To the Graduate Council:

I am submitting herewith a dissertation written by Vincent Aloi entitled "Model Based Force Estimation and Stiffness Control for Continuum Robots." I have examined the final paper copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Mechanical Engineering.

Dr. Daniel Rucker, Major Professor

We have read this dissertation
and recommend its acceptance:

Dr. Eric Barth

Dr. William Hamel

Dr. Jindong Tan

Accepted for the Council:

Dixie L. Thompson

Vice Provost and Dean of the Graduate School

To the Graduate Council:

I am submitting herewith a dissertation written by Vincent Aloi entitled "Model Based Force Estimation and Stiffness Control for Continuum Robots." I have examined the final electronic copy of this dissertation for form and content and recommend that it be accepted in partial fulfillment of the requirements for the degree of Doctor of Philosophy, with a major in Mechanical Engineering.

Dr. Daniel Rucker, Major Professor

We have read this dissertation
and recommend its acceptance:

Dr. Eric Barth

Dr. William Hamel

Dr. Jindong Tan

Accepted for the Council:

Dixie L. Thompson

Vice Provost and Dean of the Graduate School

(Original signatures are on file with official student records.)

Model Based Force Estimation and Stiffness Control for Continuum Robots

A Dissertation Presented for the

Doctor of Philosophy

Degree

The University of Tennessee, Knoxville

Vincent Aloi

May 2022

For Wally and Wendy

Acknowledgments

Graduate school has been a journey. One that I would not have been able to complete without the never ending and overwhelming support of those around me. I would like to thank my advisor Dr. Caleb Rucker for giving me the opportunity to work in his lab and for always having confidence in my abilities. I will also forever be grateful for the freedom Caleb gave me to focus on my own personal health during my many bouts of Diverticulitis.

I would like to thank all of my labmates who were a constant source of inspiration, laughs, and belonging: Kaitlin Oliver Butler, John Till, Jake Childs, Caroline Black, and Ryan Ponten. Thank you to the undergraduate researchers who helped tremendously with experiments: Drew Dove and Khoa Dang.

I would like to thank my friends: Brett Brownlee, Zachariah Nelson, Kelly Roth, and Tori Griffin, for listening to me complain about school for the past 5 years.

Lastly, I need to thank my family: Nicole, Mom, and Dad. You all are the whole reason I was able to make it this far. I love you all and thank you!

Parts of this work are based upon work supported by the National Science Foundation under CMMI-1935278 as part of the NSF/NASA/NIH/USDA/DOD National Robotics Initiative and under NSF CAREER Award IIS-1652588. Any opinion, findings, and conclusions or recommendations expressed herein are those of the author and do not necessarily reflect the views of the National Science Foundation.

Oh and we can't forget Wendy and Wally. I do not understand how anyone can grade 140 papers at 1:00 AM without a cat sat in their lap.

" ... as far as the propositions of mathematics refer to reality, they are not certain; and as far as they are certain, they do not refer to reality." - A. E.

Abstract

Continuum Robots are bio-inspired structures that mimic the motion of snakes, elephant trunks, octopus tentacles, etc. With good design, these robots can be naturally compliant and miniaturizable, which makes Continuum Robots ideal for traversing narrow complex environments. Their flexible design, however, prevents us from using traditional methods for controlling and estimating loading on rigid link robots.

In the first thrust of this research, we provided a novel stiffness control law that alters the behavior of an end effector during contact. This controller is applicable to any continuum robot where a method for sensing or estimating tip forces and pose exists. Using an integral approach, the control law is be capable of dictating different stiffness in multiple directions, both increase and decrease the stiffness of the end effector, as well as handle contacts with both rigid and compliant environments. An example of implementation is provided for a parallel continuum robot.

In the second thrust of this research, we introduce a general 3D load estimation approach that can be applied to any continuum robot with an existing kinetostatic model that maps actuation and externally applied loads into a robot pose. This method uses numerical minimization to predict a load distribution that will fit a robot model predicted shape to a directly sensed shape. Validation was preformed on a passive steel rod, a single degree of freedom tendon robot, and a two degree of freedom tendon robot.

Table of Contents

List of Tables

List of Figures

Chapter 1

Background

Continuum Robots are subset of robotics where the structures are designed to be flexible and deformable. They are ideal for any situation where compliant contact with the environment is necessary or a narrow complex pathway must be traversed. The applications range from use as soft gripers, manufacturing tools, terrain exploration devices, or surgical tools. The varying applications leads to a plethora of designs that make use of different forms of actuation and deformation modes. Some examples of these are parallel continuum robots, pneumatic/hydraulic robots, tendon actuated robots, concentric tube robots, and notched tube robots. Instead of having joints that can rotate, the manipulator itself is made of a flexible material that can bend and twist through elastic deformation [1]. This can be considered as the robot having infinite degrees of freedom, as one can imagine the robot being made of an infinite amount of infinitesimal links [2]. The hyper redundancy increases difficulty in evaluating kinematics but it also allows these manipulators to traverse complex and constricted environments, and requires a minimal amount of actuation [3].

