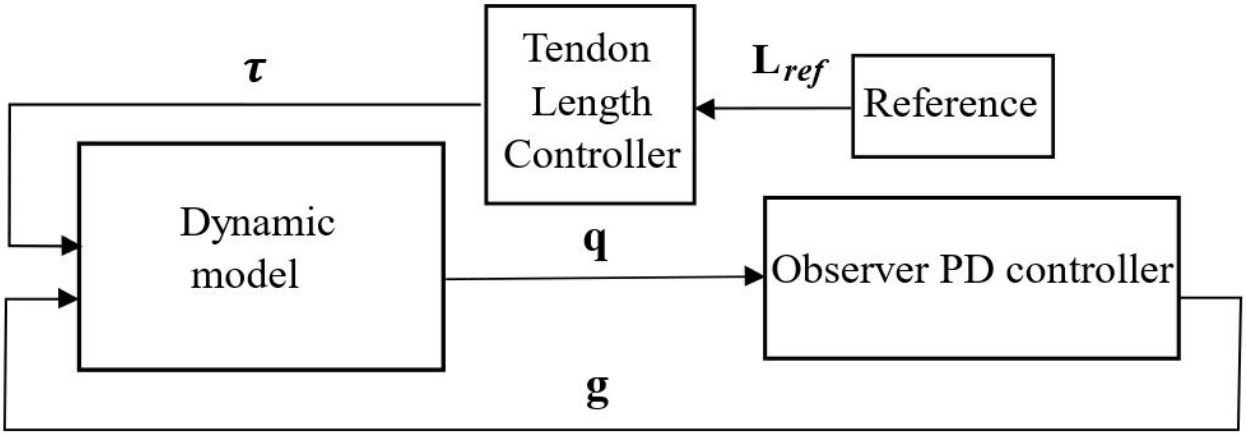


Figure 4.4: This figure shows a block diagram of the observer structure. The tendon length controller can use ether of the methods detailed in the "Tendon Actuation via Displacement Commands" subsection. $\mathbf{L}_{ref}$ is the reference tendon length, $\boldsymbol{\tau}$ is the actuation tension of the tendons, $\mathbf{q}$ is the state vector, and $\mathbf{g}$ is the observer forces and moments

Similarly, from the i^{th} link to the $(i-1)^{th}$ link is:

$$\mathbf{t}_{i,j} = \frac{\mathbf{p}_{i,j} - \mathbf{p}_{i-1,j}}{||\mathbf{p}_{i,j} - \mathbf{p}_{i-1,j}||}$$

The tendon position differences between adjacent links are:

$$\left(\mathbf{p}_{i+1,j} - \mathbf{p}_{i,j}\right) = h\frac{\mathbf{R}_{i+1} + \mathbf{R}_i}{2}e3 + \mathbf{R}_{i+1}r_{i+1,j} - \mathbf{R}_i r_{i,j}$$

$$\left(\mathbf{p}_{i,j} - \mathbf{p}_{i-1,j}\right) = h\frac{\mathbf{R}_i + \mathbf{R}_{i-1}}{2}e3 + \mathbf{R}_i r_{i,j} - \mathbf{R}_{i-1} r_{i-1,j}$$

Assuming uniform tension along the tendons, the force exerted by the j^{th} tendon on the i^{th} link is given by:

$$\mathbf{f}_{ji} = \left(\mathbf{t}_{i+1} - \mathbf{t}_i\right)\boldsymbol{\tau}_j$$

Where $\boldsymbol{\tau}_j$ is the tendon tension (i.e., the j^{th} actuator input). The resulting moment is obtained by crossing $\mathbf{f}_{i,j}$ with the moment arm, defined as the vector from the link frame to the tendon channel.

$$\ell_{i,j} = \left(\mathbf{R}_i r_{i,j}\right)^{\wedge}\left(\mathbf{t}_{i+1} - \mathbf{t}_i\right)\boldsymbol{\tau}_j$$

Tendon Actuation via Displacement Commands

Many actuation methods directly dictate forces or torques, which can be used to calculate actuation inputs like tendon tension. However, some actuators, such as stepper motors, dictate displacement directly. To accommodate displacement-based commands, the dynamic model can be modified to incorporate tendon length control.

One approach involves rewriting the model's constraints to include tendon displacements, but this significantly complicates the formulation and is beyond the scope of this work. Instead, a simpler method uses a high-gain PD controller acting on tendon tension to drive the tendon lengths to desired values:

$$\boldsymbol{\tau}_j = k_P \left(L_j - L_{j,ref}\right) + k_D \left(\dot{L}_j - \dot{L}_{j,ref}\right)$$

where L_j is the current tendon length of the j^{th} tendon, $L_{j,ref}$ is the reference tendon length of the j^{th} tendon, and k_P and k_D are proportional and derivative gains, respectively. With sufficiently large gains, the model rapidly converges to the desired tendon lengths. The tendon length is calculated by summing the distances between adjacent tendon channels:

$$L_j = \sum_{i=1}^{N} ||h_i \frac{\mathbf{R}_i + \mathbf{R}_{i-1}}{2} e_3 + \mathbf{R}_i \mathbf{r}_{i,j} - \mathbf{R}_{i-1} \mathbf{r}_{i-1,j}||$$

Alternatively, tendon displacement can be controlled by multiplying the desired change in length by a stiffness gain:

$$\boldsymbol{\tau}_j = k_R L_{j,ref}$$

Although less robust, this method treats tendon lengths as softer constraints which can be beneficial for a model that is coarsely discretized and uses corrective observer forces.

Full Model with Actuation

To incorporate actuation into the model in (4.15), we apply the forces and moments (resulting from actuation) to $\mathbf{F}$ to obtain the following:

$$\begin{bmatrix} \mathbf{M} & \mathbf{A}^\top \\ \mathbf{A} & \mathbf{0} \end{bmatrix} \begin{bmatrix} \dot{\mathbf{v}} \\ \boldsymbol{\lambda} \end{bmatrix} = \begin{bmatrix} \mathbf{F} \\ \boldsymbol{\gamma} \end{bmatrix} + \begin{bmatrix} \mathbf{B} \\ \mathbf{0} \end{bmatrix} \boldsymbol{\tau} \tag{4.16}$$

Here, $\boldsymbol{\tau} \in \mathbb{R}^{n_\tau \times 1}$ is composed of the $\boldsymbol{\tau}_j$ actuator inputs, $\mathbf{B} \in \mathbb{R}^{6n \times n_\tau}$ matrix that dictates how the actuator input are distributed across the model. Each column of $\mathbf{B}$ corresponds to the force and moment distribution across the robot associated with the actuator input from its respective element in $\boldsymbol{\tau}$. Therefore, the $6i-5$ to $6i^{th}$ row and the j^{th} column of $\mathbf{B}$ is composed of the following

$$\mathbf{B}_{i,j} = \begin{bmatrix} (\mathbf{t}_{i+1} - \mathbf{t}_i) \\ (\mathbf{R}_i r_i)^\wedge (\mathbf{t}_{i+1} - \mathbf{t}_i) \end{bmatrix}$$

Integrating the Full State Vector:

We solve for accelerations $\dot{\mathbf{v}}$ in (4.16) and use them to integrate a state vector $\mathbf{q}$ consisting of positions, orientations (represented by quaternions), and velocities of each discrete link. $\mathbf{q}$ then contains $13n$ state variables where the elements $13i-12$ to $13i$ describing the i^{th} link are $\mathbf{q}_i = [\mathbf{p}_i, \boldsymbol{\psi}_i, \dot{\mathbf{p}}_i, \mathbf{w}_i]^\top$. Here, $\boldsymbol{\psi}_i$ the quaternion describing the i^{th} link's orientation. To calculate the quaternion derivative, we use the following relationship with angular velocity:

$$\dot{\boldsymbol{\psi}}_i = \frac{1}{2} \boldsymbol{\Omega}_i \boldsymbol{\psi}_i \tag{4.17}$$

where $\boldsymbol{\Omega}_i$ denotes a skew-symmetric matrix form of the angular velocity vector $\mathbf{w}_i$ as defined below (omitting the i subscripts for clarity):

$$\boldsymbol{\Omega} = \begin{bmatrix} 0 & -w_1 & -w_2 & -w_3 \\ w_1 & 0 & -w_3 & w_2 \\ w_2 & w_3 & 0 & -w_1 \\ w_3 & -w_2 & w_1 & 0 \end{bmatrix}$$

and $\boldsymbol{\psi}_i$ can be obtained from the state vector $\mathbf{q}$. The quaternion $\boldsymbol{\psi}_i$ is used to calculate the rotation matrix $\mathbf{R}_i$, following the general formulation in [62] (dropping the i subscript for clarity):

$$\mathbf{R} = \frac{1}{||\boldsymbol{\psi}||^2} \begin{bmatrix} \psi_1^2 + \psi_2^2 - \psi_3^2 - \psi_4^2 & 2(\psi_2\psi_3 - \psi_1\psi_4) & 2(\psi_2\psi_4 + \psi_1\psi_3) \\ 2(\psi_2\psi_3 + \psi_1\psi_4) & \psi_1^2 + \psi_3^2 - \psi_2^2 - \psi_4^2 & 2(\psi_3\psi_4 - \psi_1\psi_2) \\ 2(\psi_2\psi_4 - \psi_1\psi_3) & 2(\psi_3\psi_4 + \psi_1\psi_2) & \psi_1^2 + \psi_4^2 - \psi_2^2 - \psi_3^2 \end{bmatrix} \tag{4.18}$$

When used in this way, the quaternion does not need to be normalized explicitly after each integration step [62].

4.2 Observer

In any practical sensing scenario, we can only measure a handful of points along a robot, but we would like to estimate the full robot state to the resolution of our chosen discretization. We propose that an intuitive model-based observer strategy can be used to do this. We augment forward dynamics simulation with virtual forces and moments collected into a vector $\mathbf{g}$, designed to drive the estimated robot position and orientation to the measured position and orientation. Thus, the observer dynamic equations take the form:

$$\begin{bmatrix} \mathbf{M} & \mathbf{A}(\tilde{\mathbf{q}})^\top \\ \mathbf{A}(\tilde{\mathbf{q}}) & \mathbf{0} \end{bmatrix} \begin{bmatrix} \dot{\tilde{\mathbf{v}}} \\ \tilde{\boldsymbol{\lambda}} \end{bmatrix} = \begin{bmatrix} \mathbf{F}(\tilde{\mathbf{q}}) + \mathbf{B}(\tilde{\mathbf{q}})\boldsymbol{\tau} + \mathbf{g} \\ \boldsymbol{\gamma}(\tilde{\mathbf{q}}) \end{bmatrix} \tag{4.19}$$

where $\tilde{}$ denotes an estimated variable and $\mathbf{g} \in \mathbb{R}^{6n}$ is a sparse vector containing forces and moments. $\mathbf{g}_i \in \mathbb{R}^6$ populates elements $6i-5$ to $6i$ of $\mathbf{g}$. These elements correspond to some measured point along the i^{th} link. All other elements in $\mathbf{g}$ that do not correspond to a measured state are zero. The observer forces are contained in the first half of $\mathbf{g}_i$ and are represented by $\mathbf{F}_{\mathbf{g}_i} \in \mathbb{R}^3$. These elements can be calculated by a proportional derivative control law on the model-sensor mismatch:

$$\mathbf{F}_{\mathbf{g}_i} = L_d\left(\dot{\mathbf{p}}_i - \dot{\tilde{\mathbf{p}}}_i\right) + L_p\left(\mathbf{p}_i - \tilde{\mathbf{p}}_i\right)$$

Where L_p and L_d are positive proportional and derivative gains, respectively.

The second half of $\mathbf{g}_i$ contains the moments represented by $\mathbf{m}_{\mathbf{g}_i} \in \mathbb{R}^3$. This term includes moments resulting from both $\mathbf{F}_{\mathbf{g}_i}$ and any additional moments implemented to drive the model orientation to the measured orientation.

$$\mathbf{m}_{\mathbf{g}_i} = \mathbf{r}_{im} \times \mathbf{F}_{\mathbf{g}i} + L_{dm}\left(\mathbf{w}_i - \tilde{\mathbf{w}}_i\right) + L_{pm} \log\left(\tilde{\mathbf{R}}_i^\top \mathbf{R}_i\right)^\vee$$

Where L_{pm} and L_{dm} are positive proportional and derivative gains acting on orientation and angular velocity, respectively, and $\mathbf{r}_{im}$ is the moment arm vector from the centroid of the i^{th} link to the measured point. for example, if the measured point is on the centerline a distance h_{im} further along than the centroid of link i, then $\mathbf{r}_{im} = \tilde{\mathbf{z}}_i h_{im}$

This simple observer law is difficult to destabilize because it is passively implementing a virtual spring and damper to pull the estimated state to the measured state [5]. The entire observer algorithm is indicated by the block diagram in Figure 4.4.

Depending on the sensing equipment, the velocity terms ($\dot{\mathbf{p}}$ and $\mathbf{w}_i$) may have to be computed by taking a numerical time derivative of the measured position and orientation data. For $\mathbf{p}$ this is straightforward. For angular velocity, we can obtain the angular velocity from quaternion derivatives as

$$\tilde{\mathbf{w}} = \frac{2}{||\boldsymbol{\psi}||^2} \begin{bmatrix} -\psi_2 & \psi_1 & \psi_4 & -\psi_3 \\ -\psi_3 & -\psi_4 & \psi_1 & \psi_2 \\ -\psi_4 & \psi_3 & -\psi_2 & \psi_1 \end{bmatrix} \dot{\boldsymbol{\psi}} \tag{4.20}$$

again, omitting the i subscript for clarity and using the formulation that accounts for a general (not necessarily unit) quaternion [62].

4.3 Robot Design

To validate the model and observer structure, we constructed an experimental tendon-driven robot with a mounting frame for sensors and actuators (illustrated in Figure 4.5). The backbone consists of a flexible PVC tube with helical bellows and a central carbon fiber rod.

The helical design provides high torsional rigidity while maintaining bending compliance. The carbon fiber rod prevents unwanted axial deformation. To secure the rod, six 3D-printed inserts are threaded into the tube's helical structure, each with a central hole for a snug fit. Gluing the rod to the inserts ensures a fixed backbone length. Six 3D-printed outserts, designed for tendon routing through low-friction polytetrafluoroethylene (PTFE) tubes, are threaded onto the exterior of the helical vacuum tube bellows. Set screws secure the outserts in place.

While the design can accommodate six tendons, only three are necessary for model validation. Kevlar tendons are routed through the channels and secured at designated points along the robot. For our experiments, three tendons are attached to the robot's tip in an evenly spaced circular pattern, 120° apart. The tendons' proximal ends are wound around spools with a 0.065 m radius, actuated by Nema 23 stepper motors, allowing precise control over tendon lengths. The tendons follow a straight-line path from the spools to the tip, avoiding helical wrapping around the backbone. The tendon channels are positioned 0.045 m from the backbone's center, with tendons 1, 2, and 3 attached as follows (expressed in meters):

$$\begin{aligned} \mathbf{r}_{i,1} &= [0, 0.045, 0]^\top, \\ \mathbf{r}_{i,2} &= \left[0.045 \sin\left(\frac{2\pi}{3}\right), 0.045 \cos\left(\frac{2\pi}{3}\right), 0\right]^\top \\ \mathbf{r}_{i,3} &= \left[0.045 \sin\left(\frac{4\pi}{3}\right), 0.045 \cos\left(\frac{4\pi}{3}\right), 0\right]^\top \end{aligned} \tag{4.21}$$

Motor commands from Simulink are transmitted to an Arduino Uno via serial communication, then relayed via I2C to Pololu Tic stepper drivers, which actuate the motors using $1/8^{th}$ stepping.

An optical tracking system with four OptiTrack Flex 13 cameras (sub-millimeter accuracy, 120 Hz) provides data for the observer model and system evaluation. Tracking markers are placed at six equally spaced points along the backbone of the robot (starting at 1/6 of the robot's length and progressing to the tip of the robot. An extra marker set on the base of the robot is used for calibration. All other markers are used to evaluate the position and orientation of the robot. The robot is mounted upside-down inside a darkened enclosure, which improves marker tracking. Reflective surfaces are coated with matte black

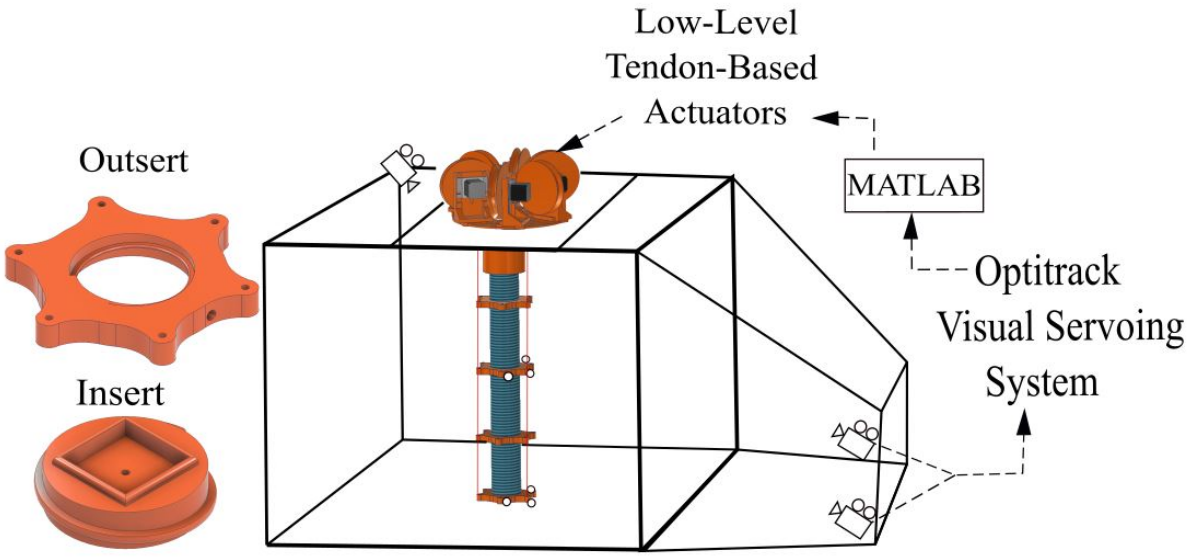


Figure 4.5: This figure shows an illustration of the experimental continuum robot. The upper left side of the figure shows the threaded outserts. These are threaded on the outside of the helical bellows tube allowing for easily added tendon routing channels. The lower left side of the figure shows the inserts. These are threaded inside the tube and allow for a carbon fiber rod to be fixed in the center of the robot. The right side of the figure shows an illustration of the apparatus in its assembled state.

paint or covered with non-infrared reflective fabric. This setup (shown in Figure 4.6) offers a robust, modular, and cost-effective continuum robot testing system.

4.4 Experiments

Multiple experiments were conducted to evaluate computational speed, open-loop model accuracy, and observer accuracy. The robot's physical properties were first calibrated. Bending rigidity $K_b = 7.48 \times 10^{-1}$ N m^2 was estimated from a cantilevered beam test using the robot structure. Torsional rigidity was set to a relatively large value ($K_t = 10K_b$) in order to model the practically torsionless behavior of the structure due to the torsionally stiff bellows backbone. The stiffness matrix is then $\mathbf{K} = \text{diag}(K_b, K_b, K_t)$. The damping matrix was calibrated as $\mathbf{D}_i = 0.1\mathbf{K}_i$ to match experimentally observed decay behavior. The inertial property ρA (mass per unit length) was calculated to be 4.16×10^{-1} kg/m from the

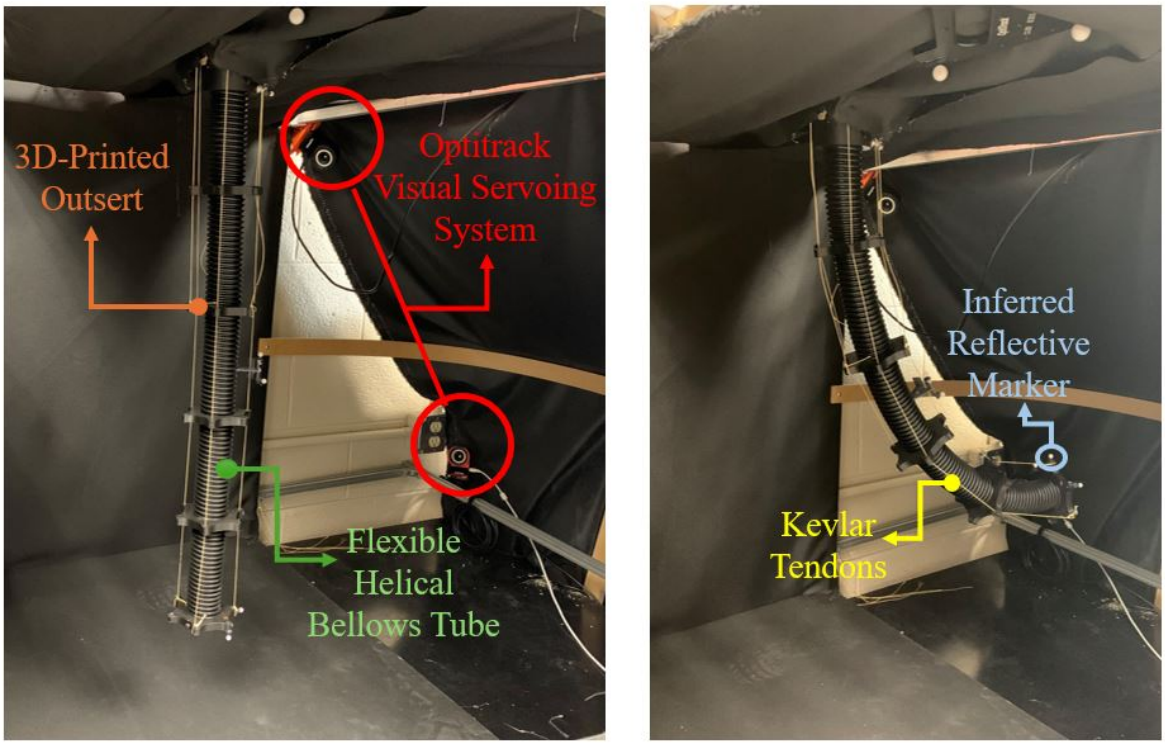


Figure 4.6: This figure shows the experimental continuum robot set up. The left side of the figure shows the robot in its undeformed shape. The right side of the figure shows the robot deforming under actuation. The optical tracking system, helical bellows backbone, 3D-printed outserts, Kevlar tendons and reflective markers are all indicated in the figure.

total manipulator mass and total length of 0.731 meters. The inertial property ρJ (cross-sectional rotational inertia per unit length) was calculated by approximating the structure as a tube with inner and outer radii of .020 and .025 meters respectively, with the previously calculated mass per unit length, resulting in $\rho\mathbf{J} = \text{diag}(1.07, 1.07, 2.13) \times 10^{-4}$ kg m.

We evaluated model solution speed and scaling using two different numerical integration routines, ODE15s (adaptive timestep) and Backward Euler ($0.015s$ timestep) and various levels of spatial discretization. These implicit integrators are well-suited for "stiff" systems of differential equations. In these test cases, a $10N$ tendon force step input was applied to excite the dynamics. Two different methods of solving the linear system were also compared, MATLAB's mldivide() function and a block tri-diagonal solver based on the Thomas algorithm [36].

We then evaluated model accuracy while executing a dynamic motion, both with the observer force and moment terms driven by sensed measurements at the tip (observer

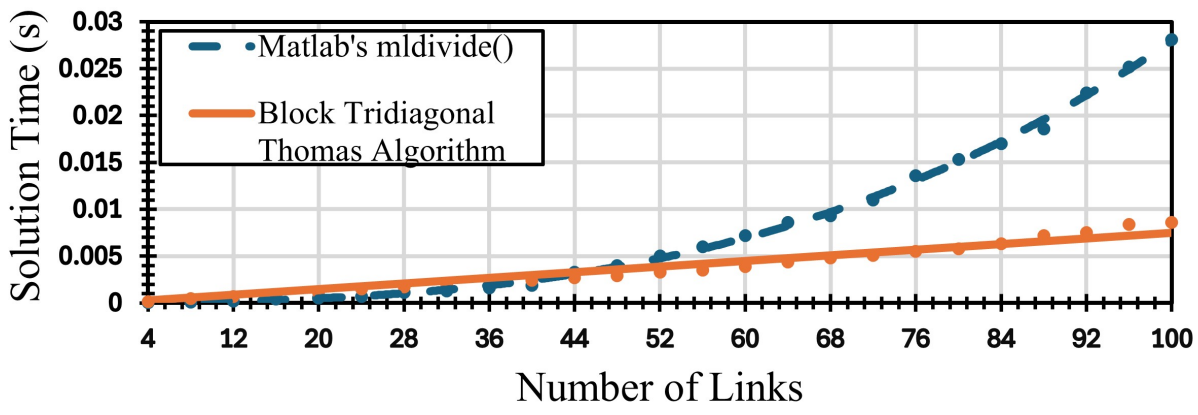


Figure 4.7: This figure shows the solution time of both MATLAB's mldivide() (indicated by the dashed line), and the Thomas block tri-diagonal algorithm (indicated by the solid line) as the number of links used to discretize the Kirchhoff rod increase. The mldivide() algorithm scales cubically but is more efficient for lower numbers of links. Alternatively, the Thomas algorithm scales linearly and is therefore more efficient for higher numbers of links.

accuracy), and without them (open-loop model accuracy). Tip and center measurements are recorded via an optical tracking system to evaluate both cases. Simulink's real-time toolbox is used with fixed-time step integration (Backward Euler, $0.025s$). The tendon length trajectory is:

$$\mathbf{L}_{ref} = \begin{bmatrix} 0.731 \\ 0.731 \\ 0.731 \end{bmatrix} - 0.0676 \begin{bmatrix} \sin(\omega t) \\ \cos\left(\omega t + \frac{5\pi}{6}\right) \\ \cos\left(\omega t + \frac{\pi}{6}\right) \end{bmatrix}$$

where $\omega = 1$ and t is time.