1.1 Kinetostatic Modeling

In conventional robotics, manipulators are constructed with rigid parts. This gives an ease of control and implementation as kinematics can be directly derived using the geometry of the structure and using the classic Denavit–Hartenberg convention. In continuum robotics, however, it is no longer possible to formulate kinematics based solely on geometry, as the manipulator itself will deform due to actuation and external loading. This poses an interesting form of control as we can now leverage this deformation to achieve tasks such as manipulating an end effector. A new type of model must now be used to predict the shape of these robots, derived from the principles of continuum mechanics. We refer to these as kinetostatic models. Here we will quickly discuss three common models.

The first is the pseudo-rigid body model [4, 5, 6]. This simplification allows for a continuum robot to be approximated as a series of rigid links. The forces are accounted for by including torsional springs at each joint. The benefit of this model is the ease of implementation, as one can use the classic Denavit-Hartenberg convention and a series of rigid body transformations for the forward kinetostatics. This results in an implicit assumption, though, that all applied loads are point forces at the joints. It has also been found to be less accurate than continuum mechanics based models.

The second model is a constant curvature model [7, 8, 9, 10, 11, 12, 13]. This discretizes the robot into a series of linear Euler or Timoshenko beams. It is assumed that each beam deforms such that it is constant curvature. Using strain energy minimization, these models can be quickly evaluated. With defined curvature, one can simply map from the base frame to the tip frame using rigid body transformations. This modeling frame work forces the assumption that all loading is applied at the transition points from one beam to the next. These loads are resolved as point moments on the joints because the constant curvature discretization does not allow for shear forces along a segment.

The final modeling approach is to use the Cosserat or Kirchoff Rod Model [4, 10, 14, 15, 16, 17]. This is what we will be focusing on in this research. This is a geometrically exact model that is derived from a static balance of a rod [18]. To use this model, it is assumed that: the cross-section is symmetric, it does not deform under loading, and the effective diameter of the cross section is significantly smaller than the length of the rod. The Cosserat theory is capable of handling deformation modes of bending, torsion, transverse shear of a cross section, and extension. The Kirchoff model is a simplification where only bending and torsion are considered. Knowing the torsion can be beneficial, as with some designs, like a concentric tube robot, torsion can have a significant effect on the overall shape [19].

The shape of a rod is described by the position of its centerline $\boldsymbol{p}(s) \in \boldsymbol{R}^3$, and a rotation matrix, $\boldsymbol{R}(s) \in \mathrm{SO}(3)$, representing its material orientation as a function of arc length s, as shown in figure 1.1. The derivatives of $\boldsymbol{p}$ and $\boldsymbol{R}$ with respect to s are defined by

$$\begin{aligned} \mathbf{p}' &= \mathbf{R}\mathbf{v} \\ \mathbf{R}' &= \mathbf{R}\mathbf{u}^{\wedge}, \end{aligned} \tag{1.1}$$

where the kinematic variables $\mathbf{v}(s)$ and $\mathbf{u}(s)$ are the linear and angular rates of change of of a material attached transformation

$$\boldsymbol{g}(s) = \begin{bmatrix} \boldsymbol{R} & \boldsymbol{p} \\ \mathbf{0}^{\top} & 1 \end{bmatrix}$$

expressed in the local body-frame coordinates. The $\wedge$ operator, as defined in [20], maps $\mathbb{R}^3$ to $\mathfrak{so}(3)$ (the skew-symmetric matrices, the Lie algebra of the Lie group SO(3)).

Performing a static balance on a section of rod (Figure 1.1) and taking a derivative, we can describe the rates of change of the internal force vector $\mathbf{n}(s)$ and the internal moment vector $\mathbf{m}(s)$ with respect to arc length:

$$\begin{aligned} \boldsymbol{n}' &= -\boldsymbol{f} \\ \boldsymbol{m}' &= -\boldsymbol{p}' \times \mathbf{n} - \mathbf{l}, \end{aligned} \tag{1.2}$$

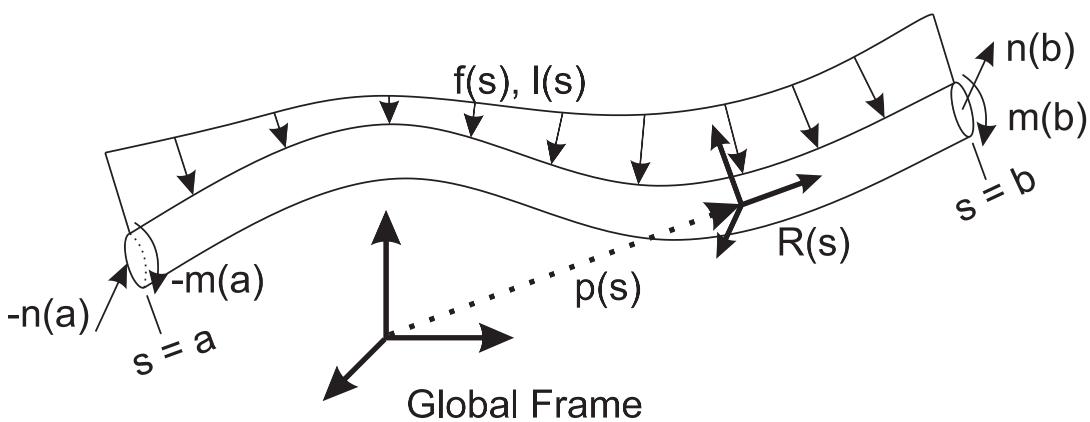


Figure 1.1: The Cosserat equations are derived by performing a static balance over a section of rod. All the state variables are defined with respect to arc length and are referenced to an arbitrary global frame.

where $\mathbf{f}$ and $\mathbf{l}$ are vectors defining the external distributed force and distributed moment per unit arc length applied to the rod. In this research we will be assuming $\boldsymbol{l} = \mathbf{0}$, and we aim to estimate $\boldsymbol{f}(s)$.

The internal force and moment vectors are related to $\mathbf{v}$ and $\mathbf{u}$ via a material constitutive law. Commonly, a linear law is used for static applications:

$$\boldsymbol{n} = \boldsymbol{R}\boldsymbol{K}_{se}\left(\boldsymbol{v} - \boldsymbol{v}^*\right), \qquad \boldsymbol{m} = \boldsymbol{R}\boldsymbol{K}_{bt}\left(\boldsymbol{u} - \boldsymbol{u}^*\right) \tag{1.3}$$

where $\boldsymbol{v}^*$ and $\boldsymbol{u}^*$ are appropriate kinematic variables of the rod in its stress free reference state ($\boldsymbol{u}^* = \mathbf{0}$ and $\boldsymbol{v}^* = [0\ 0\ 1]^\top$ for an initially straight rod), and $\mathbf{K}_{se} = \mathrm{diag}\left(GA, GA, EA\right)$ and $\mathbf{K}_{bt} = \mathrm{diag}\left(EI, EI, GJ\right)$ are stiffness matrices in terms of the rod's cross-sectional area A, Young's modulus E, shear modulus G, area moment of inertia I, and polar moment of inertia J. Combining Equations 1.1, 1.2, and 1.3, we have a system of differential equations that describe the evolution of $\boldsymbol{p}$, $\boldsymbol{R}$, $\boldsymbol{n}$, and $\boldsymbol{m}$ with respect to s.

1.2 Sensable Data

Force estimation is largely based upon what information can be gathered about a manipulator while it is in use. The information, that can be measured directly, determines what models can be used and how sensitive those models are to estimation errors. This also determines the posedness of the force estimation problem.

The easiest information to obtain is the translation/shaft angle (actuator variables) of an actuator and the forces/torques applied to it. Depending on the setup, often a user will have control of one or the other. Often translation/shaft angle can be measured though the use of encoder and force/torque sensed by either using the current draw of a motor or using a load cell.

Most methods are formulated by finding the deflection of chosen points along a manipulator. A common way to measure this is by placing electromagnetic trackers on the robot. The location and orientation of the trackers can be determined by measuring their interactions with a magnetic field.This can also be achieved by using computer vision to find colored points on the robot [21, 22, 23].

Recently, a significant amount of research has been put into methods for reconstructing the full shape of a manipulator. For tendon robots, Song [24, 10] presented a method that only required knowledge of the tip location and orientation. The paper outlines a process where the shape of a tendon robot can be approximated by a Bézier curve, using a Levenberg Marquardt algorithm for fitting. This is based on an assumption that tendon robots generally assume the shape of either a "C" or an "S". Another assumption, that is very common and required for this algorithm, is that the central backbone of the manipulator is inextensible. With this algorithm, a single magnetic tracker can be used to determine the entire shape of a tendon robot. Similarly, for catheters, it is common to use images taken through x-ray or MRI to reconstruct the shape. This is because the motion of a catheter can be assumed to be planar, so a single image is enough. Khoshnam [5] outlines an algorithm that isolates a catheter in an image and then fits a spline curve to the shape. This research will focus on using vision based methods.

Shape reconstruction has also been attempted by using Fiber Bragg Grating Sensors (FBG). This is because they are easily miniaturizable and can be used in MRIs [25]. FBG sensors work by using optical fibers designed to reflect a narrow wavelength spectrum of light. As the fiber experiences strain, the spectrum that is reflected changes predictably. This then provides a method to map spectrum measurements into strains along a manipulator. These strains can then be easily mapped into a curvature profile using a model.

Currently, the main issue with FBG sensors is that it can be difficult to measure torsion and axial extension. This is due to the bending strains being significantly larger than the axial strains. This discrepancy produces axial measurements that are too sensitive to be usable. This means that the sensor can only provide two components of a curvature vector. There has been research, though, into helically wrapping the FBGs so that torsion and axial extension can be sensed, providing a full curvature profile. Xu [7] outlines a design for this type of setup and gives a model that can map the sensed strains into curvatures.

1.3 Stiffness Control Laws for Continuum Robots

Due to the unique actuation and structure of continuum robots, traditional control methods are not always directly applicable. With the robots being naturally compliant, the interaction with the environment becomes difficult to predict. This can further be complicated by the fact that often these robots can be stiff in the axial direction but compliant in transverse directions.