All simulations were run in Simulink on an Intel Xeon E3-1505M v5 CPU. The robot started fully extended with the tip directly below the base. For open-loop experiments, tendon length is controlled via PD control ($k_P = 400$ N/m, $k_D = 100$ N/m^2). For experiments using the observer, tendon forces were dictated (as described above) proportional to their respective changes in length with a proportional constant of $k_R = 347.2$ N/m. The observer then compensates for unmodeled dynamic fluctuations in actuator force and other inaccuracies and disturbances. Observer force gains are $L_p = 3000\ \frac{N}{m}$, $L_d = 30\ \frac{Ns}{m}$, $L_{pm} = 0.5\ \frac{N}{\text{rad}}$, and $L_{dm} = 0.4\ \frac{Ns}{\text{rad}}$. These gains were obtained manually by iteratively observing the observer error performance and tuning the gains heuristically.

4.5 Results

4.5.1 Computational Speed

The computational speed tests revealed that with a step size of $0.015s$, using Backward Euler integration, real-time simulation is achievable for $n < 8$ links, with $n = 6$ links running 1.7 times faster than real time. ODE15s performs significantly better, simulating 860% faster than Backward Euler at $n = 6$ links and allowing 44-link simulations to run at 1.09 times faster than real-time speed. The scaling of computation time with number of links is shown in Figure 4.7 and was found to be $O(n^3)$ when using MATLAB's mldivide() to solve the linear system (4.16) for the accelerations and Lagrange multipliers. Using a partitioned system solve and block-tridiagonal solver based on Thomas's algorithm [58], the scaling is $O(n)$. However, mldivide() is faster in absolute terms for small numbers of links (less than 42). We use mldivide() for the remainder of the experiments.

4.5.2 Open Loop Model Estimation

With a 6-link discretization, the model is tested on a dynamic trajectory without any corrective forces applied. The positions of the tip and the center of the robot are presented in Figure 4.8. It can be observed that, despite the lack of corrective terms, the estimated positions of the tip and center (denoted as X_{Est}, Y_{Est}, and Z_{Est}) remain in close agreement with the corresponding measured robot positions (denoted by X_{Meas}, Y_{Meas}, and Z_{Meas}). This alignment demonstrates the accuracy of the model in predicting the robot's movement.

However, while the estimated values track the measured values closely, noticeable deviations are evident. As illustrated in Figure 4.9, the mean position error across all six points along the robot arm for the duration of the experiment is 0.0512 m, representing 7.62% of the total robot length. The mean position error at each measured point is 0.0209 m at point 1, 0.0401 m at point 2, 0.0486 m at the center of the robot, 0.0592 m at point 3, 0.0698 m at point 4, and 0.0761 m at the tip of the robot. This corresponds to 3.11%, 5.96% , 7.22%, 8.88%, 10.38%, and 11.31% of the robot's length respectively. The mean orientation errors at the tip are 11.60°, 3.15°, and 3.35° about the x, y, and z axes (indicated as θ_X, θ_Y,

and θ_Z), respectively. These errors, though not excessively large, show a degree of inaccuracy that can be attributed to various discrepancies between the model and the physical robot, such as artifacts of the 6-link discretization, unmodeled frictional effects between the tendons and the routing channels, and parametric inaccuracy.

4.5.3 Force Based Observer Estimation

We performed the same experiment with the model-based observer to illustrate its performance in correcting for modeling error and accurately predicting the full state of the robot, driven by a small number of sensor measurements. Recall that the observer introduces a point force at any point where position is measured and a point moment at any point where the orientation is measured. In the case of this experiment we sense position and orientation at a single point (the tip of the manipulator).

As shown in the plots of the observer and robot's tip and center positions in Figure 4.10, the addition of corrective forces significantly enhances dynamic tracking accuracy. The observer model continues to accurately follow the robot's true motion despite transient sensor noise such as temporarily corrupt sensor reading.

Figure 4.11 further highlights the improvement, with significantly reduced errors compared to the open-loop case. The mean position error across all measured points along the robot for the duration of the experiment drops to 0.0128 m, representing just 1.91% of the robot length. Additionally, the mean position error at each point decreases to 0.0077 m at point 1, 0.0121 m at point 2, 0.0144 m at the center, 0.0159 m at point 3, 0.0188 m at point 4, and 0.0083 m at the tip of the robot. This corresponds to 1.14%, 1.80%, 2.14%, 2.37%, 2.80%, and 1.23% of the robot's length, respectively. Orientation errors are also notably reduced, with mean errors at the tip recorded as 2.61°, 2.35°, and 1.43° about the x, y, and z axes, respectively.

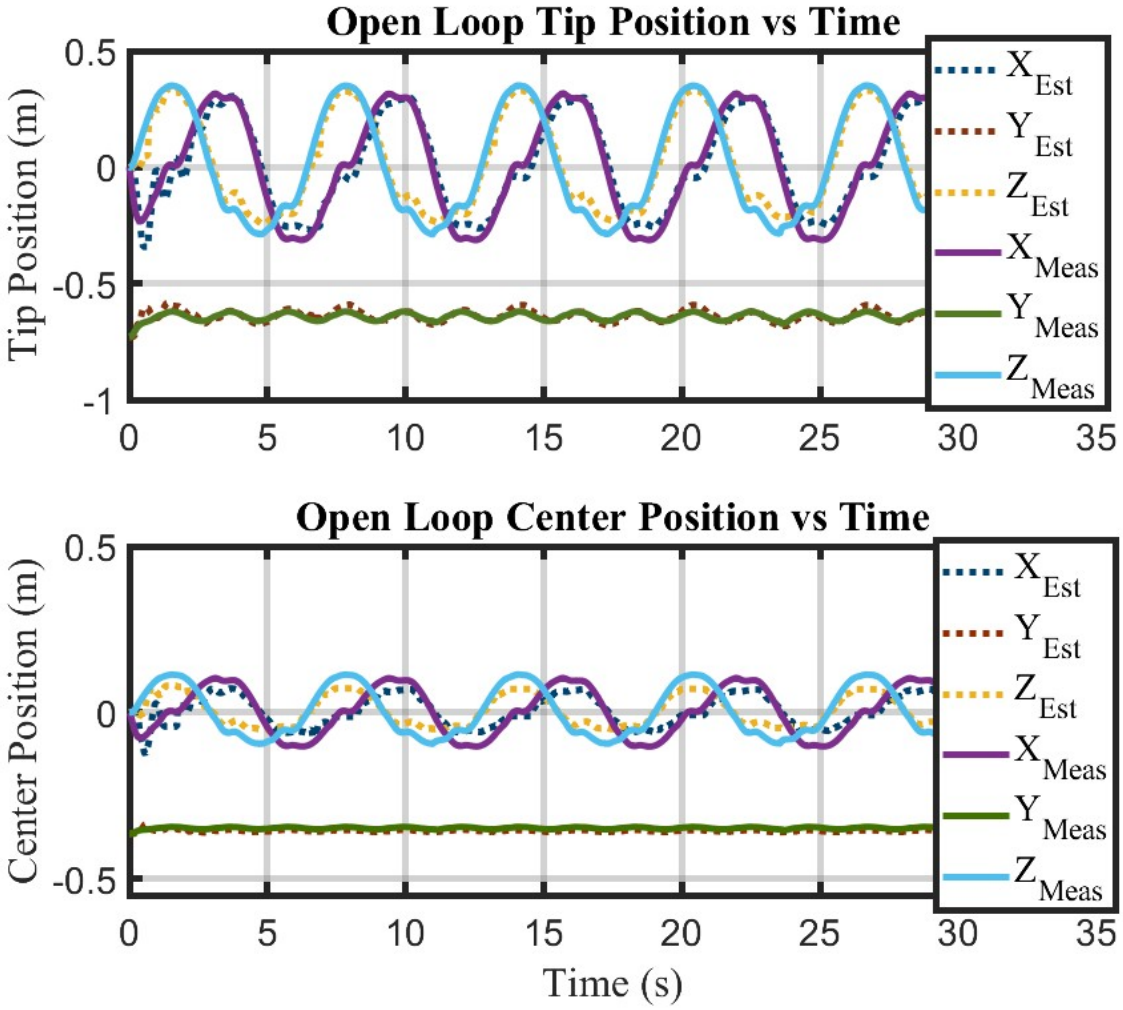


Figure 4.8: This figure shows plots of the tip and center positions of the robot and the model as it follows a given trajectory. No Restorative forces or moments are acting on the model. Here X_{Est}, Y_{Est}, and Z_{Est} represent the estimated x, y, and z values. Similarly, X_{Meas}, Y_{Meas}, and Z_{Meas} represent the measured x, y, and z values.

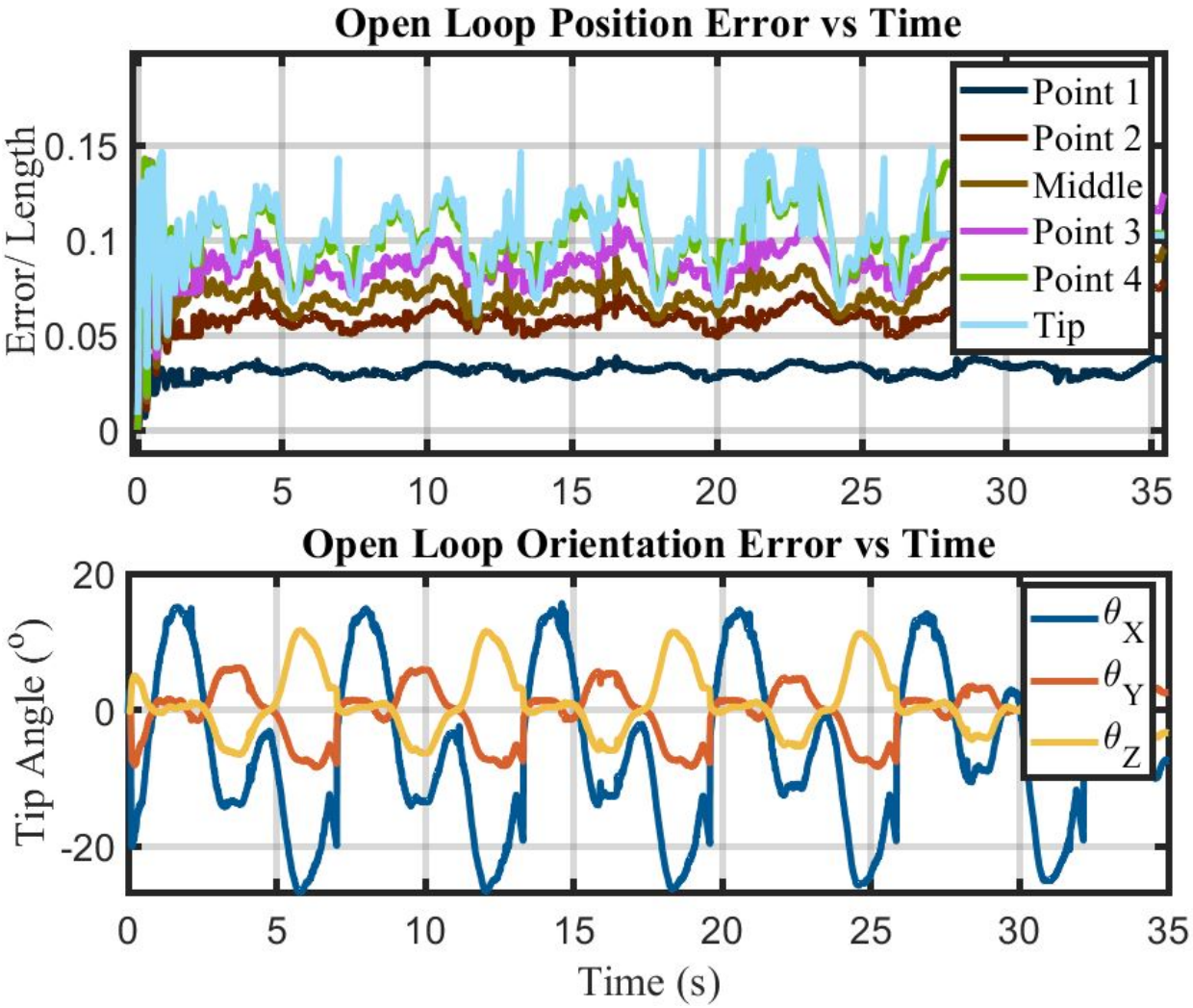


Figure 4.9: This figure contains the open loop position error magnitudes of 6 equally spaced measured points along the robot arm as well as the tip orientation error. The nondimensionalized mean position errors across all measured points along the robot for the duration of the experiment are 0.0762 of the robot's length. Mean orientation errors about the x, y, and z axis (represented as θ_X, θ_Y, and θ_Z) are 11.60°, 3.15°, and 3.35° respectively.

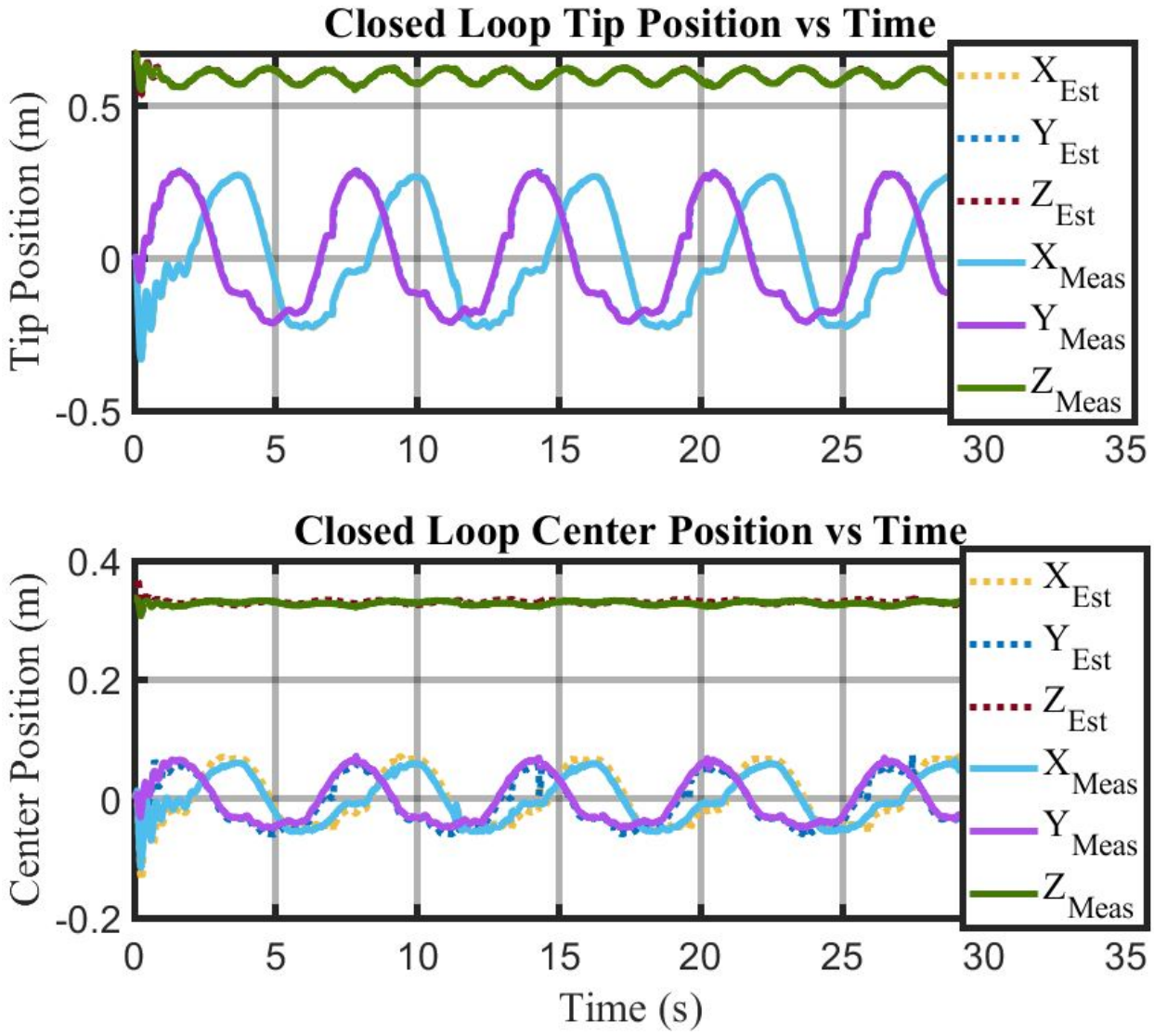


Figure 4.10: This figure shows plots of the tip and center positions of the robot and the model as it follows a given trajectory. Restorative forces and moments act on the tip of the robot to compensate for unmodeled forces. Here X_{Est}, Y_{Est}, and Z_{Est} represent the estimated x, y, and z values. Similarly, X_{Meas}, Y_{Meas}, and Z_{Meas} represents the measured x, y, and z values.

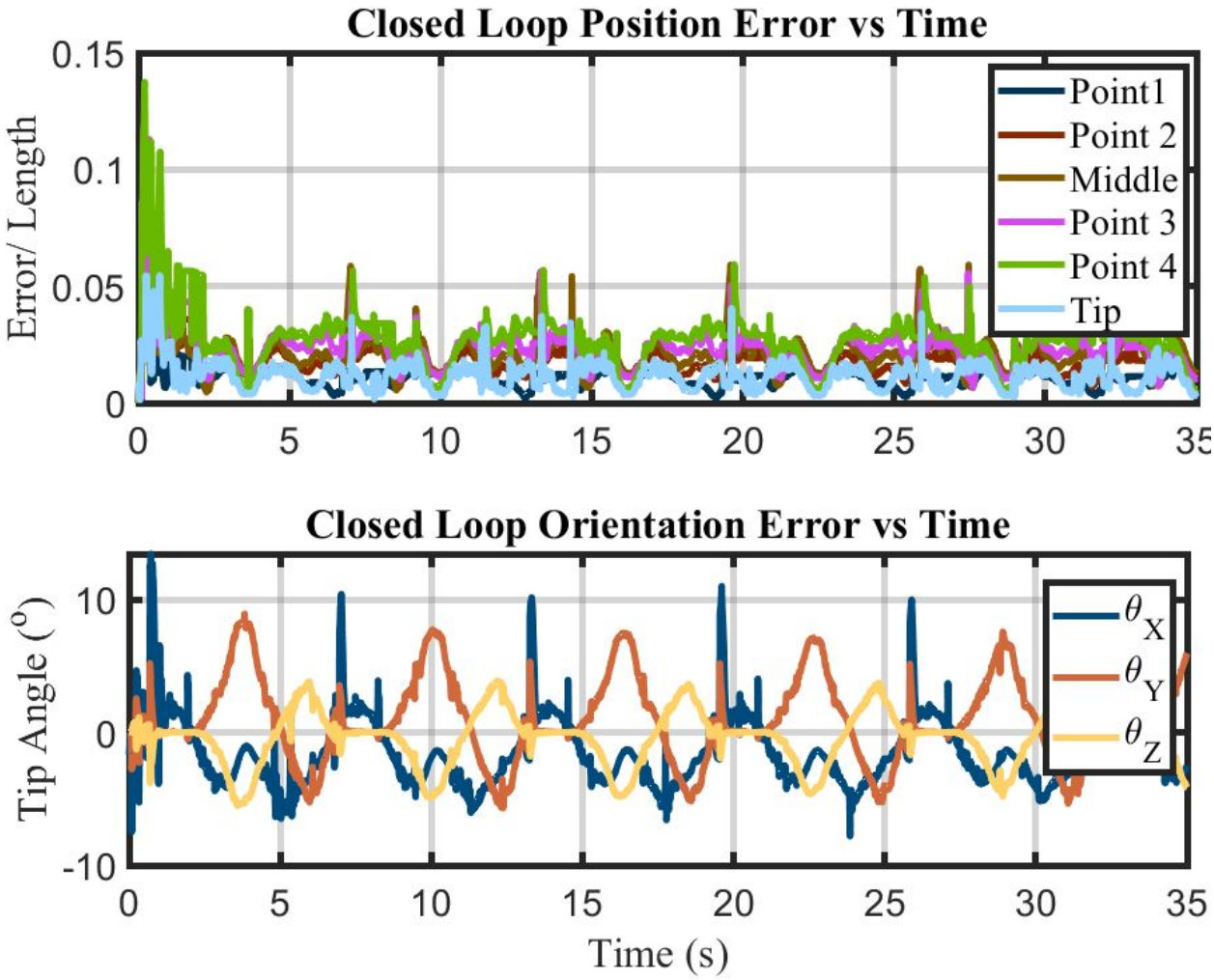


Figure 4.11: This figure contains closed loop observer position error magnitudes of 6 equally spaced measured points along the robot arm as well as tip orientation error. The nondimensionalized mean position errors across all measured points along the robot for the duration of the experiment are 0.0191 of the robot's length. Mean tip orientation errors about the x, y, and z axis (represented as θ_X, θ_Y, and θ_Z) are 2.61°, 2.35°, and 1.43° respectively.

Chapter 5

Background of Surgical Palpation via Continuum Robots

In this chapter, we discuss the literature surrounding the force sensing in a surgical setting using continuum robots, specifically concentric tube robots, in order to motivate Chapters 6 and 7.

Humans are naturally equipped with highly sensitive force sensors—our fingers. These highly sensitive biological sensors allow us to perceive not only texture but also the stiffness of objects through touch. This fundamental capability underlies the clinical technique known as palpation. Palpation, the clinical practice of feeling tissue to assess its mechanical properties, remains a fundamental component of surgical diagnosis, especially in identifying hard nodules or abnormal masses such as tumors. During physical examinations, physicians often palpate areas such as the neck or abdomen to detect abnormal stiffness, which may indicate inflammation, swollen lymph nodes, or other pathological conditions. In open surgery, palpation offers surgeons valuable tactile feedback, enabling them to locate tumors, lesions that lie beneath the tissue surface, or other bodily structures that may not be apparent from sight alone.

In recent years, robot-assisted minimally invasive surgery (RMIS) has transformed surgical practice by improving precision, dexterity, and control during procedures. Clinical studies have demonstrated that RMIS is associated with reduced postoperative pain, faster recovery times, and a lower risk of infection compared to conventional open surgery [14].

Among the various robotic platforms used for minimally invasive procedures, continuum robots, specifically concentric tube robots (CTRs) are particularly attractive due to their compact size, flexible motion, and ability to navigate within confined spaces [33]. Despite these advantages, a major drawback of CTR systems is the lack of effective force feedback, limiting the surgeon's ability to assess tissue stiffness.

Various sensor technologies, including resistive, capacitive, and piezoelectric sensors, have been developed to embed directly onto minimally invasive surgical instruments for accurate measurement of end-effector forces [72, 54]. However, these sensors typically have dimensions that are too large to be effectively integrated into CTR systems, where compactness and flexibility are crucial requirements. Consequently, alternative sensing strategies that offer high precision while maintaining the small form factor of CTRs remain a critical area for further research.

To overcome these size constraints, researchers have explored indirect methods to estimate forces based on robot deformation rather than directly measuring the forces at the end-effector, as reviewed in [32]. The inherent flexibility of CTRs allows their tube deflection to serve as an indicator of external forces acting on the robot [66, 49]. However, accurately capturing the robot's shape and deflection remains a major challenge for reliable force estimation.

Recently, various methods have been explored to sense the deformed shape of concentric tube robots (CTRs), particularly through integrated sensing technologies [74]. Among these, both fiber Bragg grating (FBG)–based shape sensing and electromagnetic (EM) tracking have been widely investigated. FBG sensors have received significant attention due to their high strain sensitivity and compatibility with medical environments [95]. By measuring wavelength shifts induced by strain, FBG sensors enable accurate curvature reconstruction and have been used for both 2-D [1, 56] and 3-D shape sensing in CTRs [97, 90]. EM tracking systems, consisting of an external field generator and miniature receiver coils, enable real-time measurement of position and orientation and have been adopted in clinical navigation systems and medical instruments, as well as in continuum robot shape and pose tracking [22, 47, 55, 85, 76].

In this work, we use EM tracking due to its relatively low cost, off-the-shelf availability, and established precedent of adoption in clinical products. These practical considerations make EM tracking well suited for rapid prototyping and deployment in compact robotic palpation tools. Importantly, the proposed model-based force estimation framework is not specific to EM sensing and could be applied with alternative tracking technologies, including fiber-based shape sensing.

In the next two chapters, we introduce novel designs of palpation tools that are integrated into a Virtuoso Endoscopy System (Virtuoso Surgical, Nashville, TN). We then use practical deflection-based force estimation algorithms to leverage the inherent compliance of the tools and the concentric tubes of the robot and apply a calibrated mechanical model that incorporates the kinematic configuration of the CTR. Furthermore, we use EM tracking integrated into a palpation tool to provide relatively low cost and robust tracking at a small scale. The tip deflection is estimated as the offset between the tip position measured by the magnetic tracker and the unloaded tip position predicted by the model, and this deflection is then used to compute contact forces in real time without embedded force sensors. This technique is compatible with the constraints of minimally invasive surgical environments and can be easily transferred to other CTR systems for force estimation.

One difficulty associated with this approach is that the compliance matrices relating tip deformations to external forces are ill-conditioned due to the extremely high axial stiffness of CTRs relative to their bending stiffness, and the tool geometry. This has been recognized as a challenge since the earliest studies on deflection-based force estimation [66, 49]. The majority of existing work addressing this issue have assumed the axial force to be negligible. Some work has attempted to develop force sensing capabilities in the stiff direction of a tube by using a FBG in combination with a laser patterned tube to increase the sensitivity in the stiff direction [31]; however, this design was not intended for use in concentric tube robots. To address this challenge, we investigate multiple solutions. In Chapter 6 we investigate using a force sensor at the base of the robot to sense forces that are transmitted along the stiff axis of the tube. In chapter 6, we also show an approach in which we systematically remove the ill-conditioning from our prediction algorithm. In Chapter 7, we address some of the limitations of the previous methods we presented by designing new palpation tools that

allow for tip deformation in the stiff direction while still maintaining the desired bending compliance. This is done by fixing an axial spring and plunger to the end of the inner tube of the robot. The inner tube can then be laser patterned in order to achieve the desired bending stiffness.

The main contributions of the work presented over the next two chapters, compared with existing approaches in the literature, are as follows:

1. Given an inner tube inner diameter of only 0.82 mm, we design a compact palpation tool integrating a 5-DOF magnetic tracking sensor within a Nitinol tube, enabling real-time tip pose measurement without substantially increasing the distal diameter or requiring embedded force sensors.

2. We use a quasi-static Cosserat-rod–based model that relates tip deflection and orientation change to external forces through a calibrated compliance matrix, enabling real-time, model-based force estimation.

3. We introduce multiple solutions to the ill-conditioning of the compliance matrix caused by high axial stiffness. We do this by multiple methods. We ether, use a numerically robust singular-value-decomposition (SVD)–based least-squares formulation to ignore the stiff direction, or by making modifications to the existing hardware that allows for axial sensing.

4. We experimentally validate the proposed force estimation method using the Virtuoso Endoscopy System to push against an external force sensor with our palpation tools.

5. We show that the proposed methods can be effectively used for tissue stiffness mapping by validating its performance in simulation using an XPBD-based deformable-tissue model and by palpating phantom tissue with hard sub surface nodules.

Chapter 6

Force Sensing and Tissue Palpation

Here, we designed and implemented a compact palpation tool integrated with the Virtuoso Endoscopy System. This method combines real-time magnetic tracking of the tool tip with a physics-based model of the concentric tube robot to infer contact forces from observed tip deflections. As shown in Figure 6.1, during palpation the robot tip is commanded to a specified position, and the tool tip deflects with a direction and magnitude that depend on the stiffness of the contacted tissue. Given the known stiffness of the robot in a particular configuration, both the applied force and the effective tissue stiffness can be inferred from the measured deflection.

This approach builds on concepts introduced in previous chapters, which developed a force-based state estimator for continuum robots for environments where dynamics significantly influence system behavior. In that work, a force-based observer was used to drive a model-predicted tip position toward a measured position, yielding an estimate of a virtual interaction force. While both approaches rely on similar underlying principles, their objectives differ: the previous framework prioritized accurate state estimation for control of robot dynamics, whereas the following work prioritizes inferring interaction forces for use in palpation and stiffness estimation.

In this chapter, we investigate the proposed position-based force sensing approach. We begin by describing the surgical robotic platform for which the algorithms are developed, along with the hardware used to enable force sensing. We then present the development of

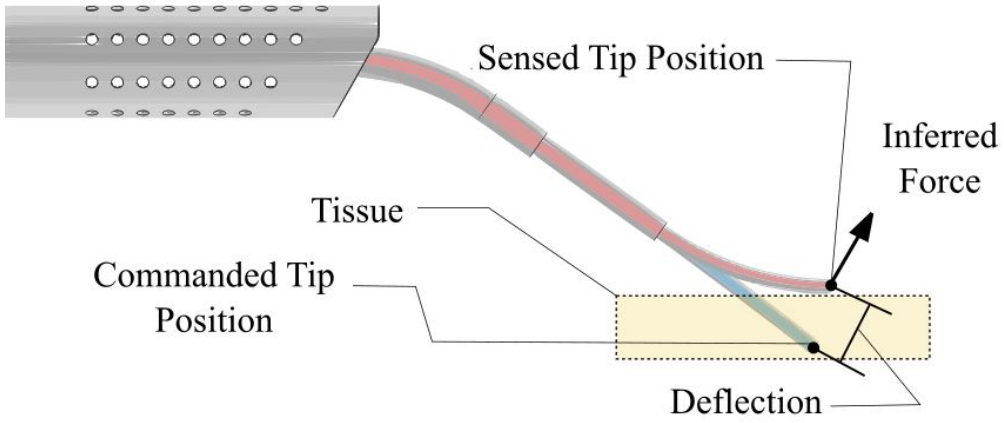


Figure 6.1: Deflection-based force sensing uses the known stiffness of the robot and the observed deflection from the commanded position to infer the tissue force.

the mathematical model and the associated force estimation algorithm. Finally, we describe the experimental procedures used to validate the proposed approach.

6.1 Hardware

This section details the robotic platform, the design of a custom palpation tool, and the integrated sensing apparatus used for force and position measurement.

Our palpation system is designed for integration with the Virtuoso surgical concentric tube robot platform. The Virtuoso system provides a dual-arm robotic platform for minimally invasive surgery. Each arm is a concentric tube manipulator comprising a pre-curved Nitinol outer tube (ID 1.19 mm, OD 1.56 mm) and a straight, Nitinol inner tube (ID 0.84 mm, OD 1.02 mm), both deployable through a sheath. When fully retracted, the tubes align and the outer pre-curved tube straightens. Each arm is housed in a removable cartridge, and a motor pack independently actuates translation and rotation of the inner and outer tubes. Using appropriate forward and inverse kinematic models, the Virtuoso robot achieves accurate control of the tool tip position and orientation. In this study, one arm is dedicated to deploying the custom palpation tool through the central lumen of its inner tube.

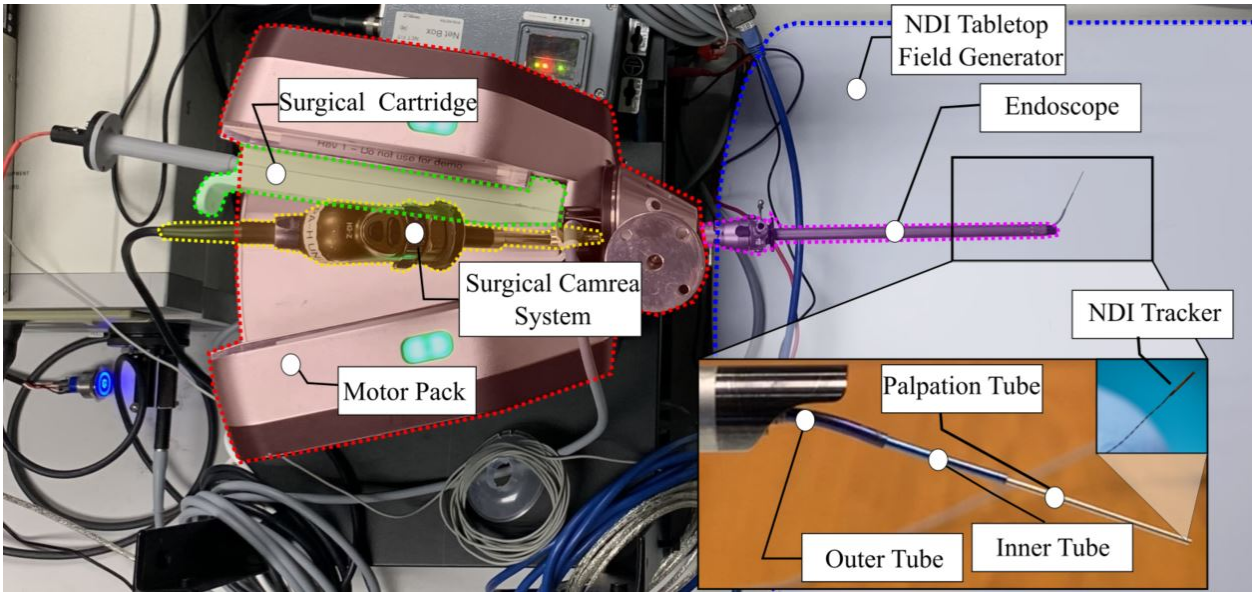


Figure 6.2: This figure shows a top-down view of the robot system used in this work. The motor pack is highlighted in red; the surgical cartridge is highlighted in green, the endoscope is highlighted in purple, the NDI field generator is highlighted in blue, and the surgical camera (not used in this work) is highlighted in yellow.

The palpation tool consists of a thin super elastic Nitinol tube (ID 0.53 mm, OD 0.63 mm) with a magnetic tracking sensor embedded at the distal tip. The tool is inserted through the robot's inner tube and housed within a dedicated removable palpation cartridge. The outer and inner tubes cooperatively control the pose and extension of the palpation tube: the outer tube primarily governs orientation, while coordinated motion of both tubes regulates extension. The palpation tube is rigidly mounted to the actuator responsible for inner tube translation, such that its extension beyond the inner tube remains fixed during palpation. This extension length can be adjusted via a set screw, allowing the effective tip stiffness to be tuned. Figure 6.2 illustrates the experimental setup and key system components.

The palpation tube is selected such that its bending stiffness is comparable to the expected stiffness of the tissue, improving the conditioning of the force estimation problem. This compliance also enables measurable deflections without applying excessive forces that could damage tissue. In this work, a target transverse tip stiffness of approximately 0.20 N/mm was achieved by extending the tube 15 mm beyond the inner tube tip. This value was determined by probing the surface of a phantom tissue using a force sensor with a probe tip of similar diameter to the palpation tube. The phantom tissue, developed by the

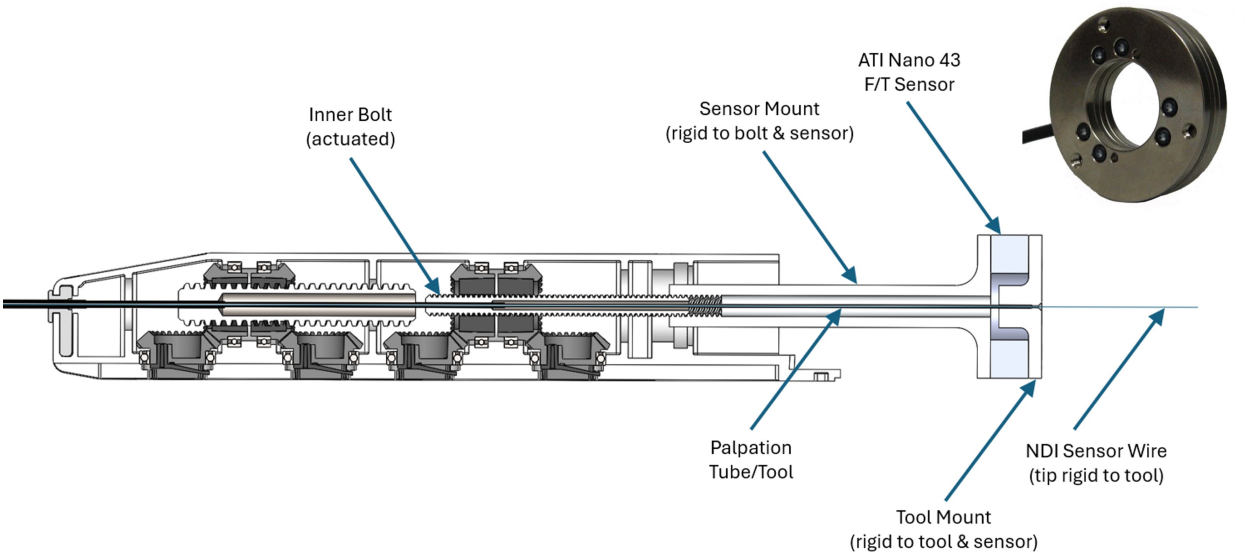


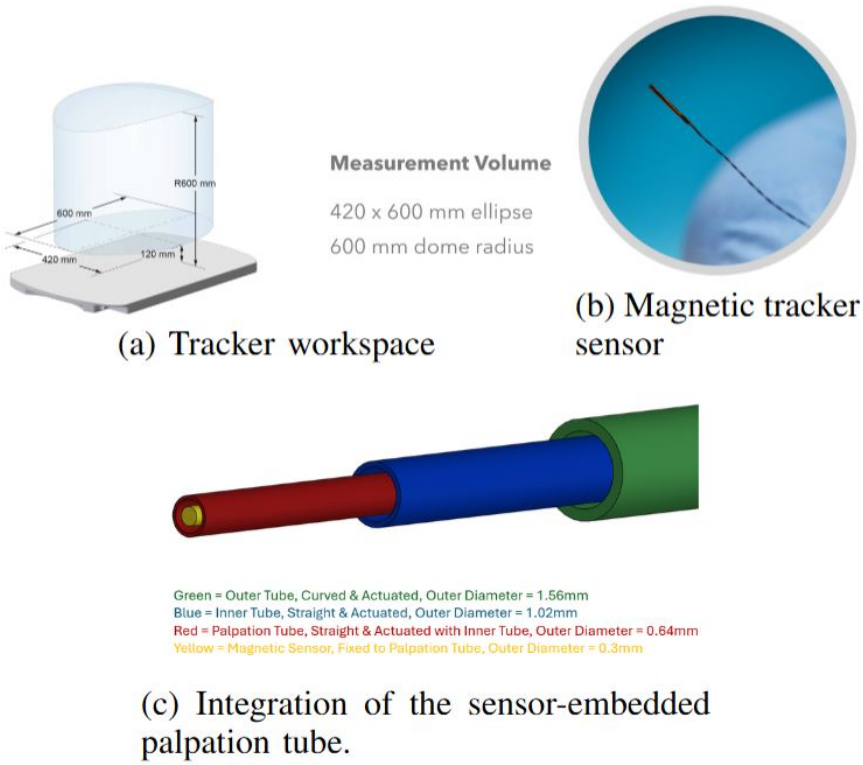
Figure 6.3: CAD drawing of palpation tool with the base force sensor mounted at the rear.

Simulation Innovation Laboratory at the University of Rochester Medical Center, is designed to mimic the mechanical properties of biological tissue [51]. Three-point bending tests were then conducted on the palpation tube to estimate its Young's modulus. Using this material property, the required extension length of the palpation tube was calculated, resulting in a value of 15 mm beyond the tip of the inner tube.

At the proximal end of the palpation tube, an optional mounting interface is provided for an ATI Nano43 six-axis force/torque sensor. This configuration was to enable direct measurement of forces along the axially stiff tangent direction of the tube, where deflection-based sensing is less effective due to minimal deformation. Although the sensor provides full six-axis measurements, its primary role in this application is axial force sensing. The sensor has a force resolution of 0.008 N and a full-scale range of ±36 N. A schematic of the cartridge and sensor integration is shown in Figure 6.3.

A 5-DOF magnetic tracking coil (0.3 × 2.5 mm, NDI Aurora; NDI, Waterloo, Ontario, Canada) is embedded at the distal tip of the palpation tube. A tabletop field generator (Fig. 6.2, 6.4) is positioned beneath the patient, providing a tracking volume with a 420 × 600 mm elliptical area and a dome height of 600 mm.

The tracking system provides 5-DOF pose measurements, including position and the local tangent direction of the tool. As summarized in Table 7.1, the system achieves a



(a) Tracker workspace

(b) Magnetic tracker sensor

(c) Integration of the sensor-embedded palpation tube.

Figure 6.4: Palpation sensors: (a) NDI tabletop field generator workspace. (b) NDI magnetic tracker sensor. (c) CAD drawing of palpation tool.

positional accuracy of 1.12 mm (precision 0.47 mm) and an orientation accuracy of 0.58° (precision 0.16°), according to manufacturer specifications. At the spatial scale of minimally invasive procedures, such as prostate surgery, orientation measurements are particularly valuable for characterizing tissue interaction. At the start of each experiment, an unloaded reference measurement is recorded. Subsequent measurements are expressed relative to this reference, and changes in position and tangent direction are used to estimate tip deflection. It is important to note that magnetic tracking accuracy can be affected by nearby metallic objects, particularly ferromagnetic materials that can cause significant distortion of the magnetic field. However, the Nitinol tubes used in this system are non-ferromagnetic and exhibit weak paramagnetic behavior, resulting in minimal perturbation of the magnetic field [21]. To experimentally verify this, a magnetic tracker was placed inside a 0.75 mm diameter hole with a depth of 5 mm in a 3D-printed fixture, both with and without a Nitinol sheath. The measured sensor positions in both configurations were recorded and compared, and the differences were found to be within the specified accuracy (1.8 mm when considering the

Table 6.1: Magnetic Tracking Accuracy (5DOF, 95% CI)

Metric	Value (mm)	Metric	Value (°)
Position Accuracy	1.12	Orientation Accuracy	0.58
Position Precision	0.47	Orientation Precision	0.16
Position Trueness	1.05	Orientation Trueness	0.49

tolerance of the hole). The error between measurements was 1.4 mm. These results indicate that the presence of the Nitinol tube minimally affects tracking accuracy.

Together, these components enable simultaneous measurement of tip deflection and axial force, forming the basis for the proposed force estimation framework.

6.2 Modeling

For real time force sensing we can formulate a standard rod model. Unlike the model described in Chapter 4 surgical robots, have negligible dynamics loading due to their comparatively small size, low distributed mass, and slow speeds. Furthermore, because the axial and shear stiffness is significantly larger than the bending stiffness we can reasonably assume Kirchhoff rod conditions. Therefore, we compute a quasi-static rod model of the concentric tube robotic system . Our implementation is adapted from [64] with modifications to improve computational efficiency. The model incorporates the geometric and material properties of each tube segment and accounts for the pre-curvature of the curved tube, bending and torsional stiffness, as well as external tip loads.

In addition to the assumptions described in [64], we make the following modeling assumptions for our quasistatic kinematic model:

No distributed loads: Because palpation primarily involves concentrated tip forces, distributed effects such as gravity and friction are neglected.

Small tip deflections: The tip deflection between evaluated configurations is assumed to be small enough that the internal force and moment at the base can be approximately computed from the applied tip wrench. This transforms the quasi-static Cosserat rod equations from a more complicated two-point boundary value problem to a much simpler initial value problem (IVP).

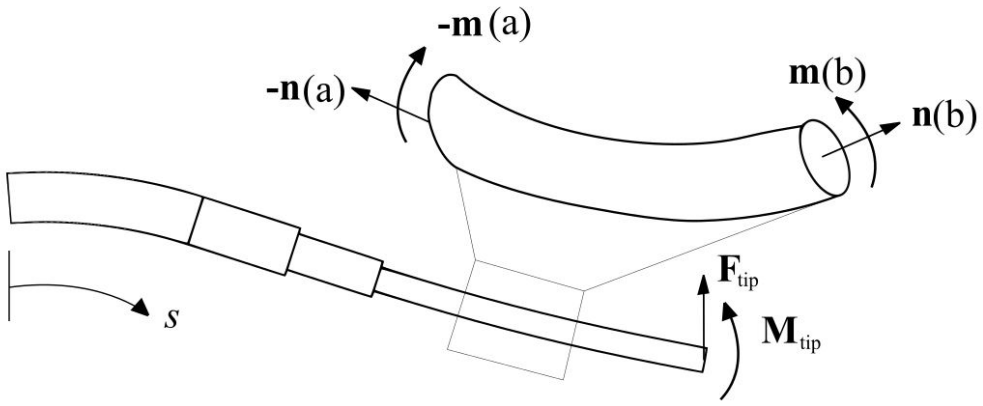


Figure 6.5: This figure shows a free body diagram of an arbitrary section of rod from a to b along s.

We can apply the same 3D Kirchhoff rod constraint equations shown in 4 for each tube in the robot as shown in 6.5. Therefore, each tube is described by an arc-length–parameterized space curve $\mathbf{p}(s) \in \mathbb{R}^3$, together with an orientation frame $\mathbf{R}(s) \in \mathrm{SO}(3)$. The frame's z-axis is aligned with the tangent direction to the curve, satisfying the kinematic relation

$$\mathbf{p}'(s) = \mathbf{R}(s)\mathbf{e}_3, \tag{6.1}$$

where $\mathbf{e}_3 = [0\ 0\ 1]^\top$ and the prime denotes differentiation with respect to arc length s. The rotational evolution of the frame is defined by

$$\mathbf{R}'(s) = \mathbf{R}(s)\hat{\mathbf{u}}(s), \tag{6.2}$$

, where the operator $\hat{(\cdot)}$ maps a vector to its corresponding skew-symmetric matrix in $\mathfrak{so}(3)$.

Following the linear constitutive law, the relationship between the strain vector and the internal moment vector at arc length s is given by

$$\mathbf{m}(s) = \mathbf{R}(s)\mathbf{K}(s)\left(\mathbf{u}(s) - \mathbf{u}^*(s)\right), \tag{6.3}$$

where $\mathbf{u}^*(s)$ denotes the intrinsic curvature of the rod in the undeformed state. The stiffness matrix is defined as

$$\mathbf{K}(s) = \begin{bmatrix} E(s)I_x(s) & 0 & 0 \\ 0 & E(s)I_y(s) & 0 \\ 0 & 0 & G(s)J(s) \end{bmatrix}, \tag{6.4}$$

where $E(s)$ is the Young's modulus, $G(s) = \dfrac{E(s)}{2(1+\nu)}$ is the shear modulus, $I_x(s)$ and $I_y(s)$ are the second moments of area about the x- and y-axes, respectively, and $J(s)$ is the polar moment of inertia.

Under the standard Kirchhoff-rod equilibrium relations [64], the internal force $\mathbf{n}(s)$ and moment $\mathbf{m}(s)$ satisfy

$$\mathbf{n}'(s) + \mathbf{f}(s) = \mathbf{0}, \tag{6.5}$$

$$\mathbf{m}'(s) + \mathbf{p}'(s) \times \mathbf{n}(s) + \mathbf{l}(s) = \mathbf{0}, \tag{6.6}$$

where $\mathbf{f}(s)$ and $\mathbf{l}(s)$ are distributed force and moment per unit length.

Because we assume only tip forces ($\mathbf{F}_{\text{tip}}$) and moments ($\mathbf{M}_{\text{tip}}$) act on the palpation tube, $\mathbf{f}(s) = \mathbf{0}$ and $\mathbf{l}(s) = \mathbf{0}$, so $\mathbf{n}(s)$ is constant and $\mathbf{m}(s)$ varies only with the moment arm. Consequently, the internal wrench at the base is related to the tip wrench by

$$\mathbf{n}_0 = \mathbf{F}_{\text{tip}}, \tag{6.7}$$

$$\mathbf{m}_0 = \mathbf{M}_{\text{tip}} + \mathbf{p}_{\text{tip}} \times \mathbf{F}_{\text{tip}}, \tag{6.8}$$

where $\mathbf{n}_0$ and $\mathbf{m}_0$ denote the internal force and moment at the base, respectively, and $\mathbf{p}_{\text{tip}}$ is the position vector from the base to the tip of the rod in the unloaded configuration.

In the Virtuoso system, a motor pack located behind the endoscope drives several transmission tubes that drive the Nitinol tubes at the tip of the endoscope. This transmission has some torsional compliance which allows twist when external loads are applied. We model this twist using a torsional compliance constant c (calibrated experimentally). If $m_{0,z}$ is the

torsional moment at the endoscope tip and θ is the outer tube rotation at the actuator, then θ_0 (the outer tube rotation at the endoscope tip) is

$$\theta_0 = \theta + c\, m_{0,z}. \tag{6.9}$$

6.3 Force Estimation Algorithm

We seek to estimate the unknown tip contact force $\mathbf{F}_{\text{tip}}$ from measured changes such as in tip position and tangent direction. Under a linearized compliance model, these measurements are related to the applied force by

$$\mathbf{A}_{pt}\widetilde{\mathbf{F}}_{\text{tip}} = \mathbf{z}, \tag{6.10}$$

where ($\widetilde{\ }$) indicates the model estimated values, $\mathbf{z}$ is the residual between the measured and modeled states, and $\mathbf{A}_{pt}$ is a Jacobian that linearly maps tip forces to $\mathbf{z}$.

We implemented two variants of solving for the estimated forces using least-squares-based algorithms. The first algorithm only considers input from the 5-DOF magnetic tracker in the tip of the tool. The second method incorporates data from a base force sensor.