Attempts have been made, though, to implement traditional techniques. One type was to use a switching control law that controlled position until contact and then control the force [26, 27]. This, however, neglects the ability to truly control the stiffness. There have also been attempts to use impedance based control methods in which the overall dynamic response can be chosen [28, 29, 30]. These methods are more focused on response to quick impacts as opposed to sustained contact. This has motivated the construction of unique stiffness control laws for continuum robots.

Often it is difficult to sense the interaction forces between the continuum robot and the environment. This has lead to control schemes that attempt to avoid needing this knowledge. In [31], a compliance controller was implemented that was designed to eliminate all contact forces during insertion of the robot. This allowed for the loading to be represented as a lumped parameter which could be minimized. This was shown to work well but was not able to fully control the compliance beyond maximizing it.

In [32, 33], an active stiffness control law was implemented. This approach used the measured deflection between the end effector and the target end effector position and required no knowledge of the actual tip force. This limited the applicability of this scheme to only be appropriate for situations where both the manipulator and environment are compliant. If either is stiff, this control law will fail to adjust the control effort due to a lack of discernible deflection.

Including sensed forces, however, allows for more robust control. In [34], a force sensor was placed between each segment of a multi-segmented continuum robot. They used this information to control robot such that it would deflect along the body while attempting to maintain the desired tip trajectory. This method was not concerned in exactly enforcing a desired stiffness of a point but the overall stiffness of the entire manipulator.

1.4 Force Estimation

Indirectly sensing the loading applied to a continuum robot is not novel in of itself. Our goal will be to provide a new more general algorithm and show how the results should be interpreted. There have been many approaches and multiple algorithms produced but there are various limitations to each one. Here we will detail some prior approaches and what assumptions were made in their implementations. A summary table is provided in Table 1.1.

Table 1.1: Summary of Literature on Force Estimation

Reference	Cite	Estimating	Dimensions	Robot/ Manipulator	Modeling Approach	Estimation Approach
Rucker (2011)	[14]	Tip Force	Planar	Tendon Actuated	Cosserat Rod Model	Kalman Filter
Khoshnam (2015)	[5]	Tip Force	Planar	Catheter	Pseudo Rigid Body	Linearization
Back (2015)	[35]	Tip Force	Planar	Catheter	Cosserat Rod Model	Linearization
Xu (2016)	[7]	Tip Force	3D	Concentric Tube Robot	Cosserat Rod Model	Linearization
Khan (2017)	[4]	Tip Force Without Axial Loads	3D	Concentric Tube Robot	Both PRB and Cosserat	Linearization
Yuan (2017)	[10]	Tip Force	Planar	Tendon Actuated	Constant Curvature	Nonlinear Optimization
Lindenroth (2017)	[9]	Tip Force	3D	Hydraulic Manipulator	Constant Curvature	Nonlinear Optimization
Black (2018)	[16]	Tip Wrench	3D	Parallel Continuum Robot	Cosserat Rod Model	Nonlinear Optimization
Back (2016, 2018)	[36, 37]	Tip Force	3D	Catheter	Constant Curvature	Linearization
Bajo (2010, 2012)	[38, 39]	Contact Locations	3D	Tendon Actuated	Constant Curvature	Dynamic Screw Theory
Santina (2020)	[12]	Contact Locations	3D	Pneumatic Actuated	Constant Curvature	Dynamic Disturbance Observer
Razban (2018)	[11]	Body Loading at Known Locations	3D	Catheter	Constant Curvature	Linear FEA
Bernth (2019)	[40]	Body Loading at Known Locations	Planar	Endoscope	Constant Curvature	Linear FEA
Zhang (2018, 2019)	[21, 22, 23]	Body Loading	3D	Soft robot	3D Elasticity	Linear FEA
Aloi (2019)	[41]	Body Loading	3D	Cantilevered Rod	Cosserat Rod Model	Nonlinear Optimization
Shiva (2019)	[13]	Body Loading	3D	Pneumatic Actuated	Constant Curvature	Not Stated
Heunis (2019)	[6]	Body Loading	3D	Catheter	Pseudo Rigid Body	Linearization
Qiao (2019, 2021)	[42, 43]	Body Loading	3D	Catheter	Cosserat Rod Model	Nonlinear Optimization
Al Ahmand (2021)	[44]	Body Loading	3D	Catheter	Cosserat Rod Model	Nonlinear Optimization
This work	N/A	Body Loading	3D	General/Tendon Actuated	General/Cosserat Rod Model	Nonlinear Optimization

1.4.1 Dynamic Observation

Using dynamic models, it is possible to locate points of contact in 3D space. One can track the motion of the manipulator to find when a contact occurs and determine the location of the contact by how the manipulator deviates from the desired trajectory as was done [38, 39]. Currently these methods have not been extended to sense the magnitudes of the forces but could be a stepping stone towards this goal.