Deflection-Only Approach

First, we will consider the case where only tip data is considered. Here, our measurement residual is the difference between the model position $\widetilde{\mathbf{p}}_{\text{tip}}$ and the measured position $\mathbf{p}_{\text{tip}}$ and the difference between the model's tangent vector $\widetilde{\mathbf{t}}_{\text{tip}} = \widetilde{\mathbf{R}}_{\text{tip}}\mathbf{e}_3$ and the measured position $\mathbf{t}_{\text{tip}}$:

$$\mathbf{z} \;=\; \begin{bmatrix} \mathbf{p}_{\text{tip}} - \widetilde{\mathbf{p}}_{\text{tip}} \\ \mathbf{t}_{\text{tip}} - \widetilde{\mathbf{t}}_{\text{tip}} \end{bmatrix}, \tag{6.11}$$

where $\mathbf{t}_{\text{tip}}$ is the tangent vector. We also define $\mathbf{A}_{pt}$ as a 5-DOF tip compliance matrix

$$\mathbf{A}_{pt} \;=\; \begin{bmatrix} \mathbf{C}_p \\ \mathbf{C}_t \end{bmatrix} \in \mathbb{R}^{6\times 3}, \tag{6.12}$$

where $\mathbf{C}_p$ and $\mathbf{C}_t$ are compliance matrices that map change in force to change in the tip position and tangent vector, respectively:

$$\begin{aligned} \mathbf{C}_p &= \frac{\partial \widetilde{\mathbf{p}}_{\text{tip}}}{\partial \widetilde{\mathbf{F}}_{\text{tip}}} \in \mathbb{R}^{3\times 3}, \\ \mathbf{C}_t &= \frac{\partial \widetilde{\mathbf{t}}_{\text{tip}}}{\partial \widetilde{\mathbf{F}}_{\text{tip}}} \in \mathbb{R}^{3\times 3}. \end{aligned} \tag{6.13}$$

Simulation results show that for tip deflections below 30°, the error associated with this linearization remains under 8%.

There are multiple ways to compute compliance in continuum robots, for example, finite differences or the derivative-propagation method [65]. In this chapter, we compute the compliance matrices using finite differences by sampling the forward model under small perturbations of the applied tip force. For a perturbation δ applied to the jth component of $\widetilde{\mathbf{F}}_{\text{tip}}$, the jth columns of $\mathbf{C}_p$ and $\mathbf{C}_t$ are given by

$$\mathbf{C}_{p,j} = \frac{\widetilde{\mathbf{p}}_j - \widetilde{\mathbf{p}}_0}{\delta}, \quad \mathbf{C}_{t,j} = \frac{\widetilde{\mathbf{t}}_j - \widetilde{\mathbf{t}}_0}{\delta}, \tag{6.14}$$

where $(\widetilde{\mathbf{p}}_0, \widetilde{\mathbf{t}}_0)$ and $(\widetilde{\mathbf{p}}_j, \widetilde{\mathbf{t}}_j)$ denote the nominal and perturbed model-predicted tip position and tangent vector, respectively.

The tip stiffness of a concentric tube robot can be much greater in some directions than others. In particularly stiff directions, displacement-based force estimation is much less accurate, or even altogether infeasible. In some cases (e.g. the Virtuoso robot) the robot design is such that the global z-axis is approximately the stiffest direction, but in general the stiff direction depends on the design of the tube pre-curvatures and the robot configuration. To systematically determine the stiff directions, we use a singular value decomposition of the tip compliance matrix $\mathbf{A}_{pt}$. As a motivating example, Table 6.2 shows the largest three singular values in a typical configuration of the Virtuoso robot. The smallest singular value is several orders of magnitude smaller than the dominant ones. If we were to directly estimate the 3D force by solving (6.10) (in a least-squares sense), small errors in sensed position would lead to huge errors in the estimated force (in the stiff direction), and the result would be unusable. Instead, we use the SVD to isolate the dominant compliance directions:

$$\mathbf{A}_{pt} \;=\; \mathbf{U}\,\mathbf{\Sigma}\,\mathbf{V}^{\mathsf{T}}. \tag{6.15}$$

The SVD is ordered such that singular values appear from largest to smallest in the diagonal Σ matrix, and their corresponding input and output directions are the columns of $\mathbf{U}$ and $\mathbf{V}$ from left to right, which allows us to isolate the dominant (non-stiff) force–deflection relationships in the system. To decide if a direction is unobservable, the expected sensor uncertainty (variance) can be divided by the singular value of the position compliance matrix. If the result is larger than an acceptable level of uncertainty (variance) in the force estimate, then it can be discarded from the estimation. In most CTR configurations, there will only be two acceptably large singular values. Thus, we arrange the force-displacement relationship in coordinates defined by the singular directions

$$\mathbf{U}^{\top}\mathbf{z} = \mathbf{\Sigma}\mathbf{V}^{\top}\widetilde{\mathbf{F}}_{\text{tip}}, \tag{6.16}$$

and then select only the two significantly compliant directions by multiplying by a selection matrix:

$$\mathbf{P}\mathbf{U}^{\top}\mathbf{z} = \mathbf{P}\mathbf{\Sigma}\mathbf{V}^{\top}\widetilde{\mathbf{F}}_{\text{tip}}, \tag{6.17}$$

where

$$\mathbf{P_c} \;=\; \begin{bmatrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \end{bmatrix}. \tag{6.18}$$

This creates a well-conditioned system of two equations in the three unknowns of $\widetilde{\mathbf{F}}_{tip}$. We can now use the minimum norm solution given by the pseudo-inverse to estimate the $\widetilde{\mathbf{F}}_{tip}$ that corresponds to the 5-DOF sensor reading $\mathbf{z}$:

$$\widetilde{\mathbf{F}}_{\text{tip}} \;=\; \left(\mathbf{P_c}\,\mathbf{\Sigma}\,\mathbf{V}^{\mathsf{T}}\right)^{\dagger}\mathbf{P_c}\,\mathbf{U}^{\mathsf{T}}\,\mathbf{z}. \tag{6.19}$$

Because of the selection matrix, force components along the stiff axial direction of the robot tip are unobservable and therefore not recovered by this approach. As a result, the estimated force effectively assumes zero contribution in the axial direction, leading to a systematic underestimation of the true force magnitude. Nevertheless, the remaining transverse force

Table 6.2: Example of the tip compliance matrix C. The inverse of each value corresponds to the principal stiffness along that mode.

Singular value (σ_i)	Magnitude [mm/N]
σ_1 (most compliant)	34.64
σ_2 (moderate)	18.03
σ_3 (stiffest)	0.0011

components can still provide valuable information, particularly for palpation tasks. To mitigate the impact of this limitation, path planning strategies can be employed to orient the robot such that interactions occur primarily along more compliant, non-axial directions, where force sensing is more reliable. We investigate this further in Chapter 7.

The estimated tip force is updated and smoothed in time using a complementary filter formulation:

$$\widetilde{\mathbf{F}}_{k+1} = \alpha\, \widetilde{\mathbf{F}}_{\text{tip}} + (1-\alpha)\, \widetilde{\mathbf{F}}_{k-1}, \tag{6.20}$$

where $\alpha \in [0, 1]$ determines the weighting between the measured and estimated signals in the complementary filter.

Deflection with Base Force Approach

While the deflection-only approach relies on discrepancies between model-predicted and measured tip position and orientation to infer contact forces, it suffers from limited observability in the stiffer directions. To address this limitation, we formulate a model to incorporate base force sensor measurements, enabling full three-degree-of-freedom force estimation at the tool tip. In this enhanced sensing configuration, the residual is reformulated to include the error between the measured and predicted forces at the base of the robot.

$$\mathbf{z} = \begin{bmatrix} \mathbf{p}_{\text{tip}} - \widetilde{\mathbf{p}}_{\text{tip}} \\ \mathbf{t}_{\text{tip}} - \widetilde{\mathbf{t}}_{\text{tip}} \\ F_b - \widetilde{F}_b \end{bmatrix}, \tag{6.21}$$

The associated augmented sensitivity matrix is

$$\mathbf{A}_{pt} = \begin{bmatrix} \mathbf{C}_p \\ \mathbf{C}_t \\ \widetilde{\mathbf{t}}_{\text{tip}}^{\mathsf{T}} + \widetilde{\mathbf{F}}_k^{\mathsf{T}} \mathbf{C}_t \end{bmatrix}. \tag{6.22}$$

The final row of $\mathbf{A}_{pt}$,

$$\widetilde{\mathbf{t}}_{\text{tip}}^{\mathsf{T}} + \widetilde{\mathbf{F}}_k^{\mathsf{T}} \mathbf{C}_t,$$

is derived from a first-order linearization of the base force response with respect to changes in the tip force. Here, $\widetilde{\mathbf{F}}_k$ maps changes in tip orientation (via $\mathbf{C}_t$) to changes in the axial base force, based on the static equilibrium model. This row captures how changes in tip force—especially along or against the tool's axial direction—translate to measurable changes in the base reaction force.

The system of equations is over-determined. We thus determine the most likely solution using a weighted least-squares approach where the equations are weighted with $\mathbf{W}$ whose values are determined by the inverse of the sensor noise covariance matrix:

$$\widetilde{\mathbf{F}}_{\text{tip}} = \left(\mathbf{A}^{\mathsf{T}} \mathbf{W} \mathbf{A}\right)^{-1} \mathbf{A}^{\mathsf{T}} \mathbf{W}\, \mathbf{z}. \tag{6.23}$$

This takes sensor accuracy into account in a statistically principled way. The estimation will rely on data according to its expected accuracy.

6.4 Initial Verification in Simulation

To evaluate the proposed algorithms, we developed a MATLAB-based simulation environment. A concentric tube robot is modeled using a static equilibrium forward kinematics formulation, with tool tip position controlled via a joystick input device. The simulated environment includes a compliant spherical object representing soft tissue. When the tool tip penetrates the sphere, a spring-based contact model generates a restoring force proportional

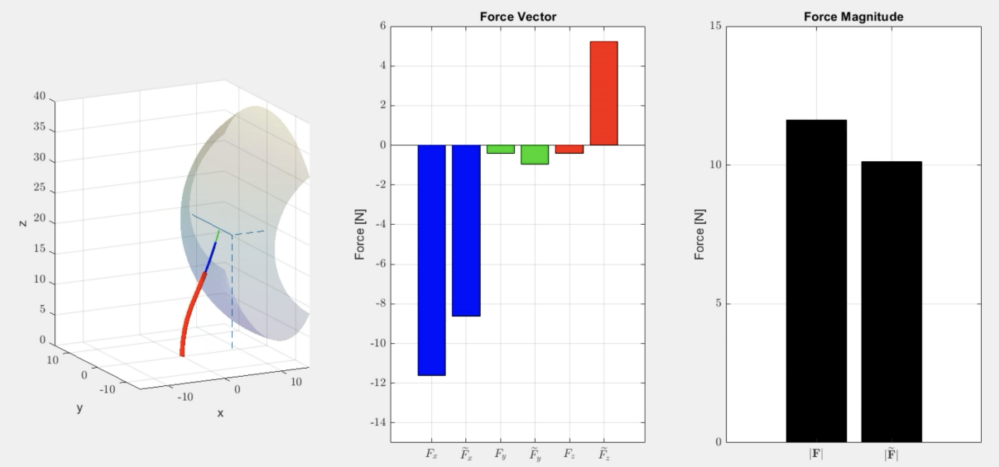


Figure 6.6: Simulation of deflection-based force estimation using only magnetic tracking sensor data.

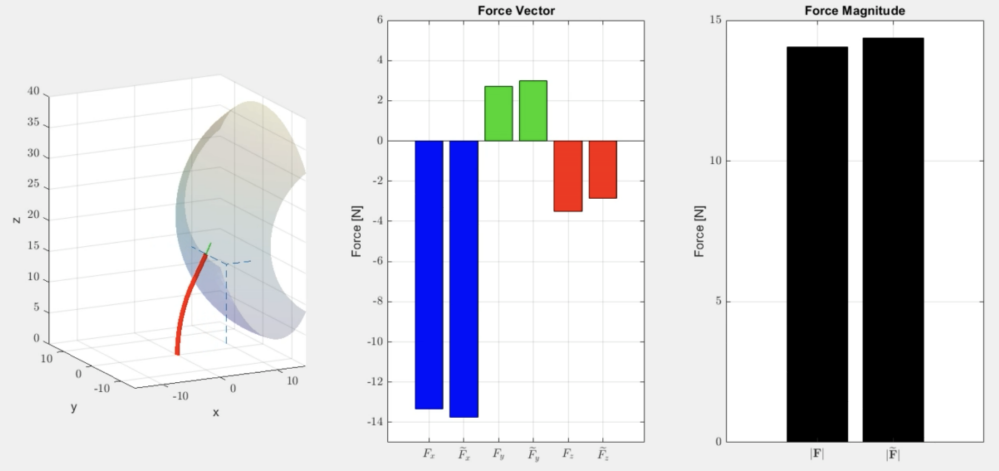


Figure 6.7: Simulation of force estimation fusing magnetic tracking data and base force sensor data.

to the penetration depth along the radial direction. This interaction force is applied to the robot model to simulate deformation.

Noisy measurements of tip position and orientation are generated by injecting zero-mean random noise consistent with sensor specifications. The proposed algorithm is then used to estimate the contact force from these simulated measurements.

For visualization, three subplots are defined. On the left is a workspace view showing the robot, tool tip, and spherical tissue; In the center, a bar chart comparing the true and estimated Cartesian force components $\big(F_x, F_y, F_z\big)$; and on the right, a bar chart comparing the true and estimated force magnitudes.

When relying solely on deflection-based sensing, the estimated axial force component deviates from the ground truth, as shown in Figure 6.6. However, the overall force magnitude remains reasonably accurate when a significant portion of the force lies in non-stiff directions. Incorporating base force measurements significantly improves estimation accuracy in all three dimensions, as demonstrated in Figure 6.7, where all force components are recovered with high fidelity. A more rigorous comparison of force estimation performance across the methods presented in this dissertation is provided in the following chapter.

These results demonstrate that incorporating base force sensing can substantially improve force estimation accuracy, particularly in configurations where axial forces are significant.

6.5 Experimental Validation

Preliminary Force Transmission Tests

During preliminary tests it was observed that the friction between the palpation tube and the inner tube was significant enough to obscure the readings on the base force sensor. Two tests were performed to quantify the friction force experienced by the palpation tubes. The first test had the palpation tube inserted into the robot and fixed to the force sensor. With the robot in the home configuration, an additional force sensor was used to push on the distal end of the palpation tube with forces ranging from 0.7-3.9 N. Data was read from both force sensors. It was observed that the force exerted by the distal force sensor was, on average, 7 times greater than the force experienced by the base sensor. An illustration of this test is shown in Figure 6.8.

Another test inserted and retracted the palpation tube in the robot. The force needed to break the static friction was then measured over varus curvatures. The range of axial force needed to break the static friction ranged from 0.2-0.6 N. An illustration of this test is shown in Figure 6.9. This frictional force is sometimes larger than the force exerted by the robot during palpation. Because of this, the subsequent experiments were performed without the use of the base force sensor. The next chapter addresses this problem.

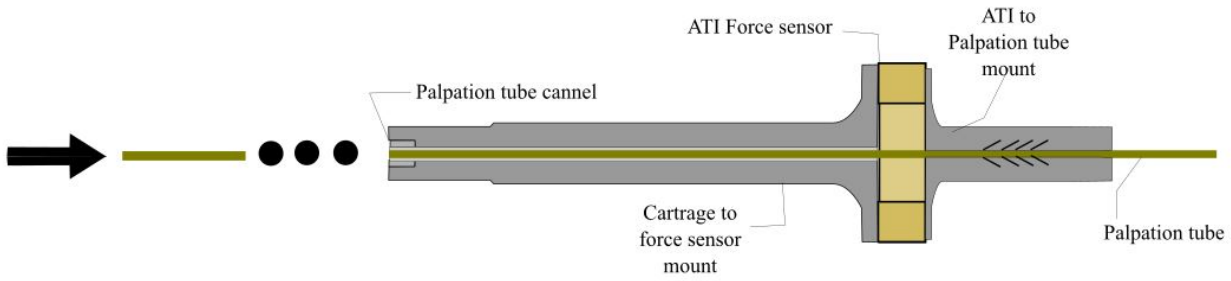


Figure 6.8: This figure shows a drawing of a sliced view of the proximal end of the robot when performing the first force transmission test.

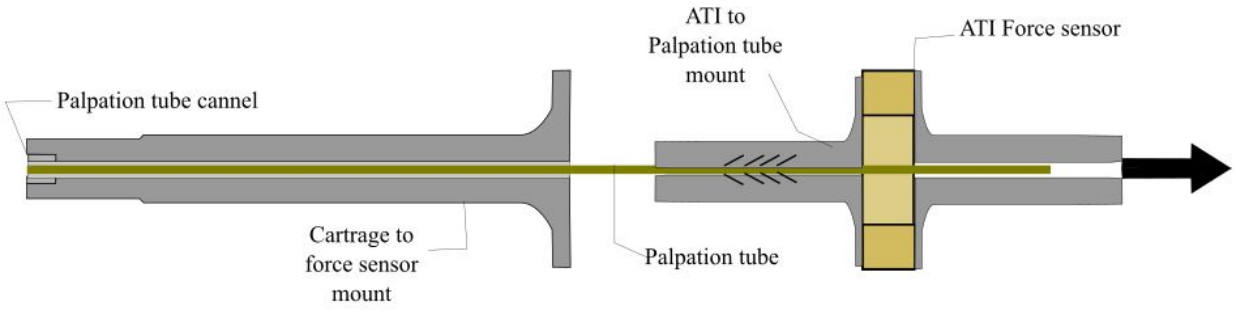


Figure 6.9: This figure shows a drawing of a sliced view of the proximal end of the robot when performing the second force transmission test.

Experiment setup

Following these initial tests, experiments were conducted to evaluate the robot's force estimation performance at multiple locations within its workspace.

For this purpose, a custom palpation apparatus was designed and 3D printed to rigidly mount an ATI Nano17 force/torque sensor (ATI Industrial Automation, Apex, NC, USA) at an adjustable distance from the endoscope tip (see Figure 6.10). The apparatus was clamped to the exterior of the endoscope sheath, through which the robot's concentric tubes were deployed.

The force sensor was fitted with a custom plate featuring a pattern of 13 holes, each with a diameter of 4 mm and a center-to-center spacing (pitch) of 5.5 mm. The holes were arranged along two perpendicular axes, with a shared central hole. Although the full pattern contains 13 holes, only five along the horizontal axis were used in this study: the central hole, along with two holes on either side. This subset was selected to avoid redundancy while remaining within the robot's dexterous workspace. Due to the symmetry of the setup,

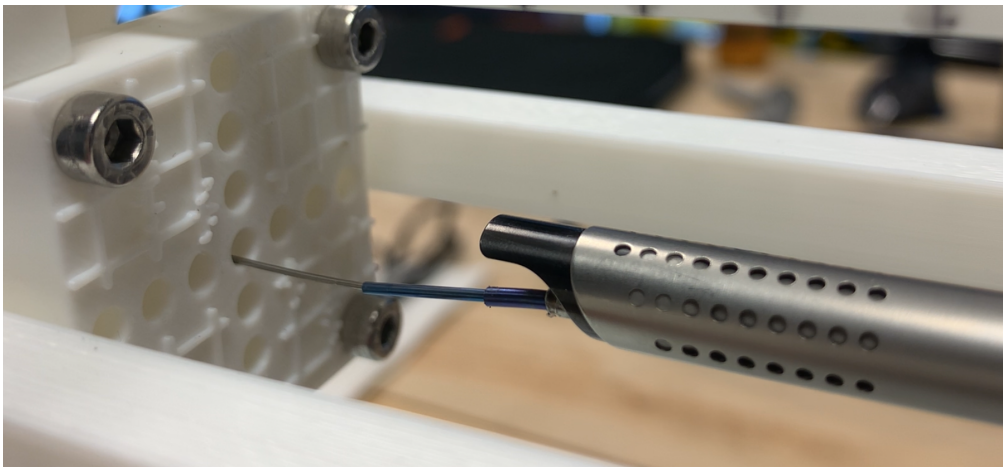


Figure 6.10: This figure shows a picture of our robot during palpation.

including the vertical axis would not provide additional information. An illustration of the robot interacting with these target locations is shown in Figure 6.11.

Data collection began with the palpation plate fixed 26 mm from the endoscope tip. For each hole, the following protocol was executed. First, the robot tip was positioned at the center of the hole. From this position, the robot was commanded to extend and apply force to the wall of the hole by translating 4–5 mm in one of four directions: up, down, left, or right, all parallel to the back wall of the hole. During each palpation, the forces measured by the external force sensor, along with the robot's tip position, orientation, and corresponding expected configuration, were recorded.

After each force application, the robot returned to the center position within the hole. To mitigate hysteresis effects, the robot was then fully retracted to its home position before returning to the center of the same hole and applying force in the next direction. This cycle of positioning, force application, and retraction was repeated for all four directions per hole (right, left, top, and bottom). Once all four directions were recorded for one hole, the process was repeated for the next hole in the sequence to ensure consistent and comprehensive coverage of the plate. This results in 20 unique palpation data points. A picture of the robot performing this palpation experiment is shown in Figure 6.10

To calibrate the robot's stiffness parameters, a subset of the collected data was used as a calibration set, while a separate subset was reserved for validation. Holes 0 through 2 were designated as the calibration set, and holes 3 and 4 were designated as the validation set.

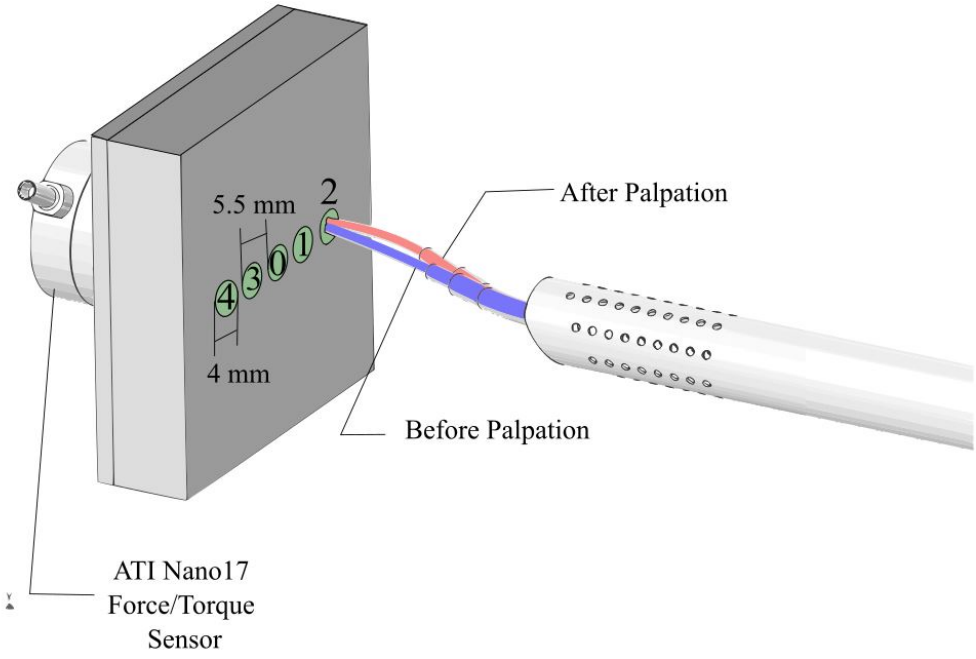


Figure 6.11: This figure shows an illustration of the palpation plate. The red indicates our palpation tube when deformed during palpation. The blue shows our palpation rod before palpation. The holes are indicated in green and labeled from 0 to 4. The diameter of each hole is 4 mm. The distance between each hole is 5.5 mm.

Model Calibration

Accurate force estimation requires an accurate kinematic model of the concentric tube robot. The mechanical properties of Nitinol exhibit significant variability in practice, and the torsional compliance of the transmission elements inside the endoscope is not known a priori. Therefore, the effective bending and torsional stiffness parameters and tube pre-curvature should be calibrated.

To identify these effective stiffness parameters, the calibration was formulated as a nonlinear least-squares minimization using the measured force magnitudes.

Let $\mathbf{F}_{\text{tip}}^{(m)}$ denote the ground-truth force measured by the reference sensor in experiment m, and let $\widetilde{\mathbf{F}}_{\text{tip}}^{(m)}(\boldsymbol{\theta})$ denote the model-predicted tip force for a given parameter vector $\boldsymbol{\theta} = [\alpha_b, c]^\top$, where α_b scales the nominal bending and torsional stiffness. The parameters were identified by minimizing the error in predicted force magnitude:

$$\boldsymbol{\theta}^\star = \arg\min_{\boldsymbol{\theta}} \sum_{m=1}^{M} \left(\|\widetilde{\mathbf{F}}_{\text{tip}}^{(m)}(\boldsymbol{\theta})\| - \mathbf{F}_{\text{tip}}^{(m)} \right)^2. \tag{6.24}$$

Table 6.3: Calibration results grouped by hole index (holes 0–2).

Hole	Mean $\lvert\mathbf{F}^{\text{est}}\rvert$ [N]	Mean $\lvert\mathbf{F}^{\text{gt}}\rvert$ [N]	MAE [N]
0	0.285	0.342	0.066
1	0.260	0.257	0.019
2	0.269	0.240	0.065
Overall	–	–	**0.055**

The optimization was solved using MATLAB's `lsqnonlin` with finite-difference Jacobians and converged after 27 iterations with a first-order optimality of 5×10^{-3}.

The resulting parameters were $\alpha_b^\star = 0.20$ and $c^\star = 0.10$ rad/(N·m). These parameters reflect additional compliance in the assembled instrument, including the transmission, bonding interfaces, and mechanical clearances, which are not explicitly captured by the nominal Cosserat-rod model.