Santina [12] used a dynamic observer to provide online state estimations of a multi-segment pneumatic robot. Due to the complexity of the dynamic model, they used a machine learning algorithm to correct the errors. The downside of this is that a set of experimental training data must be collected before hand. Using discrepancies in the shape of the manipulator, their observer allowed them to update the estimation of the external loading. This approach was able to locate the segment on which the contact occurred and the direction of the force but not the exact location.

1.4.2 Linearizing the Kinetostatics

With some models it may be appropriate to assume that the relationship between loading and deflection is linear around a given robot pose. This can result in using FEA modeling or using computations of a Jacobian that maps between loading and deflection. The downsides of these methods is that there is an implicit assumptions of small deflections.

The simplest example and implementation of this idea would be in [35] where they showed that using a planar version of the Cosserat Rod equations, with no body loads, one can setup a system of equations based off of curvature and its derivative. Back [36, 37] was also able to show that reliable results could gathered for a tendon robot with 3D applied tip loads and tracking tip deflection. These methods, though, would not be easily generalizing to body loading.

Zhang [21, 23] produced a fairly robust approach for force estimation on soft robots. They modeled a soft material by formulating 3D elasticity into a finite element analysis Hooke's law. They assumed that there was a point load at every marked point on the robot allowing for force estimation to be obtained from a simple inversion of the stiffness matrix. This method can only approximate the location of a force to its nearest FEA node. This method requires that an appropriate FEA model of the robot can be constructed and computed efficiently as well.

Razban [11] was able to implement a method that could find point forces applied at arbitrary body locations in a planar setup. The environment, in which the robot was traversing, had well defined locations of possible contact. They used a vision based system to isolate the robot and see if its pixels were mixing with the environments to isolate the contact points and measure the deflections of the manipulator. Using an FEA approach, a stiffness matrix can then be constructed with nodes aligning with the contacts. Locating the contact points this way could be very useful; however, it is unclear if this method could work with an environment that is less defined such as inside the body. In a similar vein, Bernth [40] also proposed a method for estimating planar forces applied at three known locations, with one being at the tip and used an FEA approach to solve for the loads.

The other way of approaching force estimation is to use the Jacobian mapping between the change in external loads and change in the measured shape. This approximation can allow for the estimation of body loads, as each node of a discretization can have a point force applied to it. With a constructed Jacobian, a simple Moore-Penrose psuedoinverse can be used to map from sensed shape to external loads.

In the simplest case of this implementation, the Jacobian can be found apriori using empirical data as in [5]. The problem with this is that there is a need for a large amount of data to cover a workspace. Only tip loading was considered but extending this method to body loads would drastically increase the amount of data needed to form a robust Jacobian.

Multiple other methods have been proposed that computes the Jacobian online. Xu [7] was able to derive a model that relates strain readings from a FBG to the external loading at the tip. Khan [4] was also able to derive a Jacobian relationship for both a psuedo-rigid body model and Cosserat based model for applied tip loads. Heunis [6] was able to achieve full 3D estimation of arbitrary body loading using this method. It is important to note that often the Jacobian is ill conditioned. Without using an iterative optimization scheme, the Jacobian may not be efficiently or stably minimizing the shape error.

1.4.3 Nonlinear Approaches and Numerical Optimization

In this research, we will focus on using a nonlinear numerical optimization approach for the force estimation algorithm. The basis of this is that a numerical routine can update a guess for the external loading to satisfy some constraint on the manipulator. These constraints tend to be the pose of the manipulator and static equilibrium.

For a tip force, this problem can generally be formulated using only static equilibrium and actuator forces. This is referred to as intrinsic force sensing. This is highlighted in [16] for a parallel continuum robot and [9] for a planar pneumatic robot. Rucker [14] also provided an example of this for a tendon robot, however, a Kalaman filter was used to estimate the forces and forward kinetostatics to replace numerical optimization.

Yuan [10] used a simplified planar constant curvature model for estimating forces at the tip. Using a static balance, they were able to relate the internal wrenches to deflection angles and set up a nonlinear system of equations. With measured curvatures, they used a numerical optimization scheme to find a tip load that minimized the errors in the curvature. Due to the model derivation, extending the modeling approach to 3D with external loading and torsion is non trivial.

Shiva [13] used a model based approach for estimating 3D body loads. This method was only tested and implemented for scenarios when a single load was applied. For a pneumatic robot, they were able to relate actuation pressure to axial strain in the chambers. With a constant curvature model of the backbone, a relationship between an externally applied load, actuation strain, actuation pressure, and resultant external moment can be written. The implementation of this model for force sensing was not discussed in the paper and it is unclear how to proceed from this point. The main point to mention here though, is that this approach is not general and will only work for robots that can use this specific model.

Finally, Qiao and Ah-Ahmed [42, 43, 44] implemented a force estimation algorithm similar to that of [41]. They proposed two versions, one that fits measured curvature and one that fits measured shape. They were able to detect multiple contacts in 3D but only tested this on passive instruments. One interesting note they made was that solving the minimization problem with a Kalman Filter was able to decrease runtime over a Levenberg-Marquardt algorithm.

1.4.4 Contribution of Dissertation

This work adds on to the existing literature in two parts. First, we will be presenting a integral based stiffness control law for an end effector of a continuum robot. This will allow for hybrid position/force control without switching and will be able to handle situations where either the robot or the environment is rigid.

Second, we will be presenting a general 3D force estimation approach that is non model dependent and allows for arbitrary body loading. The method is based on using sensed shape information and fitting a model predicted shape to the measured. This is an ill-posed problem but useful information can still be gathered. Along with this algorithm, a discussion of force parameterization and uncertainty will be provided.

Chapter 2

Stiffness Control of an End Effector

Continuum robots can simultaneously exhibit both compliant and stiff behavior at the end effector and can be expected to interact with both compliant and rigid environments. To help improve the control of interactions during contacts, we are proposing a stiffness control law that allows for a hybrid of position and force control. Our goal will be to make the manipulator itself, act as if it has a specific stiffness as in Figure 2.1. This relationship is defined as:

$$\boldsymbol{F} = \boldsymbol{K}_d(\boldsymbol{p} - \boldsymbol{p}_r). \tag{2.1}$$

An integral based control law can be used to enforce this relationship by tracking the error between the actual and desired displacement:

$$\begin{aligned} \boldsymbol{p}_t &= \beta(\boldsymbol{K}_d^{-1}\boldsymbol{F} - \boldsymbol{p} + \boldsymbol{p}_r) + \boldsymbol{p}_c \\ \boldsymbol{R}_t &= \boldsymbol{R}_r. \end{aligned} \tag{2.2}$$

The notation for this section is summarized in Table 2.1. This approach will track the desired reference pose in free space, but adjust the actual position whenever an external load is sensed and maintain the reference orientation. This control law is capable of handling situations where the manipulator is in contact with both

Table 2.1: Notation for Stiffness Control

Symbol	**Definition**
$\boldsymbol{p}$	Position of end effector
$\boldsymbol{R}$	Orientation of end effector
$\boldsymbol{g}$	Full end effector frame
$\boldsymbol{K}_d$	Desired stiffness matrix for end effector
$\boldsymbol{r}$	Rod Routing
$\boldsymbol{F}$	Force exerted by end effector
$\boldsymbol{q}$	Actuator variable
$\boldsymbol{\tau}$	Force applied to actuator
$\boldsymbol{J}$	Jacobian mapping from actuator variables to pose
α	Damping parameter for inverse kinetostatics
β	Integral gain
$\hat{\cdot}$	Model estimated value
Subscript r	Reference/free space command
Subscript t	Target command before damped inverse kinetostatics
Subscript c	Commanded value
Subscript s	Directly sensed value
Subscript i	A variable of rod “i”

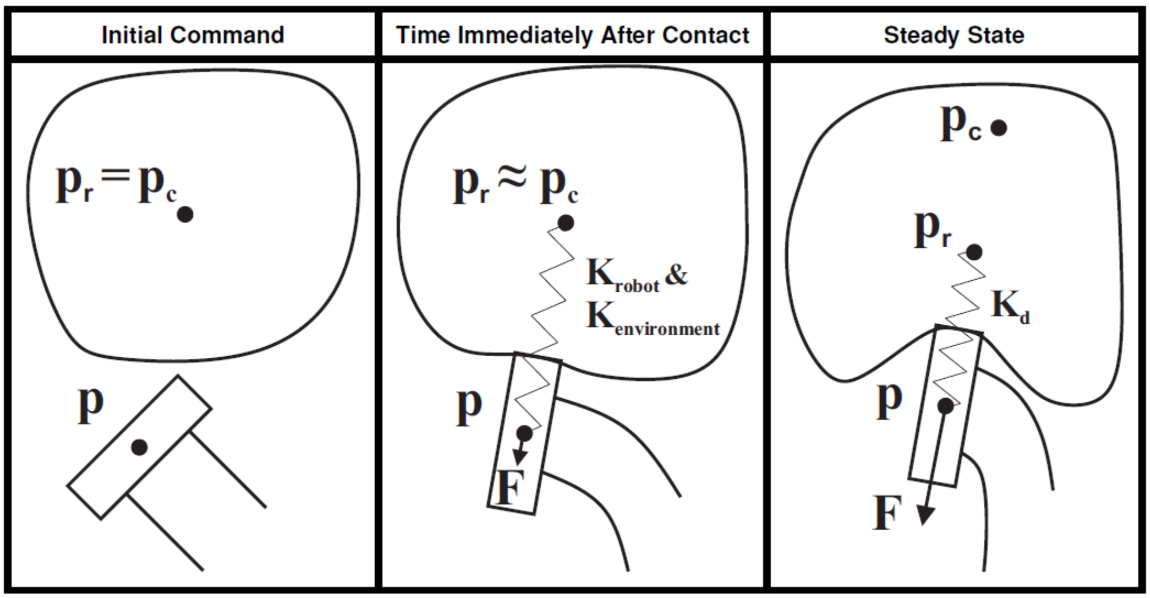


Figure 2.1: The goal of a stiffness controller is for the end effector to be positioned such that the resultant force and position of the end effector is related by a desired stiffness constant at steady state. Here we present an example where, after contact, the robot becomes coupled with the environment in a system of springs.

compliant and stiff environments. It is also capable of working when the stiffness of the end effector is high and the environment is compliant. The algorithm is capable of both decreasing and increasing the stiffness experienced at the end effector, and allows for us to implement different stiffness constants for different loading directions. Lastly, this method does not involve switching between position control and force control but allows for both to be controlled simultaneously. Now, we will provide an example of how this control law can be implemented for a Parallel Continuum Robot. This work was published in the ASME DSCC 2018 proceedings [17].