Calibration performance The calibration data consisted of three hole locations (indices 0–2) in Figure 6.11 left. Table 6.3 summarizes the mean predicted and measured force magnitudes and the mean absolute error (MAE) across the four directions for each calibration hole. The overall mean error across all twelve calibration trials was 0.055 N. The influence of tracking error on force estimation is further examined in the next subsection through a sensitivity analysis.

6.5.1 Validation Results

A separate validation dataset was collected from two additional holes (indices 3–4 in Figure 6.11 left) to assess generalization. Figure 6.12 illustrates the comparison between the estimated (green) and measured (red) palpation force vectors in different experiments. Since the tool primarily palpates against the wall surface, the force vectors are projected onto the XY-plane to highlight their directional agreement at each palpation location. For better visualization, the vector lengths are uniformly scaled while preserving the relative orientations and magnitudes between the estimated and measured forces.

Using the same calibrated parameters, the model was then evaluated on two unseen holes (indices 3–4). The mean absolute error in force magnitude for the validation set was

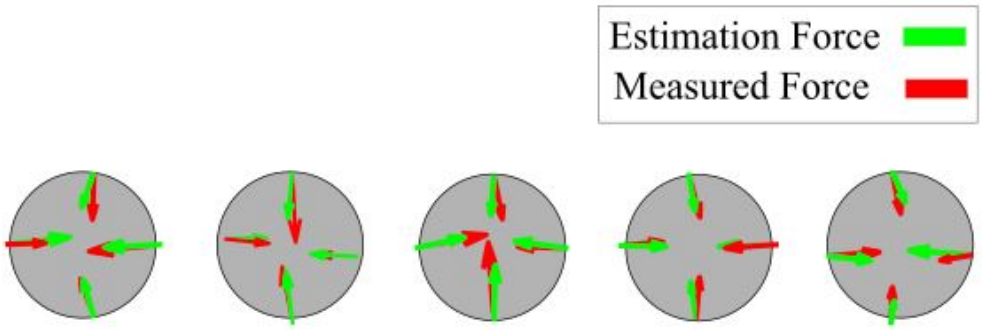


Figure 6.12: Figure shows the estimated (green) and measured (red) palpation force vectors on each palpation hole.

Table 6.4: Validation results at unseen holes 3–4 using calibrated parameters.

Hole	Mean $\lvert\mathbf{F}^{\text{est}}\rvert$ [N]	Mean $\lvert\mathbf{F}^{\text{gt}}\rvert$ [N]	MAE [N]
3	0.288	0.316	0.055
4	0.312	0.293	0.044
Overall	–	–	**0.049**

0.049 N, closely matching the calibration accuracy. This confirms that the identified stiffness parameters generalize well across workspace locations and contact directions.

To analyze directional accuracy, the estimated and ground-truth forces were decomposed into the local eigen-frame defined by the SVD of the compliance matrix. This reveals the accuracy along each principal stiffness direction. As shown in Table 6.5, the smallest error occurs along the first eigen-direction (most compliant bending axis), and the largest along the third eigen-direction corresponding to the (configuration-dependent) stiff axial direction which our algorithm effectively ignores. The pattern is consistent between calibration and validation datasets.

Both the calibration and validation results demonstrate consistent accuracy in estimating force magnitude in different locations and directions.

6.5.2 Simulation Environment

The physics-based, real-time simulation environment from [82] is used to simulate palpation of deformable tissue, shown in Figure 6.13. The deformable tissue is modeled with Extended Position-Based Dynamics (XPBD) and a Neo-Hookean strain energy density function [48].

Table 6.5: Mean absolute error (MAE) by eigen-direction in the tip frame.

Eigen-direction	Description	MAE [N]
	Calibration (holes 0–2)	
F_{eig1}	Primary bending (compliant)	0.0366
F_{eig2}	Secondary bending	0.0792
F_{eig3}	Axial (stiff)	0.1053
Force norm	Overall $\lvert\mathbf{F}\rvert$ error	**0.0554**
	Validation (holes 3–4)	
F_{eig1}	Primary bending (compliant)	0.0336
F_{eig2}	Secondary bending	0.0651
F_{eig3}	Axial (stiff)	0.1037
Force norm	Overall $\lvert\mathbf{F}\rvert$ error	**0.0494**

The quasi-static kinematic model of the concentric-tube robot (see Section 6.2) is implemented in the simulator. We again assume small relative tip deflections between time steps so that the force and moment at the base of the CTR is known, enabling a single forward integration pass to compute the full state of the CTR. Given the fast (∼1000 Hz) update rate of the simulator, this approximation is valid and significantly reduces computational effort. The CTR model in the simulator also supports point forces at any integration point along its length, so the base force and moment become:

$$\mathbf{n}_0 = \sum_i \mathbf{F}_i, \qquad \mathbf{m}_0 = \sum_i \mathbf{r}_i \times \mathbf{F}_i, \tag{6.25}$$

where subscript i denotes quantities at the i^{th} integration point. The forward integration pass must also then be split where there are nonzero point forces (in addition to at tube boundaries), as nonzero point forces result in a discrete change in the internal force.

The tip position of the CTR and the total force acting on the CTR are made available through a ROS interface, and perturbed using Gaussian noise according to Table 7.1 to mimic real-life data.

6.5.3 Stiffness Sensing in Simulation

Given our estimated force on the tip of the CTR and the total displacement of the tissue surface, we can estimate the stiffness seen at the contact area (in the direction of palpation

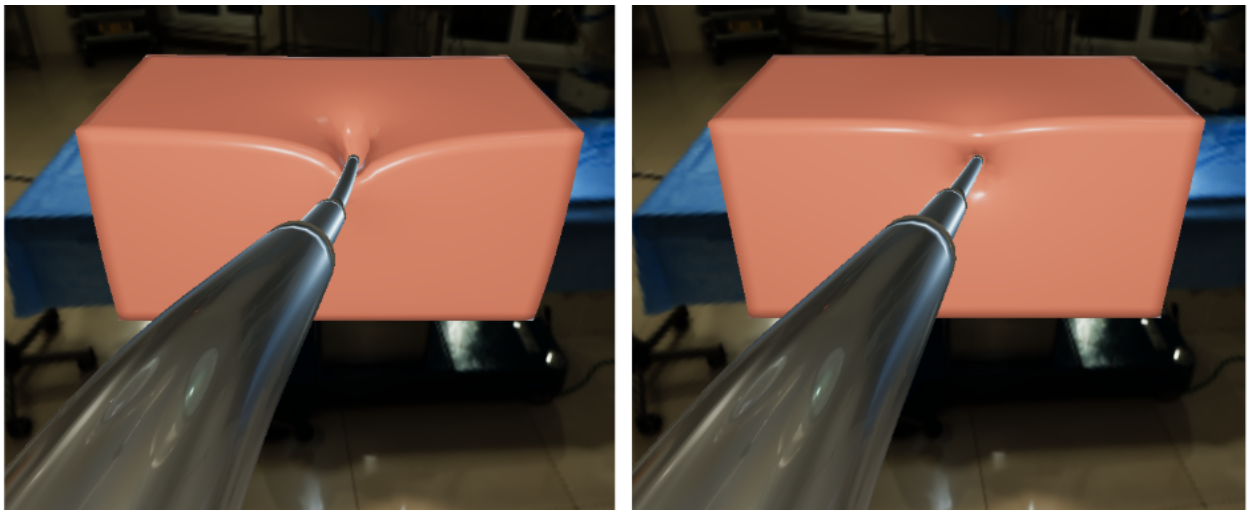


Figure 6.13: Screenshots from the simulator showcasing CTR interaction with deformable tissue. The tissue block is 2.5 cm $\times$ 1.25 cm $\times$ 1.25 cm.

- normal to the tissue surface) as the ratio of the force to tissue displacement $||\mathbf{F}||/||\mathbf{d}||$. We then do a linear regression on the force and displacement data during the palpation motion (line of best fit) to estimate the stiffness at the contact region. An example is shown in Figure 6.14.

By palpating multiple points on the tissue surface, we can generate a stiffness map of the tissue surface. A localized change in stiffness may indicate the boundary of an embedded object (e.g., a tumor), with different stiffness than the surrounding tissue. In simulation, we automatically palpated the top tissue surface (see Figure 6.13, left) in a 1 $\times$ 1 mm grid, covering the entire tissue surface. Palpation at a point on the tissue surface consists of a 4 mm uniform motion vertically downward into the tissue, a pause at the bottom, and a uniform motion upward out of the tissue. The simulated tissue block we palpated had two embedded spheres with 10 times higher stiffness than the surrounding tissue, shown in Figure 6.14, top-right. The top of the left sphere is 1.5 mm beneath the tissue surface, and the top of the right sphere is a little deeper at 2.3 mm beneath the tissue surface. The resulting stiffness map can be seen in Figure 6.14, bottom-right, and demonstrates our ability to detect stiff subsurface objects.

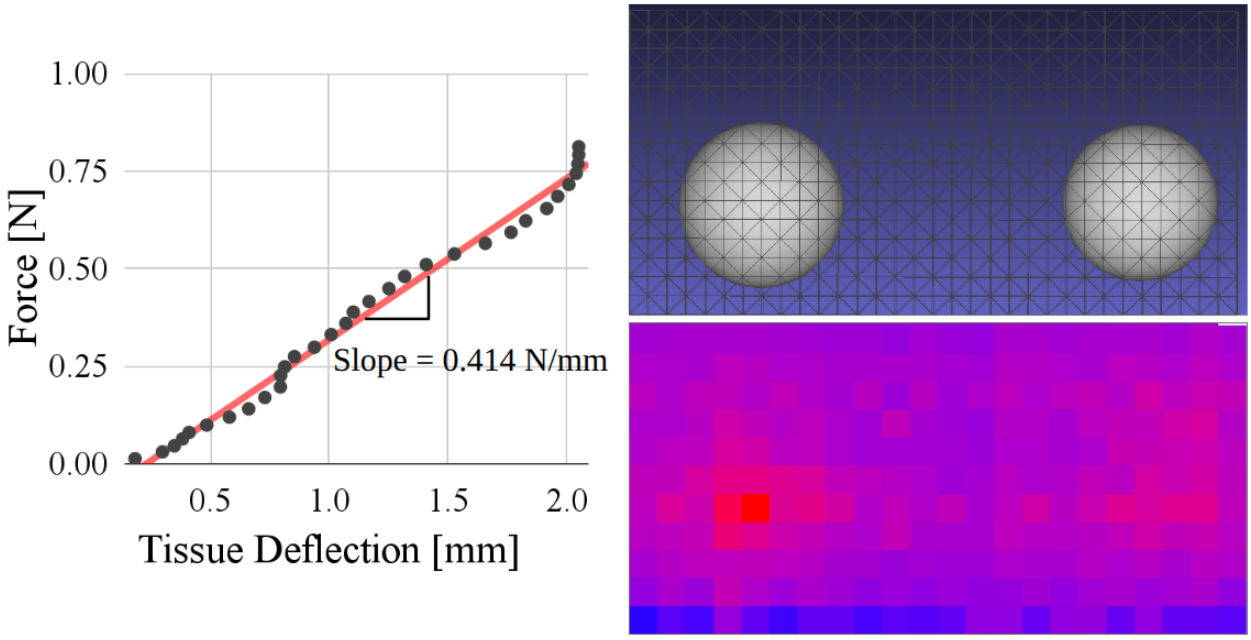


Figure 6.14: Tissue stiffness sensing via palpation in simulation. **Left:** Force–displacement curve for a representative palpation, where the slope defines local tissue stiffness. **Top Right:** Tissue block with two embedded stiff spherical inclusions. **Bottom Right:** Stiffness map obtained from vertical palpation on a 1×1 mm grid, revealing the embedded objects.

6.5.4 Phantom Stiffness Mapping

We finally demonstrate palpation-based stiffness mapping on phantom tissue to illustrate the feasibility of subsurface tumor localization. The phantom is a 20mm thick PVA hydrogel slab with a Young's modulus of 47 kPa, as experimentally measured using a Hertzian contact model. It also contains an embedded 3D-printed PLA cylindrical nodule with a height and diameter of 7 mm, buried 5mm below the surface. To construct the stiffness map, the robot is first actuated to bring the palpation tool into contact with the tissue surface. The the magnetic sensor data is recorded, and the tool tip is then commanded to palpate the tissue along the surface normal by 7 mm, after which the sensor data is recorded again.

Following each indentation, the robot returns to the home configuration and then repeats the procedure at the next location. A total of 90 points (arranged in a 9×10 grid) were sampled. Using the proposed algorithm, the force and stiffness at each point are computed. A smooth stiffness map of the region of interest was then generated over the dataset using a LOWESS (locally weighted regression) fit with a span of 0.3, as shown in Figure 6.15.

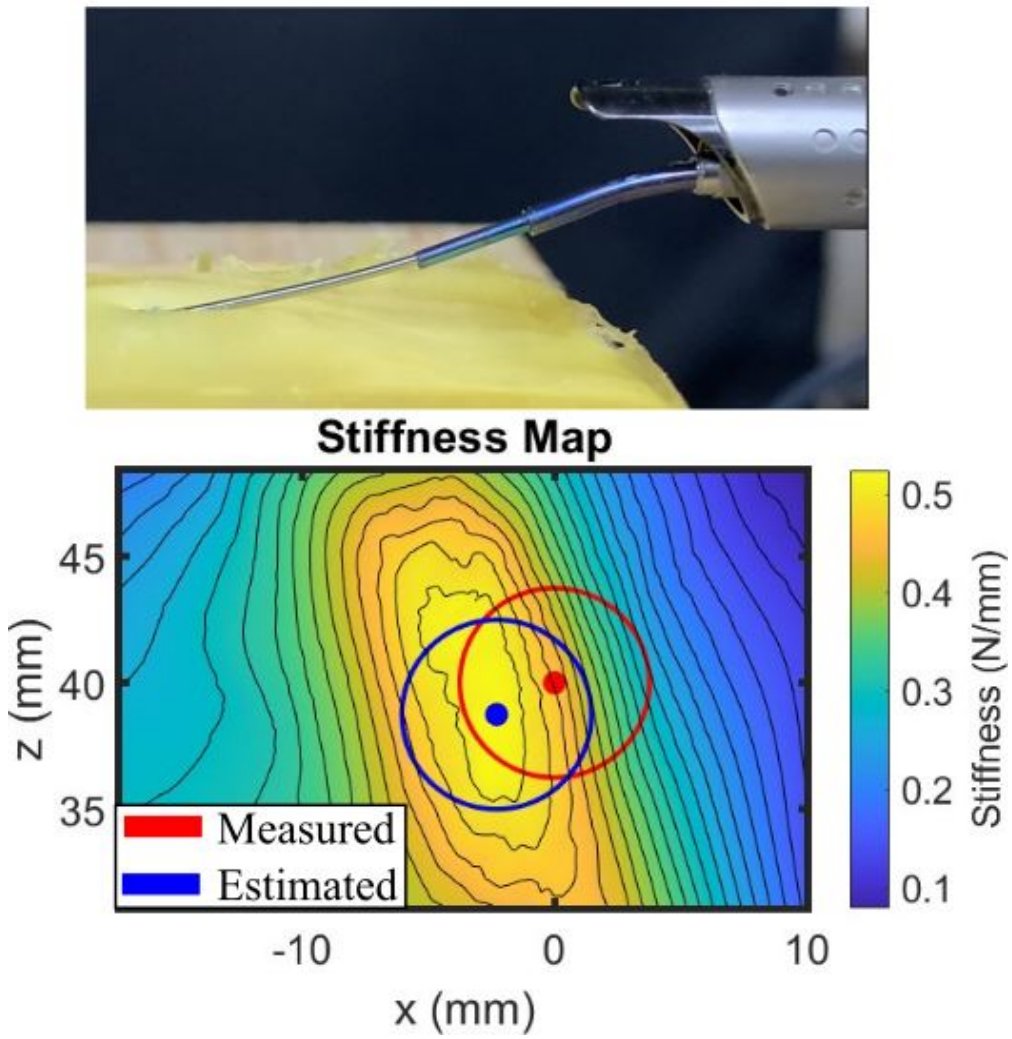


Figure 6.15: The top of figure shows the robot during palpation of the tissue phantom. The bottom panel presents a palpation-generated stiffness map of the phantom tissue containing an embedded nodule located 5 mm below the surface. The estimated tumor location is defined as the point of maximum stiffness and is indicated by a blue marker. The ground-truth tumor location is shown in red. Circles are overlaid on both to indicate the approximate tumor boundaries.

The estimated projected tumor centroid location is indicated by the region of maximum stiffness. A circle is overlaid to represent the projected dimensions of the tumor centered at this location. The ground-truth tumor center location and projection are shown in red. The center to center error between the two is 2.6 mm.

6.6 Chapter Summary

This chapter has presented the design and validation of a magnetic tracking–based palpation tool for force estimation in concentric tube robots. By integrating magnetic tracking with the CTR statics model, we estimated contact forces with a mean average error of 0.05 N. Evaluation using a leave-one-group-out cross-validation scheme yielded consistent accuracy across held-out palpation locations, demonstrating good generalization across the workspace. In addition to force estimation experiments, the method was further used to demonstrate palpation-based stiffness mapping on soft tissue phantoms and in a physics-based XPBD simulation environment to localize stiff subsurface structures. The Next chapter will focus on enhancing the palpation tool to enable stiffness estimation in the stiff direction.

Chapter 7

Three Axis Force Sensing and Tissue Palpation

In the previous chapter, we developed a palpation tool for the Virtuoso robotic system by inserting an additional Nitinol tube, with a magnetic tracker fixed at its tip, through the tool channel of the robot's inner tube and mounting a force sensor at its proximal end. When extended a preset length beyond the tip of the inner tube, this palpation tube exhibited lower bending stiffness than either of the concentric tubes in the Virtuoso system, allowing for increased deflection during tissue palpation. Tip forces were then inferred from this deflection using a mechanics-based model of the robot.

The axial stiffness of the palpation tube was significantly greater than its bending stiffness. As a result, the proximal force sensor was intended to measure tip forces transmitted along the tangent direction of the tube. While this system demonstrated the ability to sense forces in the flexible (bending) directions, friction along the tool channel rendered the force sensor ineffective over the expected force range encountered during palpation. This limitation prevented reliable force sensing in the stiff direction, which can constitute a significant portion of the applied forces depending on the robot's orientation. Additionally, the thin walls required to accommodate the magnetic tracker, while fitting within the tool channel, made the design mechanically fragile.

To address these limitations and improve measurement isotropy, we redesigned the palpation tool by incorporating a spring-suspended tip that allows displacement along the

local tangent direction. This spring-loaded mechanism reduces the effective compressive stiffness of the system, enabling improved force transmission to the sensor. The bending stiffness of the palpation tube was also reduced to increase force sensitivity and better match the stiffness of the tissues being palpated. Collectively, these modifications result in a more isotropic sensor capable of measuring forces with comparable sensitivity in all three spatial directions.

Modifications to the previous mechanics model are also developed to fully leverage the capabilities of the new design. This approach is validated through both simulation and experimental studies. In simulation, we compare the force-sensing performance of the previous and proposed designs when given an applied tip force that interacts with the tip at various orientations. In physical experiments, we evaluate the accuracy of force estimation by pressing the robot against a reference force sensor. We further demonstrate the system's ability to detect subsurface tumors with clinically relevant stiffness contrasts. Finally, this work presents the first demonstration of a concentric tube robot performing tumor localization through surface palpation of a tissue-mimicking phantom with hard embedded subsurface nodules.

7.1 Methodology

This chapter uses the same concepts discussed in the previous chapter. As before, we design and implement a new palpation tool that is compact and able to be integrated with the Virtuoso Endoscopy System. Our tool uses real time magnetic tracking of the tool tip in combination with a physics based model of the Virtuoso system to infer contact forces from observed deviations from model predicted tip states.

7.1.1 Hardware

As in the previous chapter, the proposed palpation system is designed for integration with the Virtuoso surgical concentric tube robot platform. The Virtuoso system provides a dual-arm robotic platform for minimally invasive surgery, in which each arm consists of a concentric tube manipulator deployed through an endoscope. Each manipulator comprises a pre-curved

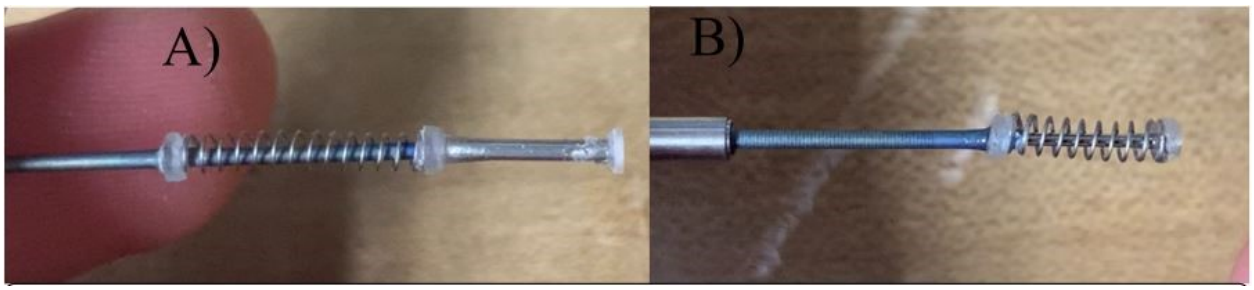


Figure 7.1: Palpation tool prototypes are shown. A) shows a design that translates along the outside of the inner tube of the Virtuoso robot. B) show a design that translates along the inside of the inner tube of the Virtuoso robot.

Nitinol outer tube (ID 1.19 mm, OD 1.56 mm) and a straight Nitinol inner tube (ID 0.84 mm, OD 1.02 mm), both of which can be independently translated and rotated via a proximal motor pack. These manipulators are housed within a removable cartridge.

The robot's motion is generated through coordinated actuation of these tubes: extension of the pre-curved outer tube induces bending, while rotation of the outer tube changes the plane of curvature. Combined with insertion and retraction of the inner tube, this results in a conical workspace that enables dexterous tip positioning. In this chapter, one of the Virtuoso cartridges is modified to serve as a dedicated palpation tool.

In addition to the Virtuoso platform, the system can be integrated with a KUKA LBR Med14 robotic manipulator, a collaborative robot certified for use in medical applications, with a repeatability of ± 0.1 mm to ± 0.15 mm. While the previous chapter considered only the Virtuoso system, some of the experiments presented here utilize the LBR Med manipulator due to its repeatability.

7.1.2 Tool Design

Spring Tip Design

Because the axial stiffness of the inner tube is significantly greater than its bending stiffness, force sensing based on axial deformation is challenging, particularly under the dimensional constraints of the robot. As a result, there exists a stiff direction (identified through the compliance matrix in the mathematical model) in which forces are effectively unobservable. Consequently, force components along this direction are not captured, leading to a systematic

underestimation of the true reaction force. Depending on the robot orientation and contact conditions, this missing component can constitute a significant portion of the total force.

To address this limitation, we modify the robot tip by incorporating a spring-loaded plunger mechanism that enables compliant motion along the axial direction. Specifically, a surgical cartridge is adapted such that the tip includes a spring and plunger assembly, allowing axial compression under applied load. Two design configurations are considered, as shown in Figure 7.1: an outer plunger design (A), in which the plunger slides along the exterior of the inner tube, and an inner plunger design (B), in which the plunger is housed within the inner tube.

The plunger position can be measured either through embedded sensing at the tip or, in the future, via computer vision using the endoscopic camera. In this chapter, we focus on the embedded magnetic tracking approach, while vision-based methods are left for future work. Although design (A) offers increased internal space for additional tools, its larger profile reduces compactness; therefore, design (B) is used throughout this study.

In design (B), the plunger consists of a stainless steel tube (OD 0.71 mm, ID 0.51 mm) inserted into the distal end of the inner tube. A blunt plastic tip is affixed to the distal end of the plunger to prevent tissue puncture and to house the magnetic tracking coil used for tip position and orientation sensing. A plastic collar is fixed to the exterior of the inner tube, and a helical spring (selected to match the expected tissue stiffness) is attached between the collar and the plastic tip. This arrangement allows the spring to compress when the plunger is driven into the inner tube under axial load.

Robot Bending Compliance Adjustment

While the Virtuoso robot exhibits structural compliance, the stiffness of the robot can still be significantly larger than that of the soft tissue being palpated. To improve sensitivity to tissue interaction forces, the bending stiffness of the distal inner tube is reduced using laser patterning. As shown in Figure 7.2, a patterned section is introduced near the distal region of the tube to locally reduce bending stiffness while maintaining structural integrity.

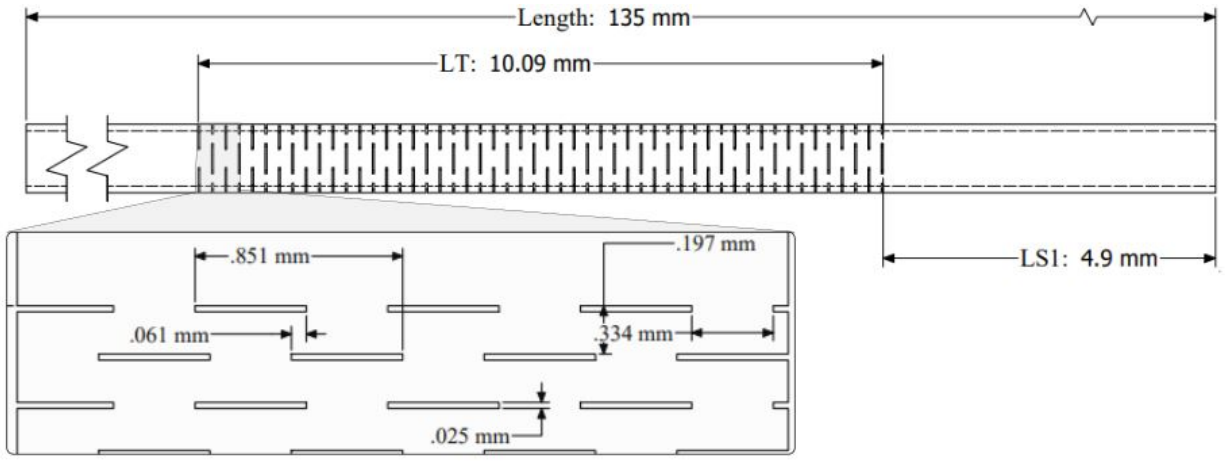


Figure 7.2: This figure shows a drawing of the laser patterned inner tube.

To design the compliance of the palpation tool such that it matches the stiffness of the target tissue, we first characterized the stiffness of both the tissue and the concentric tubes used in the Virtuoso robot.

The stiffness of the inner and outer tubes was measured using a three-point bending test. Each tube was supported by a jig with contact points spaced 37 mm apart, and a force sensor was used to apply a load at the midpoint. The resulting force–deflection data were recorded and used to estimate the effective Young's modulus. The inner tube exhibited a Young's modulus of 1928.6 N/mm^2, while the outer tube exhibited a modulus of 10888.4 N/mm^2. These values were then used to estimate the overall bending stiffness of the robot through its compliance matrix, enabling comparison with the stiffness of the target tissue.

To characterize the tissue, we analyzed a prostate phantom developed by the Simulation Innovation Laboratory at the University of Rochester Medical Center, designed to replicate the mechanical properties of prostate tissue [51]. Surface stiffness measurements were obtained by probing the phantom with a force sensor using a tip of the same diameter as the palpation tool, and recording the corresponding force–displacement response. This process was repeated for phantoms containing embedded tumors at depths of 5 mm, 10 mm, and 15 mm to capture the range of stiffness variations encountered during palpation. The measured stiffness values ranged from 0.07 to 0.2 N/mm, which were used as the target design range for the palpation tool. Matching the tool stiffness to this range improves the conditioning of the force estimation problem.

Table 7.1: Magnetic Tracking Accuracy (5DOF, 95% CI)

Metric	Value (mm)	Metric	Value (°)
Position Accuracy	1.12	Orientation Accuracy	0.58
Position Precision	0.47	Orientation Precision	0.16
Position Trueness	1.05	Orientation Trueness	0.49

Assuming a laser patterned section of 15 mm (chosen so that the robot can perform palpation tasks within the robot's visual work space), the desired robot bending stiffness reduction was determined to be 70-85%. This was achieved by laser patterning a brickwork pattern shown in Figure 7.2, designed using an iterative finite element analysis approach detailed in [15].

Magnetic tracker

To provide tip sensory information, we use an NDI Aurora tracking coil (NDI; Waterloo, Ontario, Canada) secured at the tip of the plunger. This is the same sensor system discussed in the previous chapter, however, we here summarize the specifications. This sensor measures 5 degrees of freedom (DOF). 3 of the measured DOF are used to reconstruct the cartesian coordinates of the sensor. 2 of the measured DOF are used to reconstruct the Pitch and Yaw of the sensor, with the exclusion of the Roll DOF.

As shown in Table 7.1, the positional accuracy and precision of the tracking system are 1.12 mm and 0.47 mm, respectively. Similarly, the orientation accuracy and precision are 0.58° and 0.16°, respectively.

The tracker wires are routed through the tool channel of the inner tube. To prevent damage to the wires and avoid interference with the motion of the plunger, a wire tensioning mechanism was designed and integrated at the base of the cartridge. A schematic of this device is shown in Figure 7.3. The tensioner is mounted at the proximal end of the cartridge, where the force sensor described in the previous section was located.

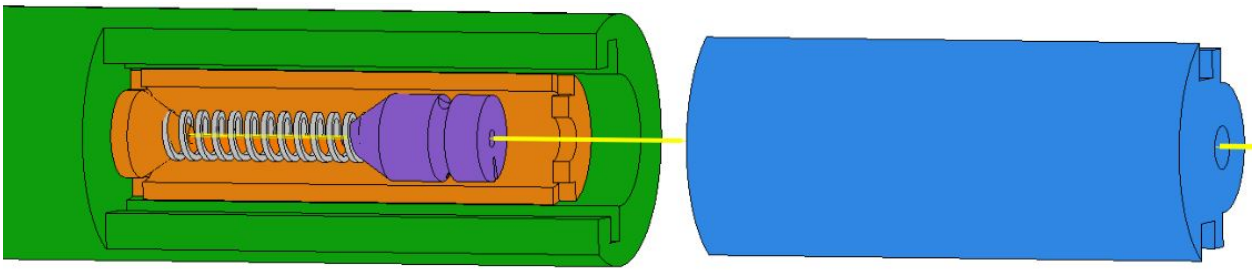


Figure 7.3: This figure shows a 3D drawing of the wire tensioner device. The Green shows the housing structure of the device, the orange shows the spring holder, the purple shows the piston, the blue shows the cover and the yellow shows the NDI wires.

The tensioning mechanism consists of a housing, spring holder, spring, piston, and cover. The piston is attached to the wire and compresses the spring, maintaining a tension and thereby improving the durability of the sensor wiring. The spring used in this mechanism is significantly more compliant than the tip plunger spring to ensure that it does not interfere with the force sensing behavior of the system.

The NDI sensor works in conjunction with a table top generator that is placed beneath the workspace of the robot. The field generator provides a 420 $\times$ 600 mm elliptical tracking volume with a height of 600 mm.

We note that our tumor localization experiments relied on magnetic tracking for the ground-truth tumor location as well as the force sensing and tool localization. The expected agreement between two magnetic sensor measurements at the same location is 1.6 mm (calculated by combining the two independent sensor accuracies in quadrature).

7.1.3 Kinematics Algorithm

Rod Model

To perform real time force sensing we developed a mathematical model using a Kirchhoff rod approach that can capture the kinematics, internal elastic deformation, and external tip loads. We will first detail the model of the robot without a spring tip plunger. This model is identical to the model described in the previous chapter, which is an adapted version of the model in [64], with modifications to improve computational efficiency.

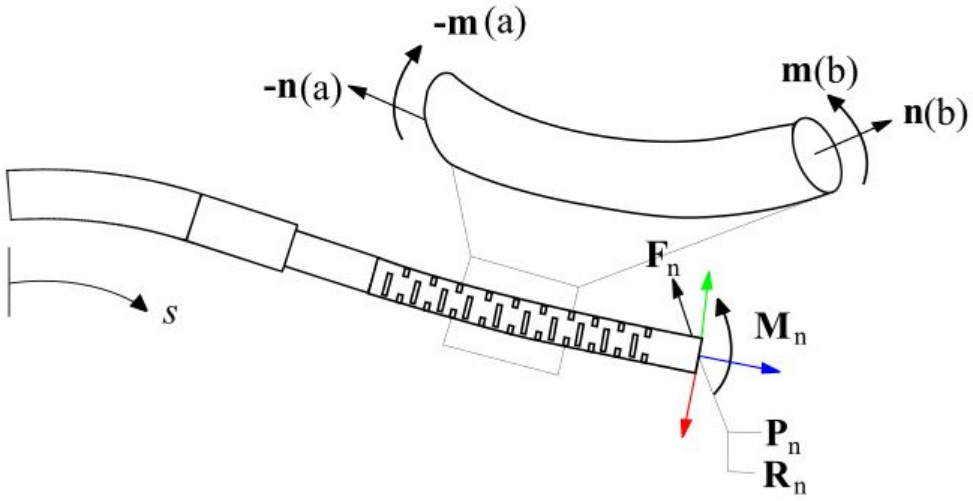


Figure 7.4: This figure shows a diagram of the internal forces and moments along a section of the robot without the plunger at the tip.

We then modify this model to incorporate the mechanics of the spring tip. The entire model is described here for completeness, however, because the models are initially identical the reader can skip to 7.1.3 if the previous model is well understood.

Both the models (with and without the spring tip) assume that forces and moments are concentrated at the tip of the manipulator. Additionally, gravitational effects are assumed to be negligible because of the size of the manipulator, and friction is not modeled here. To simplify the model further and reduce the model from a two-point boundary value problem to an initial value problem (IVP), we will make a small tip deflection assumption. i.e., the tip deflection between the evaluated configurations is assumed to be small enough so that the internal forces and moments at the base can be approximately computed from the applied tip wrench.

Each section of the robot manipulator can be described by a rod where the evolution of the centerline position of the material frame along the spatial dimension s of the rod described by $\mathbf{p}(s) \in \mathbb{R}^3$. Likewise, the orientation of the material frame along the rod is described by $\mathbf{R}(s) \in \mathrm{SO}(3)$. The z-axis of the orientation frames align tangentially with the rod. Thus, the spatial derivative of the position of the material frame can be described as

$$\mathbf{p}'(s) = \mathbf{R}(s)\mathbf{e}_3, \tag{7.1}$$

where $\mathbf{e}_3 = [0\ 0\ 1]^T$ and the prime denotes differentiation with respect to arc length s. We can also describe the spatial derivative of the material frame's orientation using the curvature of the rod $\mathbf{u}(s) \in \mathbb{R}^3$:

$$\mathbf{R}'(s) = \mathbf{R}(s)\widehat{\mathbf{u}}(s). \tag{7.2}$$

We use the $\widehat{(\cdot)}$ operator to map a vector to it corresponding skew-symmetric matrix in $\mathfrak{so}(3)$. We use the following constitutive law to describe the internal moments caused by the deformation of the manipulator from some initial pre-curvature, indicated by $(\cdot)^*$:

$$\mathbf{m}(s) = \mathbf{R}(s)\mathbf{K}(s)\left(\mathbf{u}(s) - \mathbf{u}^*(s)\right), \tag{7.3}$$

where the stiffness matrix is defined as:

$$\mathbf{K}(s) = \begin{bmatrix} E(s)I_x(s) & 0 & 0 \\ 0 & E(s)I_y(s) & 0 \\ 0 & 0 & G(s)J(s) \end{bmatrix}, \tag{7.4}$$

and $E(s)$ is the Young's modulus, $G(s) = \dfrac{E(s)}{2(1+\nu)}$ is the shear modulus, $I_x(s)$ and $I_y(s)$ are the second moments of area about the x- and y-axes, respectively, and $J(s)$ is the polar moment of inertia.

We sum the forces along a section of the quasi-static rod (from a point a to point b along a dummy variable σ) and represent them as

$$\int_a^b \mathbf{f}(\sigma)\, d\sigma + \mathbf{n}(b) - \mathbf{n}(a) = \mathbf{0}, \tag{7.5}$$

where the external distributed forces are represented by $\mathbf{f}(s) \in \mathbb{R}^3$, the internal forces are represented as $\mathbf{n}(s) \in \mathbb{R}^3$. In the same way, we sum the moments about an arbitrary point along the rod $\mathbf{p}_o$ from point a to point b yielding

$$\begin{aligned} \int_a^b \mathbf{l}(\sigma)d\sigma \ + \ \mathbf{m}(b) - \mathbf{m}(a) + \ (\mathbf{p}(b) - \mathbf{p}_o)\times\mathbf{n}(b) - \\ (\mathbf{p}(a) - \mathbf{p}_o)\times\mathbf{n}(a) = \mathbf{0}, \end{aligned} \tag{7.6}$$

where $\mathbf{l}(s) \in \mathbb{R}^3$ denotes the distributed external moment per unit length and $\mathbf{m}(s) \in \mathbb{R}^3$ is the internal moment. Taking the derivative of (7.5) and (7.6) with respect to s, we obtain the standard Kirchhoff rod equilibrium relations:

$$\mathbf{n}'(s) + \mathbf{f}(s) = \mathbf{0}, \tag{7.7}$$

$$\mathbf{m}'(s) + \mathbf{p}'(s) \times \mathbf{n}(s) + \mathbf{l}(s) = \mathbf{0}. \tag{7.8}$$

However, because we assume all external forces and moments are concentrated at the tip, $\mathbf{n}(s)$ remains constant along the manipulator. Additionally, $\mathbf{m}(s)$ varies only with the moment arm defined by $\mathbf{p}_\mathrm{n} - \mathbf{p}(s)$. Therefore, the internal forces and moments at the base of the robot can be represented as:

$$\mathbf{n}_0 = \mathbf{F}_\mathrm{n}, \tag{7.9}$$

$$\mathbf{m}_0 = \mathbf{M}_\mathrm{n} + \mathbf{p}_\mathrm{n} \times \mathbf{F}_\mathrm{n}, \tag{7.10}$$

Where $\mathbf{F}_\mathrm{n} \in \mathbb{R}^3$ and $\mathbf{M}_\mathrm{n} \in \mathbb{R}^3$ are the applied forces and moments at the end of the rod ($\mathbf{p}_\mathrm{n} \in \mathbb{R}^3$), respectively. $\mathbf{n}_0 \in \mathbb{R}^3$ and $\mathbf{m}_0 \in \mathbb{R}^3$ indicate the internal forces and moments at the base of the manipulator, respectively.

We also model transmission torsional compliance using a constant c (calibrated experimentally). If $m_{0,z}$ is the torsional moment at the endoscope tip and θ is the outer tube rotation at the actuator, then θ_0 (the outer tube rotation at the endoscope tip) is

$$\theta_0 = \theta + c\, m_{0,z}, \tag{7.11}$$

Modeling Tip Plunger

We can adapt our robot model to incorporate the spring tip plunger design detailed in previous sections. For this addition, we will assume the plunger only deforms along the axis of the inner tube of our robot. This assumption is reasonable as the bending stiffness of the stainless steel tube used to make the plunger is much greater than that of the Nitinol inner tube. Thus, we will model this plunger as an axial spring attached to the tip of our

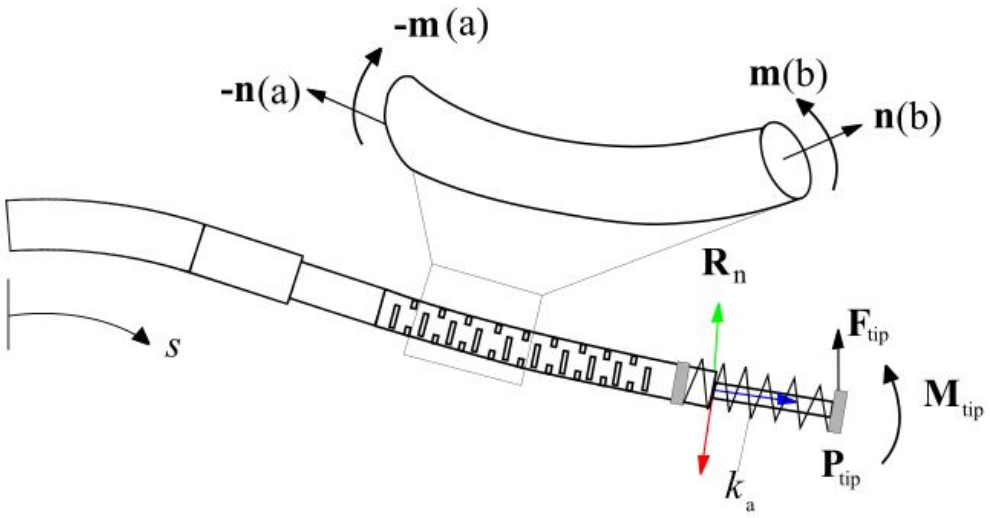


Figure 7.5: This figure shows a diagram of the internal forces and moments along a section of the robot with the plunger at the tip.

Kirchhoff rod model, where the new tip of our manipulator ($\mathbf{p}_{\text{tip}}$) is located at the end of the axial spring. For reference see Figure 7.5.

The Kirchhoff rod terminates at the base of the axial spring. We will indicate this termination point of the rod's arc length s by the subscript "n". The orientation of the point of termination of the rod $\mathbf{R}_{\text{n}}$ and that of the tip of the manipulator $\mathbf{R}_{\text{tip}}$ remain identical because the spring only compresses about the z-axis of the of the termination point of the rod described by: $\mathbf{R}_{\text{n}}\mathbf{e}_3$. We can find the tip position of the undeformed axial spring, indicated by the superscript $(\cdot)^*$, by translating along the z-axis of the termination frame by the length of the undeformed spring l_s:

$$\mathbf{p}^*_{\text{tip}} = \mathbf{p}_{\text{n}} + \mathbf{R}_{\text{n}}\mathbf{e}_3 l_s. \tag{7.12}$$

Similarly, the deformed tip can be found by applying a displacement (d_{s}) along the same axis:

$$\mathbf{p}_{\text{tip}} = (\mathbf{R}_{\text{n}}\mathbf{e}_3)d_{\text{s}} + \mathbf{p}^*_{\text{tip}}. \tag{7.13}$$

Assuming that the axial spring deforms linearly, we can write d_{s} as the component of some tip force ($\mathbf{F}_{\text{tip}}$) acting along the z-axis of the rod's termination frame divide by the stiffness

of the axial spring (k_a):

$$d_\mathrm{s} = \frac{\mathbf{F}_\mathrm{tip} \cdot \mathbf{R}_\mathrm{n}\mathbf{e}_3}{k_\mathrm{a}}. \tag{7.14}$$

Because the internal forces remain constant across the manipulator, the forces at the tip of the manipulator are equal to that at the termination point of the rod. However, the moment arm is affected by the compression of the distal spring. Therefore, we can represent the forces and moments at the termination point of the rod as

$$\begin{aligned} \mathbf{F}_\mathrm{n} &= \mathbf{F}_\mathrm{tip}, \\ \mathbf{M}_\mathrm{n} &= \mathbf{M}_\mathrm{tip} + (\mathbf{p}_\mathrm{tip} - \mathbf{p}_\mathrm{n}) \times \mathbf{F}_\mathrm{tip}. \end{aligned} \tag{7.15}$$

When substituting 7.15 into 7.9 and 7.10 we can easily calculate the forces and moments at the base of the robot, allowing us to integrate the shape and find the deflected tip location.

7.1.4 Force Estimation Algorithm

We use the model detailed above to estimate contact forces at the tip of our robot ($\widetilde{\mathbf{F}}_\mathrm{tip}$). This is done by establishing the following relationship between the estimated forces and the difference between estimated and measured tip positions

$$\mathbf{A}_{pt}\widetilde{\mathbf{F}}_\mathrm{tip} = \mathbf{z}$$

Where $\mathbf{A}_{pt}$ is a matrix that describes the relevant compliance of our robot and $\mathbf{z}$ contains the residual between measured and estimated states. For this work, we can measure the tangent vector and the position of the tip. Therefore, our residual is:

$$\mathbf{z} = \begin{bmatrix} \mathbf{p}_\mathrm{tip} - \widetilde{\mathbf{p}}_\mathrm{tip} \\ \mathbf{t}_\mathrm{tip} - \widetilde{\mathbf{t}}_\mathrm{tip} \end{bmatrix}, \tag{7.16}$$

where $\widetilde{\mathbf{p}}_\mathrm{tip}$ is the estimated tip position and $\widetilde{\mathbf{t}}_\mathrm{tip} = \mathbf{R}_\mathrm{tip}\mathbf{e}_3$ is the estimated tangent vector. We build $\mathbf{A}_{pt}$ with the relevant compliance matrices, such that the estimated forces are mapped

to the residuals:

$$\mathbf{A}_{pt} \;=\; \begin{bmatrix} \mathbf{C}_p \\ \mathbf{C}_t \end{bmatrix} \in \mathbb{R}^{6\times 3}, \tag{7.17}$$

where the description of the position compliance is contained in $\mathbf{C}_p = \frac{\partial \mathbf{p}_{\text{tip}}}{\partial \mathbf{F}_{\text{tip}}} \in \mathbb{R}^{3\times 3}$, and the description of the tangent vector compliance is contained in $\mathbf{C}_t = \frac{\partial \mathbf{t}_{\text{tip}}}{\partial \mathbf{F}_{\text{tip}}} \in \mathbb{R}^{3\times 3}$. These matrices can be obtained by the linear relation between force and displacement about the current state, calculated using the robot mechanics model in the current configuration:

$$\mathbf{C}_{p,j} = \frac{\mathbf{p}_j - \mathbf{p}_0}{\delta}, \quad \mathbf{C}_{t,j} = \frac{\mathbf{t}_j - \mathbf{t}_0}{\delta}, \tag{7.18}$$

where $(\mathbf{p}_0, \mathbf{t}_0)$ and $(\mathbf{p}_j, \mathbf{t}_j)$ denote the nominal and perturbed model-predicted tip position and tangent vector, respectively. The tip force is estimated using a weighted least-squares formulation, which allows the position and tangent errors to be scaled by appropriate nominal values. Let $\mathbf{W}$ denote the weighting matrix. The estimated tip force is then obtained as

$$\widetilde{\mathbf{F}}_{\text{tip}} \;=\; (\mathbf{A}_{pt}^T \mathbf{W} \mathbf{A}_{pt})^{-1} \mathbf{A}_{pt}^T \mathbf{W}\, \mathbf{z}. \tag{7.19}$$

The estimated tip force can then be temporally smoothed using a complementary filter:

$$\widetilde{\mathbf{F}}_{k+1} = \alpha\, \widetilde{\mathbf{F}}_{\text{tip}} + (1-\alpha)\, \widetilde{\mathbf{F}}_k, \tag{7.20}$$

where tilde $\widetilde{(\cdot)}$ denotes an instantaneous estimate computed from the current sensor measurement, and $\alpha \in [0, 1]$ is the filter constant determining the temporal smoothness.

7.2 Experiments

7.2.1 Simulation Experiments

To compare the methods presented in this dissertation, a series of simulations were conducted.

The first method, presented in Chapter 6, considers a palpation tool without the spring-loaded tip introduced in this chapter. As a result, the tool exhibits significantly higher stiffness close to the tangent of the tip, preventing reliable force estimation along this axis. To address this limitation, singular value decomposition (SVD) is used to identify the stiff direction and force components along this direction are effectively ignored.

The second method, also presented in Chapter 6, utilizes the same palpation tool as method 1, but incorporates a force sensor at the proximal end of the tool. This sensor is intended to measure forces transmitted along the tangent direction of the palpation tube. While this approach is theoretically sound, experimental results demonstrated that friction within the tool channel significantly attenuates force transmission, preventing reliable measurements within the force range typical of palpation. However for this simulation these friction forces are not considered, to evaluate an ideal case.

Finally, we consider the palpation tool introduced in this chapter, which incorporates a compliant, spring-loaded tip to enable force sensing in all directions.

For each method, the robot model was evaluated across multiple configurations. In Configuration 1, the robot is fully retracted, and only the palpation tool extends beyond the endoscope. In Configuration 2, both the inner and outer tubes are extended by 20 mm, aligning their distal tips. In Configuration 3, the inner tube is extended by 20 mm while the outer tube remains fully retracted within the endoscope.

At each configuration, simulated forces with a magnitude of 0.2 N were applied to the tip of the robot. This magnitude is consistent with the forces observed experimentally in Chapter 6. The applied force direction was varied within the bending plane of the robot, with angles ranging from -90° to $+90^\circ$. An angle of 0° corresponds to a force applied along the tangent direction of the tip, while $\pm 90^\circ$ correspond to forces applied perpendicular to the tangent, either opposing or aligning with the direction of curvature. An illustration of the force application is shown in Figure 7.6.

Table 7.2: Model Parameters

	EI (N mm^2)	GJ (N mm^2)	Curvature (rad/mm)
Inner Tube	1.68x10^3	1.30x10^3	0
Outer Tube	1.15x10^4	8.81x10^3	0.075
Palpation Tool	1.98x10^2	1.30x10^3	0

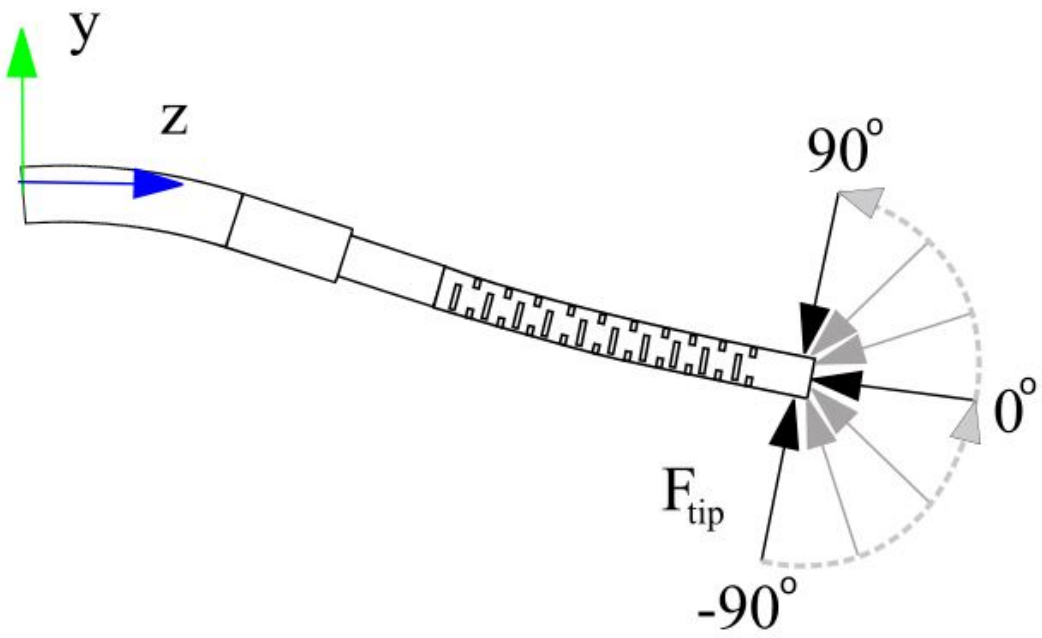


Figure 7.6: This figure shows an illustration of how the tip forces ($\mathbf{F}_{\text{tip}}$) are applied in simulation.