2.1 Kinetostatics and End Effector Force Estimation for a Parallel Continuum Robot

This section is summary of work published in [45, 46, 47, 16]. This will outline how a Parallel Continuum Robot (PCR) is modeled and how this model can be used to perform forward kinetostatics, estimation of tip loads, and inverse kinetostatics. A PCR is constructed with multiple push pull rods arranged topologically parallel. These rods are attached to actuators and terminate at an end plate, which we refer to as the end effector. In this work, we used linear actuators, therefore the actuator variable, q, refers to the actuator stroke length. Arranging six rods in a Stewart-Gough configuration allows for full six degree of freedom control of the end effector. An example construction is shown in Figure 2.2.

The modeling framework is based on classical Cosserat rod mechanics where the shapes of the individual links can be described via the differential equations of a Cosserat rod and boundary constraints can be applied for the entire robot. The

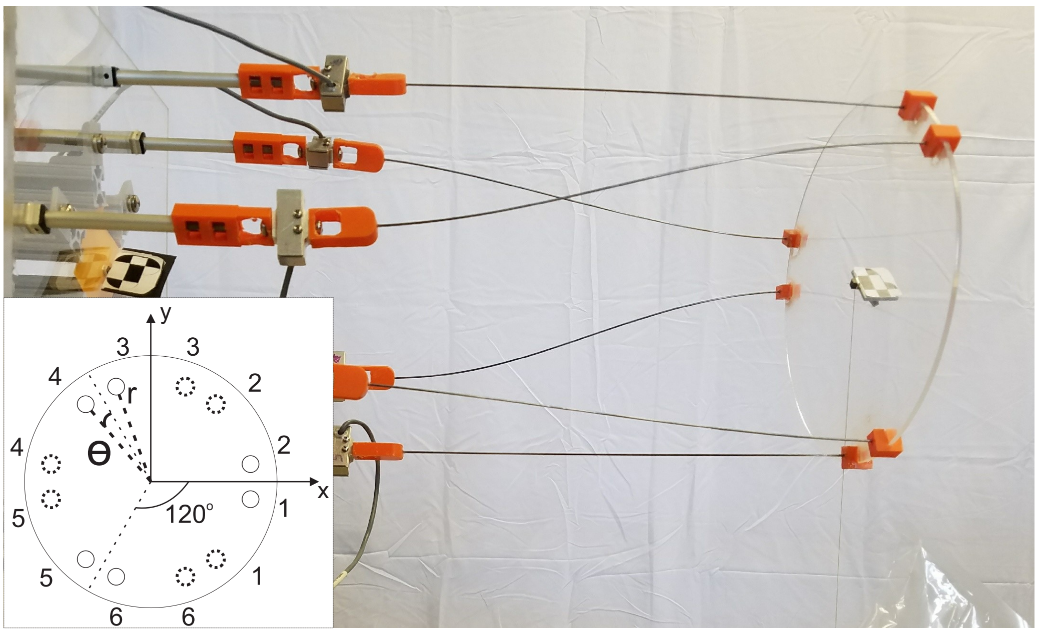


Figure 2.2: This is the PCR used for the stiffness control experiments. The linear actuators can be seen at the base. Using PLA 3-D printed mounts, the actuators are connected directly to the load cells. The load cells are then connected to the spring steel rods. The diagram, in the bottom left, shows the hole patterns for how the rods were affixed. The number represents a specific rod. The solid circles represent where the rods intersect the base. The dashed circles represent where the rods intersect the end-effector. "r" is the distance from the center that the rods are affixed. θ is the angular offset of a rod from the x, 120°, or 240° axes.

distal boundary conditions include static equilibrium equations:

$$\begin{aligned}&\sum_{i=1}^{n}\left[\boldsymbol{n}_i\left(L\right)\right]-\boldsymbol{F}=\boldsymbol{0}\\&\sum_{i=1}^{n}\left[\boldsymbol{p}_i\left(L\right)\times\boldsymbol{n}_i\left(L\right)+\boldsymbol{m}_i\left(L\right)\right]-\boldsymbol{p}\times\boldsymbol{F}-\boldsymbol{M}=\boldsymbol{0}\end{aligned}\tag{2.3}$$

where $\boldsymbol{p}$ refers to the position of the end effector, $\boldsymbol{p}_i$ refers to the position of a specific rod “i”, and L refers to the total length of a rod.