This experiment evaluates and compares the limitations of each algorithm under varying tip orientations during palpation. To isolate algorithmic performance, no internal friction, sensor noise, or modeling error is included in the simulation, allowing all methods to be assessed under idealized conditions.

The model parameters are consistent across all simulations and are listed in Table 7.2. These parameters were determined experimentally from the laser-patterned tube. In all cases, the length of the palpation tool is fixed at 20 mm.

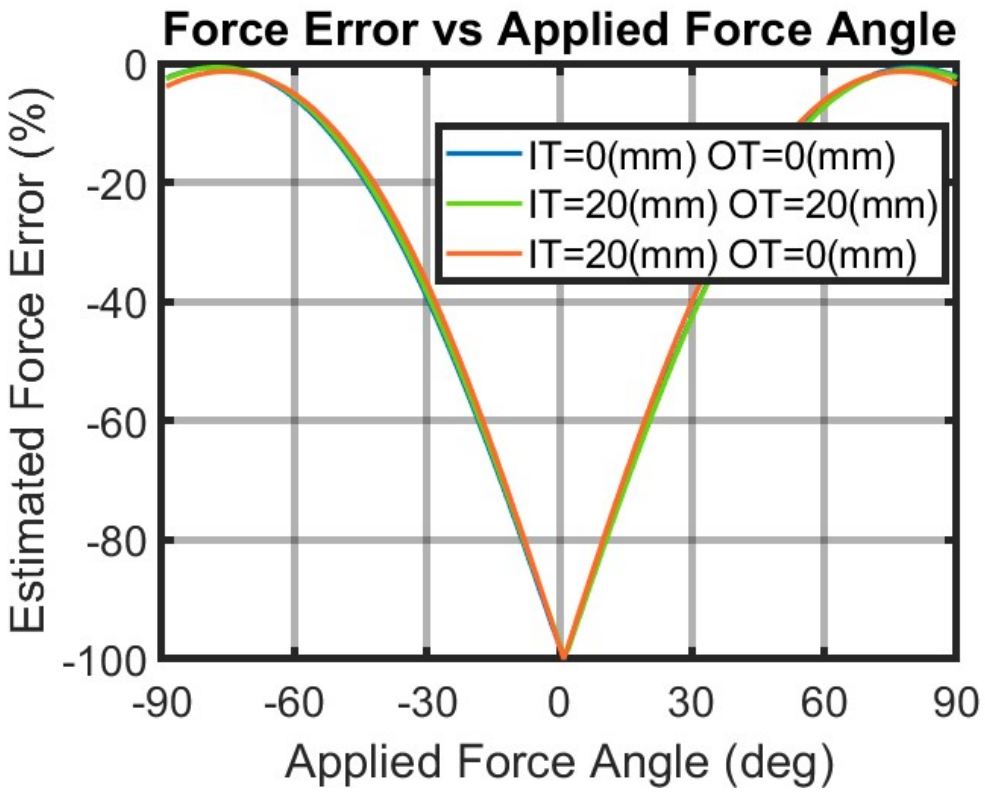


Figure 7.7: This figure shows the percentage error in simulation of the estimated forces when the estimator ignores the component of the force in the stiff direction of the robot. The x axis shows the angle the forces are applied to the robot tip, with 0° being when the force is applied directly against the tangent vector of the tip. This figure shows three different robot configurations. IT stands for the inner tube translation. OT stands for outer tube translation.

The results of the first simulation are shown in Figure 7.7. As observed, the algorithm exhibits relatively low error when the applied force is near perpendicular to the tool tip (approximately ±60° to 90°). However, as the force becomes more aligned with the tangent vector of the tool, the error increases, reaching up to −100%, indicating a complete underestimation of the force. This behavior is expected, as the method neglects force components in the stiff direction, which is closely aligned with the tip tangent.

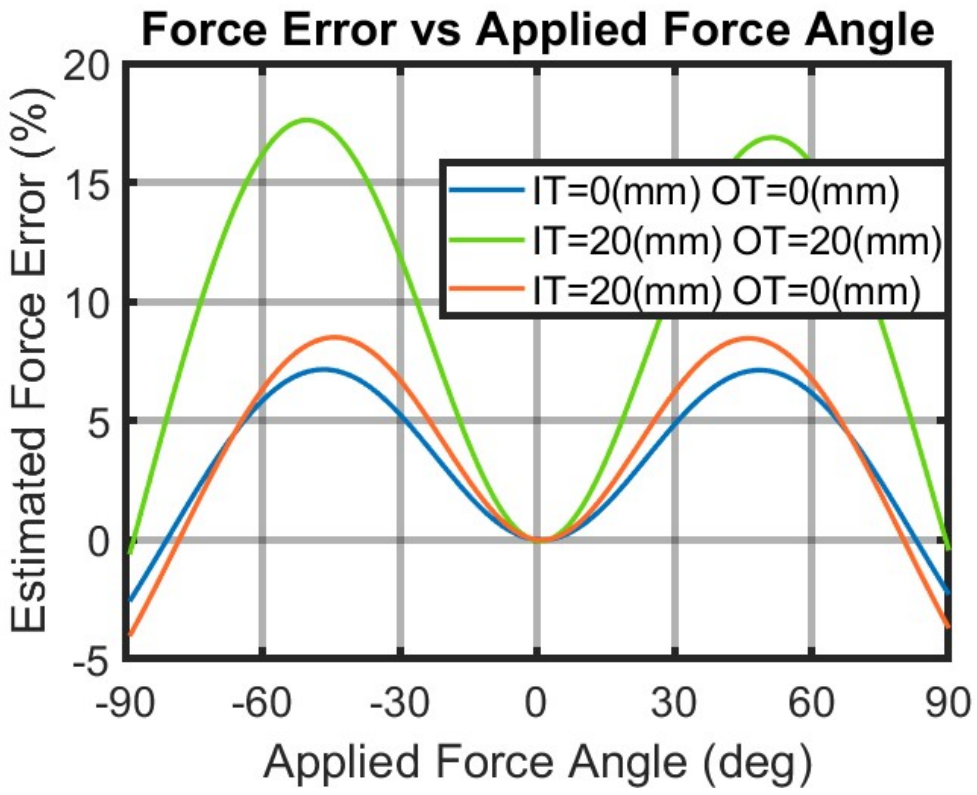


Figure 7.8: This figure shows the percent error in simulation of the estimated forces when the estimator includes information from a base force sensor. The x axis shows the angle the forces are applied to the robot tip, with 0° being when the force is applied directly against the tangent vector of the tip. This figure shows three different robot configurations. IT stands for inner tube translation. OT stands for outer tube translation.

The results of the second simulation are shown in Figure 7.8. In this case, the estimation error is significantly reduced compared to the first method. However, non-negligible errors remain. These discrepancies arise primarily from the linear approximation used in the force estimation algorithm, which is based on the undeformed configuration. In contrast, the ground truth model exhibits nonlinear deformation under applied tip forces, leading to residual error due to linearization.

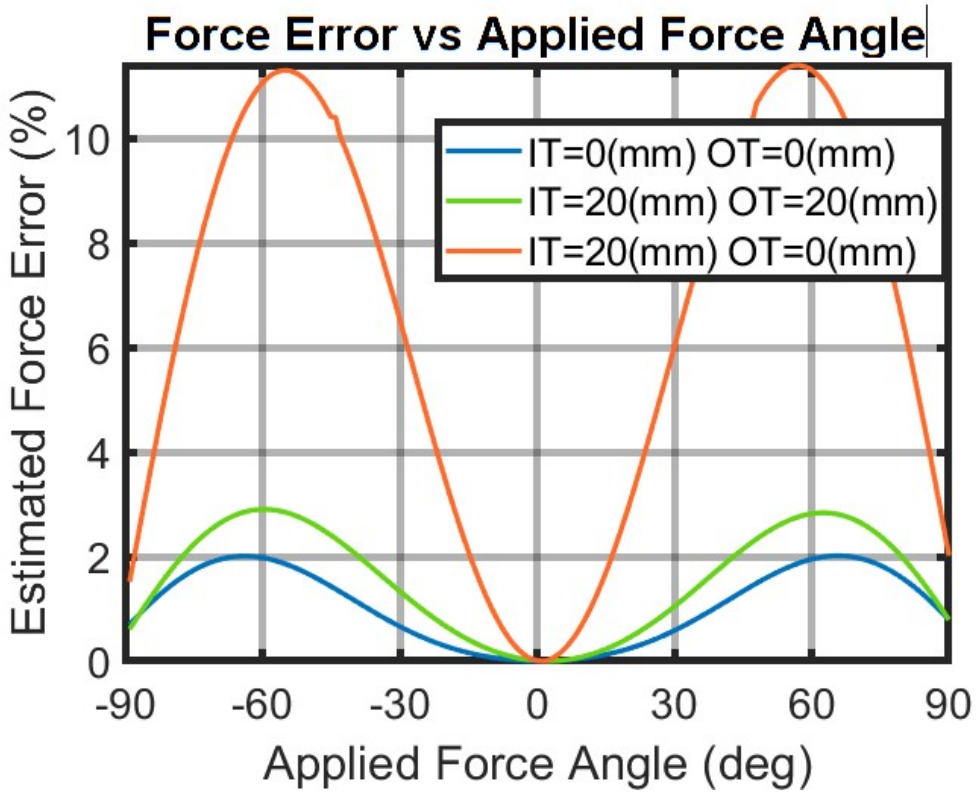


Figure 7.9: This figure shows the percentage error in simulation of the estimated forces when using our spring tip design. The x axis shows the angle the forces are applied to the robot tip, with 0° being when the force is applied directly against the tangent vector of the tip. This figure shows three different robot configurations. IT stands for inner tube translation. OT stands for outer tube translation.

The results of the third simulation, shown in Figure 7.9, demonstrate further improvement, with lower force estimation error than both previous methods. Interestingly, this result differs from the expectation that Methods 2 and 3 would perform nearly identically. The maximum error occurs in Configuration 3, where the inner tube is extended while the outer tube remains retracted. In this configuration, the system compliance is highest, increasing the influence of nonlinear effects. Notably, when the applied force is aligned with the tangent of the tool tip, the estimation error approaches 0% across all configurations. This is because the spring tip behaves approximately linearly under axial loading, allowing for highly accurate force estimation when the deflection is dominated by the spring.

7.2.2 Force Estimation Experiments

External Force Sensor Validation

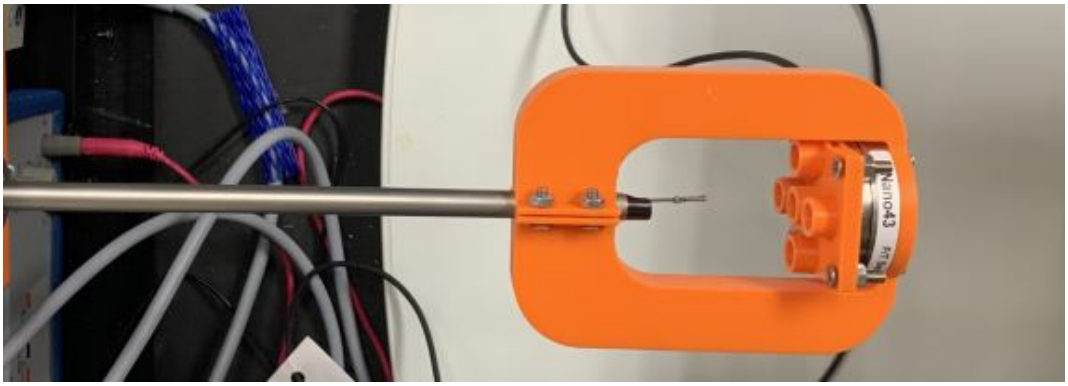


Figure 7.10: This figure shows our force sensor apparatus fixed to the exterior of the robot endoscope.

To evaluate the accuracy of the proposed force sensing approach, experiments were conducted in which the robot palpates against an external ground-truth force sensor (ATI Nano43, ATI Industrial Automation, Apex, NC, USA). A custom palpation apparatus was 3D printed to mount the Nano43 sensor at an adjustable distance from the endoscope tip, as shown in Figure 7.10. The front face of the sensor was fitted with a plate containing five cylindrical cavities, each with an inner diameter of 8 mm, a depth of 7 mm, and a center-to-center spacing of 15 mm. The cavities are arranged in a cross pattern, with one central cavity and four additional cavities aligned along the horizontal and vertical axes. Data was collected with the plate positioned such that its surface was 35 mm from the endoscope tip. For each cavity, the following protocol was executed. First, the robot tip was positioned approximately 2 mm inside the cavity, centered within the opening. From this position, the robot was commanded to extend and apply force to the cavity walls by translating 8 mm in one of four directions: up, down, left, and right. Corresponding to motions parallel to the back surface of the cavity. During each interaction, the forces measured by the external sensor, along with the robot's tip position, orientation, and corresponding model configuration, were recorded.

Because of the distance from the center of the plate to the outer edges of the cavities and the closeness of the palpation plate to the endoscope, the outer edges experience significant

loading along the stiff axis of the robot. Palpation in these configurations would not be possible using the palpation design described in the previous chapter.

After each force application, the robot returned to the center position within the cavity. To mitigate hysteresis effects, the robot was then fully retracted to its home position before returning to the same cavity and executing the next push direction. This process was repeated for all reachable directions within each cavity. Once all directions were tested, the procedure was repeated for the remaining cavities to ensure comprehensive spatial coverage. The center cavity was excluded in this experiment because of its proximity to one of the robot's singularity points. This procedure was repeated 4 times in total, yielding 64 distinct force measurements. As in the previous chapter, a subset of the data is used to calibrate the physical parameters of the robot. For this experiment, we will use data from four of the cavities for calibration and the remainder to verify.

This calibration is necessary because the effective stiffness of the palpation tool exhibits significant variability in practice, and the torsional compliance of the transmission elements within the endoscope is not known a priori. Accordingly, the axial, bending, and torsional stiffness parameters were identified using a nonlinear least-squares approach based on measured force magnitudes.

Let $\mathbf{F}_{\text{tip}}^{(m)}$ denote the ground-truth force magnitude measured by the reference sensor in experiment m, and let $\widetilde{\mathbf{F}}_{\text{tip}}^{(m)}(\boldsymbol{\theta})$ denote the model-predicted tip force for a given parameter vector $\boldsymbol{\theta} = [\alpha_a, \alpha_b, c]^\top$, where α_a and α_b scale the nominal axial and bending stiffness, respectively, and c is the torsional stiffness. The parameters were identified by minimizing the error in force magnitude:

$$\boldsymbol{\theta}^\star = \arg\min_{\boldsymbol{\theta}} \sum_{m=1}^{M} \left(\|\widetilde{\mathbf{F}}_{\text{tip}}^{(m)}(\boldsymbol{\theta})\| - \mathbf{F}_{\text{tip}}^{(m)} \right)^2. \tag{7.21}$$

The optimization was solved using MATLAB's `lsqnonlin` with finite-difference Jacobians.

The resulting parameters were $\alpha_a^\star = 0.70$, $\alpha_b^\star = 0.31$, and $c^\star = 0.15$ rad/(N·m). These values reflect additional compliance in the assembled instrument, including contributions

from the transmission system, bonding interfaces, and mechanical clearances, which are not explicitly captured by the nominal rod model.

After calibration, the force magnitudes predicted by the estimation algorithm were compared to those measured by the reference sensor. For the calibration dataset, the mean absolute error (MAE), expressed as a percentage of the measured force magnitude, was 17.1%. Across all 64 data points, the MAE was 18.5%.

These results are comparable to those obtained with the previous design; however, the present approach includes palpation orientations involving significant loading along stiff directions, which posed challenges for the previous design.

7.2.3 Stiffness Estimation Experiments

Experiments were conducted to evaluate the performance of the proposed stiffness estimation framework. The experimental setup is shown in Fig. 7.11. A phantom tissue block containing an embedded tumor was used to evaluate stiffness-based tumor localization. The palpation tool was equipped with an EM tracking sensor, which measured the deflected tip pose during probing.

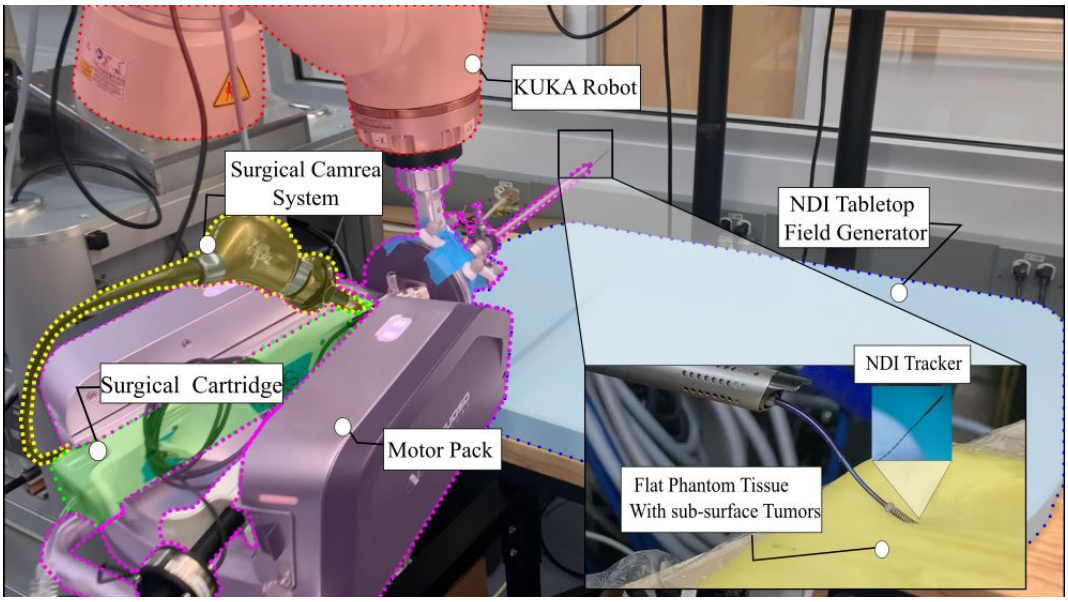


Figure 7.11: Experimental setup used for automated palpation of the phantom using the robotic systems and EM tracking sensor.

Tool Sensitivity Test

To test realistic surface stiffness discrimination, gelatin based tumors were placed 5 mm below the surface of the flat phantom tissue. These tumors have a stiffness 7 times greater than that of the phantom tissue. This is consistent with the difference in tissue stiffness that occurs between healthy prostate tissue and malignant prostate tumors, which ranges from 3 to 7 times stiffer [96, 11, 35]. The surface of the tissue above the tumor was then palpated using the Virtuoso system and our palpation tool. The stiffness readings were then compared to stiffness results from palpating the phantom tissue without any gelatin based tumors under the surface. The average tissue stiffness at the surface of the phantom tissue above the gelatin based tumor, as determined by our palpation tool, was 0.41 N/mm. In contrast the average stiffness seen at the surface of the phantom tissue without any subsurface lesions was 0.27 N/mm.

Stiffness Mapping Using KUKA

Following this, we preformed an experiment to demonstrate localization of a tumor through grid-based palpation and stiffness mapping. For these experiments we used 3D printed cylindrical tumors because their shape is well characterized and position sensors can be easily embedded in at their centers, which allows for more accurate evaluation. The simulated tumors are PLA cylinders with a height and diameter of 7 mm. These tumors are then placed 5 mm under the surface of the phantom tissue.

A grid of palpation points was defined on the phantom. At each grid location, the robot first moved the tool tip to the surface contact position and recorded the nominal tip pose. The robot then executed a probing motion along a predefined pushing direction with a fixed depth. During the probing motion, the deflected tip pose was recorded using the EM tracking sensor. The difference between the nominal pose and the deflected pose was used to estimate the local stiffness at each probing location and construct a stiffness map of the scanned region.

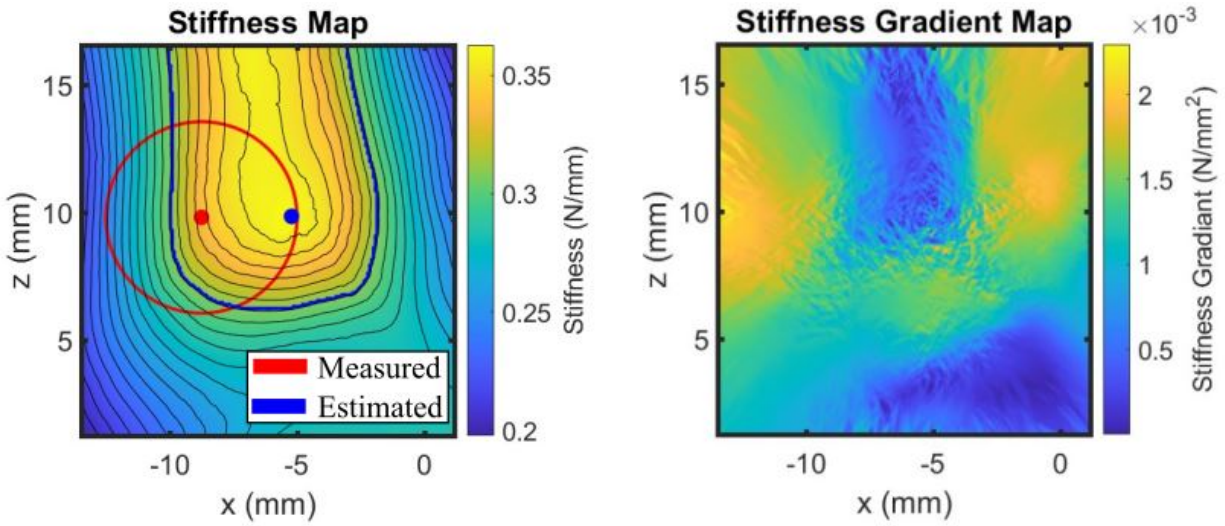


Figure 7.12: Stiffness map obtained using grid-based robotic palpation. Tumor localization error: 3.5 mm. The red dot indicates the measured tumor position. The red circle indicates the outer edge of the tumor based on the "ground-truth" measured tumor location. The blue Dot indicates the estimated tumor position. The blue line indicates the outer edge of the tumor based the palpation generated stiffness map.

The KUKA robot was used to move the tool tip to commanded Cartesian setpoints across the grid. Meanwhile, the Virtuoso system is kept at a constant configuration with the inner and outer tubes extended 20 mm. At each grid point, the robot first moves to the surface contact position and records the nominal tip pose using the EM tracking sensor. The robot then executes the probing motion by advancing the tool tip along the predefined pushing direction (normal to the tissue) with a commanded depth of 6 mm. Force and tissue stiffness are calculated by our algorithm using the sensor data. The grid consists of 240 points (12 rows of 20 points each). After all grid points are visited, we generate a smooth stiffness map of the region of interest using a LOWESS fit (local linear regression) with a span of 0.3.

The resulting stiffness map and a map of its gradient are shown in Figure 7.12. The estimated center point is indicated by the blue dot on the figure, and the estimated tumor boundary is then predicted using contour representing the stiffest 15%. The estimated Tumor takes a tear shape. This is because the palpation tool cannot effectively reach the distal tissue without contacting the tumor when pushing. This issue can be addressed by coordinating

Virtuoso and KUKA movements to palpate the tissue using more intelligent path planning algorithms involving preoperative imaging.

When compared to the recoded tumor position from the magnetic tracker in the tumor (indicated in red), the center to center localization error is 3.5 mm. The estimated tumor diameter is 7.9 mm (less than a millimeter larger than the actual size). Along with the expected sensor uncertainty of 1.6 mm, additional measurement uncertainty may be caused by the two magnetic sensor coils (the ground truth coil and the palpation sensor coil) being in close proximity, potentially creating interference. We suspect that this is also contributing to the 3.5 mm localization error.

7.3 Chapter Summary

This chapter has presented the design and validation of a magnetic tracking–based palpation tool for force estimation in concentric tube robots. By integrating a 5-DOF magnetic tracker with a quasi-static Cosserat rod model, we developed a compact approach to estimate contact forces without requiring dedicated force sensors. We modify one of our concentric tubes using laser patterning and an axial spring to be able to provide 3-axis tip sensing. We show this method has lower force estimation error in simulation than our previous method of removing the component of the force aligned with the stiff direction of the robot. We also show the accuracy of our method by palpating a force sensor and comparing the magnitude of the force sensor with that of the estimated force from our algorithm. Finally, we perform palpation experiments on phantom tissue with sub surface inclusions to mimic tumors. We locate the tumor center to within 3.5 mm, showing the validity of our approach in tumor localization.

Chapter 8

Conclusion

The motivation behind this work is to improve the capabilities of continuum robots, with a particular focus on enhancing state and force estimation to enable advanced control frameworks and robotic palpation for estimating force and stiffness properties of biological tissue.

Throughout this dissertation, we have addressed these challenges by developing and validating methods for state and force estimation in continuum robotic systems. In chapters 3 and 4, we introduced a modeling framework and a force-based observer suitable for control applications. This work advances continuum robot modeling through a constrained Lagrangian formulation that is computationally efficient for control implementation. Chapter 3 presented a planar model-based observer and task-space controller, validated in simulation, while Chapter 4 extended these results to the fully three-dimensional case with an emphasis on state estimation. These approaches were validated through both simulation and experimental studies on a large flexible robotic system.

Building on these principles, this work then explored applications in surgical robot palpation using concentric tube robots. While force sensing with concentric tube robots has been previously demonstrated, prior to this work, sensing at such small scales using commercial surgical systems had not been achieved. Additionally, force estimation along the stiff axial direction of the robot remained an open challenge. In Chapters 6 and 7, we presented hardware designs and force estimation algorithms for identifying stiffness variations in tissue using commercial systems, while also addressing the challenge of sensing

in the stiff direction. Chapter 6 introduced a novel force-sensing approach using a surgical robot, and Chapter 7 improved upon this design with a more robust tool that addresses key limitations of the initial system. These methods were validated through both simulation and experimental evaluation, demonstrating, for the first time, tumor localization using stiffness maps generated from concentric tube robotic palpation.

While this work establishes a foundation for estimation and palpation in continuum robots, several opportunities remain for future research. The 3D modeling framework developed in this dissertation can be further leveraged to implement and evaluate more advanced control strategies. Additionally, the force estimation methods presented here can be improved by incorporating nonlinear mappings, as the current linear compliance formulation introduces estimation error.

Further improvements to palpation performance may be achieved through more robust tumor localization techniques applied to the stiffness maps (obtained from palpation), such as data-driven or learning-based methods. Hardware enhancements also present a promising direction. For example, incorporating an additional NDI sensor at the proximal end of the compliant section described in Chapter 7 could reduce reliance on kinematic assumptions and improve estimation accuracy. Finally, replacing dedicated position sensors with vision-based tracking from a surgical camera could simplify the system architecture while enabling greater flexibility for integrating additional surgical tools.

In summary, this dissertation advances the state of the art in continuum robot modeling, estimation, and palpation by introducing frameworks for force and state estimation and demonstrating their application to real-world robotic systems. While challenges remain, the methods and insights presented here provide a foundation for future developments in continuum robotics, particularly in medical applications where accurate sensing and interaction are critical. Ultimately, it is hoped that the contributions presented in this dissertation, along with continued work by the research community, will lead to meaningful advancements in technologies that improve human health and safety.

Bibliography

[1] Al-Ahmad, O., Ourak, M., Vlekken, J., and Vander Poorten, E. (2021). Fbg-based estimation of external forces along flexible instrument bodies. *Frontiers in Robotics and AI*, 8:718033. 54

[2] Antman, S. S. (2005). *Nonlinear Problems of Elasticity*. Springer Science, 2nd edition. 10, 27

[3] Armanini, C., Boyer, F., Mathew, A. T., Duriez, C., and Renda, F. (2023). Soft robots modeling: A structured overview. *IEEE Transactions on Robotics*, 39(3):1728–1748. 7

[4] Ataka, A., Abrar, T., Putzu, F., Godaba, H., and Althoefer, K. (2020). Observer-based control of inflatable robot with variable stiffness. In *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 8646–8652. 7

[5] Azizkhani, M., Gunderman, A. L., Godage, I. S., and Chen, Y. (2023). Dynamic control of soft robotic arm: An experimental study. *IEEE Robotics and Automation Letters*, 8(4):1897–1904. 42

[6] Baumgarte, J. (1972). Stabilization of constraints and integrals of motion in dynamical systems. *Computer methods in applied mechanics and engineering*, 1(1):1–16. 35

[7] Bergou, M., Audoly, B., Vouga, E., Wardetzky, M., and Grinspun, E. (2010). Discrete viscous threads. *ACM Trans. Graph.*, 29(4). 6

[8] Bergou, M., Wardetzky, M., Robinson, S., Audoly, B., and Grinspun, E. (2008). Discrete elastic rods. In *ACM SIGGRAPH 2008 Papers*, SIGGRAPH '08, New York, NY, USA. Association for Computing Machinery. 6

[9] Boyer, F., Lebastard, V., Candelier, F., and Renda, F. (2021). Dynamics of continuum and soft robots: A strain parameterization based approach. *IEEE Transactions on Robotics*, 37(3):847–863. 6

[10] Burgner-Kahrs, J., Rucker, D. C., and Choset, H. (2015). Continuum Robots for Medical Applications: A Survey. *IEEE Transactions on Robotics*, 31(6):1261–1280. 2

[11] Cao, R., Huang, Z., Varghese, T., and Nabi, G. (2013). Tissue mimicking materials for the detection of prostate cancer using shear wave elastography: A validation study. *Medical physics*, 40(2):022903. 103

[12] Chirikjian, G. S. (1994). Hyper-redundant manipulator dynamics: a continuum approximation. *Advanced Robotics*, 9(3):217–243. 5

[13] Cianchetti, M., Laschi, C., Menciassi, A., and Dario, P. (2018). Biomedical applications of soft robotics. *Nature Reviews Materials*, 3:143–153. 1

[14] Dagnino, G. and Kundrat, D. (2024). Robot-assistive minimally invasive surgery: trends and future directions. *International Journal of Intelligent Robotics and Applications*, 8(4):812–826. 53

[15] Dang, K. T., Qiu, S., Hatch, C., Connor, P., Qin, T., Alterovitz, R., Webster III, R. J., and Rucker, C. (2024). Design of transmission tubes for surgical concentric push-pull robots. In *2024 International Symposium on Medical Robotics (ISMR)*, pages 1–7. IEEE. 87

[16] Della Santina, C., Duriez, C., and Rus, D. (2023). 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. 7

[17] Dupont, P. E., Simaan, N., Choset, H., and Rucker, C. (2022). Continuum robots for medical interventions. *Proceedings of the IEEE*, 110(7):847–870. 2

[18] Ehsani-Seresht, A. and Hashemi-Pour Moosavi, S. (2020). Dynamic modeling of the cable-driven continuum robots in hybrid position-force actuation mode. *Journal of Mechanisms and Robotics*, 12(5):051002. 5, 37

[19] Falkenhahn, V., Hildebrandt, A., Neumann, R., and Sawodny, O. (2017). Dynamic control of the bionic handling assistant. *IEEE/ASME Transactions on Mechatronics*, 22(1):6–17. 7

[20] Falkenhahn, V., Mahl, T., Hildebrandt, A., Neumann, R., and Sawodny, O. (2015). Dynamic modeling of bellows-actuated continuum robots using the euler–lagrange formalism. *IEEE Transactions on Robotics*, 31(6):1483–1496. 6

[21] Florian, G., Gabor, A. R., Nicolae, C.-A., Rotaru, A., Stănică, N., Bîzdoacă, N., and Rotaru, P. (2020). Thermomechanical, calorimetric and magnetic properties of a ni–ti shape-memory alloy wire: G. florian et al. *Journal of Thermal Analysis and Calorimetry*, 140(2):527–544. 61

[22] Franz, A. M., Haidegger, T., Birkfellner, W., Cleary, K., Peters, T. M., and Maier-Hein, L. (2014). Electromagnetic tracking in medicine—a review of technology, validation, and applications. *IEEE transactions on medical imaging*, 33(8):1702–1725. 54

[23] Gallentine, J. and Barth, E. J. (2023). 3d printed gerotor pump geometries for soft-robot actuation. *ASME/BATH 2023 Symposium on Fluid Power and Motion Control*, page V001T01A019. 1

[24] Gaston, J. B., Kumar, N. S., Barth, E. J., and Rucker, C. (2026). A 3d discrete elastic rod model and observer for continuum robots. *Journal of Mechanisms and Robotics*, 18(3):031002. 2, 27

[25] George Thuruthel, T., Ansari, Y., Falotico, E., and Laschi, C. (2018). Control strategies for soft robotic manipulators: A survey. *Soft Robotics*, 5(2):149–163. PMID: 29297756. 7

[26] Ghoul, A., Djeffal, S., Wang, H., Kara, K., and Hadjili, M. L. (2024). Enhancing surgical robotics: A dynamic model and optimized control strategy for cable-driven continuum robots. *Journal of Mechanisms and Robotics*, 17(2):021002. 5

[27] Gilbert, H. B. (2021). On the mathematical modeling of slender biomedical continuum robots. *Frontiers in Robotics and AI*, 8:2296–9144. 7

[28] Godage, I. S., Medrano-Cerda, G. A., Branson, D. T., Guglielmino, E., and Caldwell, D. G. (2016). Dynamics for variable length multisection continuum arms. *The International Journal of Robotics Research*, 35(6):695–722. 6

[29] Gravagne, I. A., Rahn, C. D., and Walker, I. D. (2003). Large-Deflection Dynamics and Control for Planar Continuum Robots. *IEEE/ASME Transactions on Mechatronics*, 8(2):299–307. 8

[30] Guillard, H. and Bourles, H. (2000). Robust feedback linearization. In *Proc. 14 th International Symposium on Mathematical Theory of Networks and Systems*. 21

[31] He, X., Handa, J., Gehlbach, P., Taylor, R., and Iordachita, I. (2013). A submillimetric 3-dof force sensing instrument with integrated fiber bragg grating for retinal microsurgery. *IEEE Transactions on Biomedical Engineering*, 61(2):522–534. 55

[32] Hu, A. and Sun, Y. (2025). Contact force estimation of continuum robots without embedded sensors: A review. *Advanced Intelligent Systems*, page e202500786. 54

[33] Hu, Z., Li, J., and Wang, S. (2023). Design and kinematics of a robotic instrument for natural orifice transluminal endoscopic surgery. *IEEE/ASME Transactions on Mechatronics*, 28(5):2840–2851. 54

[34] Jawed, M. K., Novelia, A., and O'Reilly, O. M. (2018). *A primer on the kinematics of discrete elastic rods*. Springer. 6

[35] Ji, Y., Ruan, L., Ren, W., Dun, G., Liu, J., Zhang, Y., and Wan, Q. (2009). Stiffness of prostate gland measured by transrectal real-time shear wave elastography for detection of

prostate cancer: a feasibility study. *The British journal of radiology*, 92(1097):20180970. 103

[36] Kareem, A. (2025). Block tridiagonal solver, matlab central file exchange, https://www.mathworks.com/matlabcentral/fileexchange/76078-block-tridiagonal-solver. 45

[37] Katzschmann, R. K., Santina, C. D., Toshimitsu, Y., Bicchi, A., and Rus, D. (2019). 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. 5, 8

[38] Krauss, H. and Takemura, K. (2024). Enhanced model-free dynamic state estimation for a soft robot finger using an embedded optical waveguide sensor. *IEEE Robotics and Automation Letters*, 9(7):6123–6129. 7

[39] Laschi, C. (2016). Soft Robotics: from scientific challenges to technological applications. In George, T., Dutta, A. K., and Islam, M. S., editors, *Micro- and Nanotechnology Sensors, Systems, and Applications VIII*, volume 9836, page 983626. International Society for Optics and Photonics, SPIE. 1

[40] Laschi, C., Mazzolai, B., and Cianchetti, M. (2016). Soft robotics: Technologies and systems pushing the boundaries of robot abilities. *Science robotics*, 1,1. 1

[41] Li, M., Kang, R., Branson, D. T., and Dai, J. S. (2018). Model-free control for continuum robots based on an adaptive kalman filter. *IEEE/ASME Transactions on Mechatronics*, 23(1):286–297. 7

[42] Li, S., Kruszewski, A., Guerra, T.-M., and Nguyen, A.-T. (2022). Equivalent-input-disturbance-based dynamic tracking control for soft robots via reduced-order finite-element models. *IEEE/ASME Transactions on Mechatronics*, pages 1–12. 8

[43] Liang, X., He, G., Su, T., Wang, W., Huang, C., Zhao, Q., and Hou, Z.-G. (2022). Finite-time observer-based variable impedance control of cable-driven continuum manipulators. *IEEE Transactions on Human-Machine Systems*, 52(1):26–40. 7

[44] Linn, J. (2020). Discrete cosserat rod kinematics constructed on the basis of the difference geometry of framed curves—part i: discrete cosserat curves on a staggered grid. *Journal of Elasticity*, 139(2):177–236. 6

[45] Linn, J., Hermansson, T., Andersson, F., and Schneider, F. (2017). Kinetic aspects of discrete cosserat rods based on the difference geometry of framed curves. In *ECCOMAS Thematic Conference on Multibody Dynamics, Prague, Czech Republic*, pages 163–176. 6

[46] Loo, J. Y., Tan, C. P., and Nurzaman, S. G. (2019). H-infinity based extended kalman filter for state estimation in highly non-linear soft robotic system. In *2019 American Control Conference (ACC)*, pages 5154–5160. 7

[47] Luo, X., Wan, Y., and He, X. (2015). Robust electromagnetically guided endoscopic procedure using enhanced particle swarm optimization for multimodal information fusion. *Medical physics*, 42(4):1808–1817. 54

[48] Macklin, M. and Muller, M. (2021). A constraint-based formulation of stable neo-hookean materials. In *Proceedings of the 14th ACM SIGGRAPH Conference on Motion, Interaction and Games*, MIG '21, New York, NY, USA. Association for Computing Machinery. 76

[49] Mahvash, M. and Dupont, P. E. (2011). Stiffness control of surgical continuum manipulators. *IEEE Transactions on Robotics*, 27(2):334–345. 54, 55

[50] Mbakop, S., Tagne, G., Frouin, M.-H., Melingui, A., and Merzouki, R. (2021). Inverse dynamics model-based shape control of soft continuum finger robot using parametric curve. *IEEE Robotics and Automation Letters*, 6(4):8053–8060. 6, 8

[51] Melnyk, R., Ezzat, B., Belfast, E., Saba, P., Farooq, S., Campbell, T., McAleavey, S., Buckley, M., and Ghazi, A. (2020). Mechanical and functional validation of a perfused, robot-assisted partial nephrectomy simulation platform using a combination of 3d printing and hydrogel casting. *World journal of urology*, 38(7):1631–1641. 60, 86

[52] Murray, R. M., Li, Z., and Sastry, S. S. (1994). *A Mathematical Introduction to Robotic Manipulation*. CRC Press, Boca Raton, FL. 29

[53] Müller, D., Veil, C., and Sawodny, O. (2020). Disturbance observer based control for quasi continuum manipulators. *IFAC-PapersOnLine*, 53(2):9808–9813. 21st IFAC World Congress. 7

[54] Nakai, A., Kuwana, K., Saito, K., Dohi, T., Kumagai, A., and Shimoyama, I. (2017). Mems 6-axis force-torque sensor attached to the tip of grasping forceps for identification of tumor in thoracoscopic surgery. In *2017 IEEE 30th International Conference on Micro Electro Mechanical Systems (MEMS)*, pages 546–548. IEEE. 54

[55] Nakamoto, M., Nakada, K., Sato, Y., Konishi, K., Hashizume, M., and Tamura, S. (2008). Intraoperative magnetic tracker calibration using a magneto-optic hybrid tracker for 3-d ultrasound-based navigation in laparoscopic surgery. *IEEE transactions on medical imaging*, 27(2):255–270. 54

[56] Qiao, Q., Borghesan, G., De Schutter, J., and Vander Poorten, E. (2021). Force from shape—estimating the location and magnitude of the external force on flexible instruments. *IEEE Transactions on Robotics*, 37(5):1826–1833. 54

[57] Qin, L., Peng, H., Huang, X., Liu, M., and Huang, W. (2024). Modeling and simulation of dynamics in soft robotics: A review of numerical approaches. *Current Robotics Reports*, 5(1):1–13. 6

[58] Quarteroni, A., Sacco, R., and Saleri, F. (2010). *Numerical mathematics*, volume 37. Springer Science & Business Media. 37, 47

[59] Renda, F., Armanini, C., Lebastard, V., Candelier, F., and Boyer, F. (2020). A geometric variable-strain approach for static modeling of soft manipulators with tendon and fluidic actuation. *IEEE Robotics and Automation Letters*, 5(3):4006–4013. 6

[60] Renda, F., Cianchetti, M., Abidi, H., Dias, J., and Seneviratne, L. (2017). Screw-based modeling of soft manipulators with tendon and fluidic actuation. *Journal of Mechanisms and Robotics*, 9(4):041012. 6

[61] Rone, W. S. and Ben-Tzvi, P. (2014). Mechanics modeling of multisegment rod-driven continuum robots. *Journal of Mechanisms and Robotics*, 6(4):041006. 5

[62] Rucker, C. (2018). Integrating rotations using nonunit quaternions. *IEEE Robotics and Automation Letters*, 3(4):2979–2986. 41, 42

[63] Rucker, C., Barth, E. J., Gaston, J., and Gallentine, J. C. (2022). Task-space control of continuum robots using underactuated discrete rod models. In *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, pages 10967–10974. 2, 10, 37

[64] Rucker, D. C., Jones, B. A., and Webster III, R. J. (2010). A geometrically exact model for externally loaded concentric-tube continuum robots. *IEEE transactions on robotics*, 26(5):769–780. 62, 64, 88

[65] Rucker, D. C. and Webster, R. J. (2011a). Computing jacobians and compliance matrices for externally loaded continuum robots. In *2011 IEEE international conference on robotics and automation*, pages 945–950. IEEE. 66

[66] Rucker, D. C. and Webster, R. J. (2011b). Deflection-based force sensing for continuum robots: A probabilistic approach. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 3764–3769. IEEE. 54, 55

[67] Rucker, D. C. and Webster III, R. J. (2011). Deflection-Based Force Sensing for Continuum Robots: A Probabilistic Approach. *IEEE/RSJ International Conference on Intellegent Robots and Systems*, pages 3764–3769. 7

[68] Rucker, D. C. and Webster III, R. J. (2011). Statics and dynamics of continuum robots with general tendon routing and external loading. *IEEE Transactions on Robotics*, 27(6):1033–1044. 17, 37

[69] Rus, D. and Tolley, M. T. (2015). Design, fabrication and control of soft robots. *Nature*, 521:467–475. 1

[70] Santina, C. D., Pallottino, L., Rus, D., and Bicchi, A. (2019). Exact task execution in highly under-actuated soft limbs: An operational space based approach. *IEEE Robotics and Automation Letters*, 4(3):2508–2515. 8

[71] Santina, C. D., Truby, R. L., and Rus, D. (2020). Data–driven disturbance observers for estimating external forces on soft robots. *IEEE Robotics and Automation Letters*, 5(4):5717–5724. 7, 17

[72] Seibold, U., Kubler, B., and Hirzinger, G. (2005). Prototype of instrument for minimally invasive surgery with 6-axis force sensing capability. In *Proceedings of the 2005 IEEE International Conference on Robotics and Automation*, pages 496–501. IEEE. 54

[73] Shao, X., Pustina, P., Stölzle, M., Sun, G., De Luca, A., Wu, L., and Santina, C. D. (2024). Model-based control for soft robots with system uncertainties and input saturation. *IEEE Transactions on Industrial Electronics*, 71(7):7435–7444. 7

[74] Shi, C., Luo, X., Qi, P., Li, T., Song, S., Najdovski, Z., Fukuda, T., and Ren, H. (2016). Shape sensing techniques for continuum robots in minimally invasive surgery: A survey. *IEEE Transactions on Biomedical Engineering*, 64(8):1665–1678. 54

[75] Shi, C., Luo, X., Qi, P., Li, T., Song, S., Najdovski, Z., Fukuda, T., and Ren, H. (2017). Shape sensing techniques for continuum robots in minimally invasive surgery: A survey. *IEEE Transactions on Biomedical Engineering*, 64(8):1665–1678. 7

[76] Song, S., Li, Z., Meng, M. Q.-H., Yu, H., and Ren, H. (2015). Real-time shape estimation for wire-driven flexible robots with multiple bending sections based on quadratic bézier curves. *IEEE Sensors Journal*, 15(11):6326–6334. 54

[77] Spong, M. (1994). 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 vol.1. 20

[78] Stewart, K., Qiao, Z., and Zhang, W. (2022). State estimation and control with a robust extended kalman filter for a fabric soft robot. *IFAC-PapersOnLine*, 55(37):25–30. 2nd Modeling, Estimation and Control Conference MECC 2022. 7

[79] Talbot, W., Nubert, J., Tuna, T., Cadena, C., Dümbgen, F., Tordesillas, J., Barfoot, T. D., and Hutter, M. (2024). Continuous-time state estimation methods in robotics: A survey. 7

[80] Thieffry, M., Kruszewski, A., Duriez, C., and Guerra, T.-M. (2019). Control design for soft robots based on reduced-order model. *IEEE Robotics and Automation Letters*, 4(1):25–32. 8

[81] Till, J., Aloi, V., and Rucker, C. (2019). Real-time dynamics of soft and continuum robots based on cosserat rod models. *The International Journal of Robotics Research*, 38(6):723–746. 6, 17

[82] Tobin, S., Wu, Y., Liu, F., and Rucker, C. (2025). First-order position based simulation of deformable concentric-tube robots interacting with tissue. *Journal of Medical Robotics Research*. 76

[83] Trivedi, D. and Rahn, C. D. (2014). Model-based shape estimation for soft robotic manipulators: The planar case. *Journal of Mechanisms and Robotics*, 6(2):021005. 6

[84] Trivedi, D., Rahn Christopher, D., Kier William, M., and Walker Ian, D. (2008). Soft robotics: Biological inspiration, state of the art, and future research. *Applied Bionics and Biomechanics*, 5(3):99–117. 1

[85] Tully, S., Kantor, G., Zenati, M. A., and Choset, H. (2011). Shape estimation for image-guided surgery with a highly articulated snake robot. In *2011 IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1353–1358. IEEE. 54

[86] Walid, A., Zheng, G., Kruszewski, A., and Renda, F. (2021). Discrete cosserat method for soft manipulators workspace estimation: An optimization-based approach. *Journal of Mechanisms and Robotics*, 14(1):011012. 6

[87] Walker, I. D. (2013). Continuous backbone "continuum" robot manipulators. *International Scholarly Research Notices*, 2013(1):726506. 2

[88] Wang, C., Frazelle, C. G., Wagner, J. R., and Walker, I. D. (2021a). Dynamic control of multisection three-dimensional continuum manipulators based on virtual discrete-jointed robot models. *IEEE/ASME Transactions on Mechatronics*, 26(2):777–788. 6, 7

[89] Wang, H., Wang, C., Chen, W., Liang, X., and Liu, Y. (2017). Three-dimensional dynamics for cable-driven soft manipulator. *IEEE/ASME Transactions on Mechatronics*, 22(1):18–28. 5

[90] Wang, H., Yan, Z., Gao, Y., Wang, W., and Du, Z. (2021b). 3-d force sensing strategy of laryngeal continuum surgical robot based on fiber bragg gratings. *IEEE Transactions on Instrumentation and Measurement*, 70:1–10. 54

[91] Wang, Y., Wang, Y., Mushtaq, R. T., and Wei, Q. (2024). Advancements in soft robotics: A comprehensive review on actuation methods, materials, and applications. *Polymers*, 16(8). 1

[92] Webster III, R. J. and Jones, B. A. (2010). Design and Kinematic Modeling of Constant Curvature Continuum Robots: A Review. *International Journal of Robotics Research*. 2

[93] Webster III, R. J. and Jones, B. A. (2010). Design and kinematic modeling of constant curvature continuum robots: A review. *The International Journal of Robotics Research*, 29(13):1661–1683. 5

[94] Xie, Q., Wang, T., and Zhu, S. (2022). Simplified dynamical model and experimental verification of an underwater hydraulic soft robotic arm. *Smart Materials and Structures*, 31(7):075011. 5

[95] Xu, R., Yurkewich, A., and Patel, R. V. (2016). Curvature, torsion, and force sensing in continuum robots using helically wrapped fbg sensors. *IEEE Robotics and Automation Letters*, 1(2):1052–1059. 54

[96] Zhang, M., Nigwekar, P., Castaneda, B., Hoyt, K., Joseph, J. V., di Sant'Agnese, A., Messing, E. M., Strang, J. G., Rubens, D. J., and Parker, K. J. (2008). Quantitative characterization of viscoelastic properties of human prostate correlated with histology. *Ultrasound in medicine & biology*, 34(7):1033–1042. 103

[97] Zhang, T., Ping, Z., and Zuo, S. (2021). Miniature continuum manipulator with three degrees-of-freedom force sensing for retinal microsurgery. *Journal of Mechanisms and Robotics*, 13(4):041002. 54

[98] Zheng, T., Han, Q., and Lin, H. (2024a). Full state estimation of continuum robots from tip velocities: A cosserat-theoretic boundary observer. *IEEE Transactions on Automatic Control*, pages 1–13. 7

[99] Zheng, T., McFarland, C., Coad, M., and Lin, H. (2024b). Estimating infinite-dimensional continuum robot states from the tip. 7

Vita

Joshua Gaston is a doctoral candidate in Mechanical Engineering at the University of Tennessee, Knoxville. His research focuses on modeling, sensing, and control of continuum and soft robotic systems, with particular emphasis on force and state estimation. He received his Bachelor of Science in Mechanical Engineering from Tennessee Tech University, where he developed an interest in robotics. His work aims to advance the development of continuum robots, especially for application in minimally invasive surgical technologies. Following the completion of his doctorate, he intends to pursue a career studying and advancing the capabilities of robots.