A second condition comes from the geometry of the manipulator. The links are attached to the end effector at a specified location from the centroid, r_i, which is fixed for all poses. This gives the distal end of a rod the boundary condition

$$\boldsymbol{p}+\boldsymbol{R}\boldsymbol{r}_i-\boldsymbol{p}_i=\boldsymbol{0}\text{ for } i=1...n\tag{2.4}$$

In the case of the manipulator considered in this paper, the link joints at the end effector are fixed. The internal moment vector $\boldsymbol{m}_i(L)$ must balance with the external applied moment $\boldsymbol{M}$. The orientation of each rod at the distal end must match the orientation of the end effector. Written in a reduced form, this constraint is:

$$\left[\log\left(\boldsymbol{R}_i\left(L\right)^{\top}\boldsymbol{R}\right)\right]^{\vee}=\boldsymbol{0}\text{ for } i=1...n\tag{2.5}$$

where log() is the matrix natural logarithm.

At $s_i = L$, the boundary conditions consist of the geometric constraints for each link and the equilibrium equations for the entire end effector. Since these equations involve the distal variables of all the rods (pose, internal force, and internal moment), this set of distal boundary conditions couple all the independent rod models together. In this work, we used a nonlinear least squares Levenberg-Marquardt algorithm. This is a shooting method where the proximal values of each rod is guessed, the Cosserat rod equations are integrated, and then the boundary conditions are evaluated. The actuator variables dictate the initial global z-coordinate of each rod such that $p_i(0)e_3 = q_i$ where e_3 is third standard basis vector.

For our standard forward kinetostatics with tip loading, our unknowns are just the proximal wrenches of each rod, $\boldsymbol{w}_i$. Our known inputs are q_i, $\boldsymbol{F}$, and $\boldsymbol{M}$. Solving the boundary value problem, we can estimate the pose of the end effector, $\hat{\boldsymbol{g}}$, and the forces on each actuator $\hat{\boldsymbol{\tau}}$. The hat, $\hat{\cdot}$, indicates and estimated value. We notate this functionally as:

$$\{\hat{\boldsymbol{g}}, \hat{\boldsymbol{\tau}}\} = FK_{q,F,M}(\boldsymbol{q}, F, M). \tag{2.6}$$

For the force estimation algorithm, we can simply shift around our unknown values and know values to estimate both the pose and tip wrench. This requires us to measure the forces on the actuators which is achieved using load cells. Here, our unknowns are $\boldsymbol{w}_i$, $\boldsymbol{F}$, and $\boldsymbol{M}$. Solving the boundary value problem gives us an estimate of $\hat{\boldsymbol{g}}$, $\hat{\boldsymbol{F}}$, and $\hat{\boldsymbol{M}}$. Our known inputs are $\boldsymbol{q}$ and $\boldsymbol{\tau}$. We notate this functionally as:

$$\{\hat{\boldsymbol{g}}, \hat{\boldsymbol{F}}, \hat{\boldsymbol{M}}\} = FK_{q,\tau}(\boldsymbol{q}, \boldsymbol{\tau}). \tag{2.7}$$

A similar method can also be used to formulate the inverse kinetostatics, however, for the sake of efficiency, we can use equation 2.7 to compute a Jacobian to approximate the inverse kinetostatic mapping from tip pose to actuator variables.

2.2 End Effector Stiffness Control for a Parallel Continuum Robot

The control loop can be described in three main blocks: the control law, inverse kinetostatics, and the model based estimation. A simplified block diagram is provided in Figure 2.3 and the full block diagram used for the implementation in this chapter is provided in Figure 2.4. The robot is assumed to use low-level servo-controlled actuators. A standard kinematics control loop is implemented by taking a damped

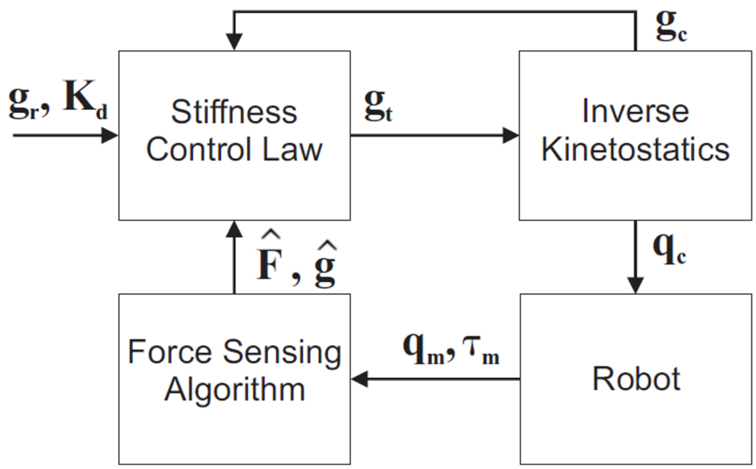


Figure 2.3: This stiffness controller is an integral approach that enforces a desired stiffness constant for a continuum robot. It consists of a block which does kinetostatic control, a block that modifies the target position to achieve the desired stiffness, and a block which solves the nonlinear kinetostatics formulation to sense end effector force and pose.

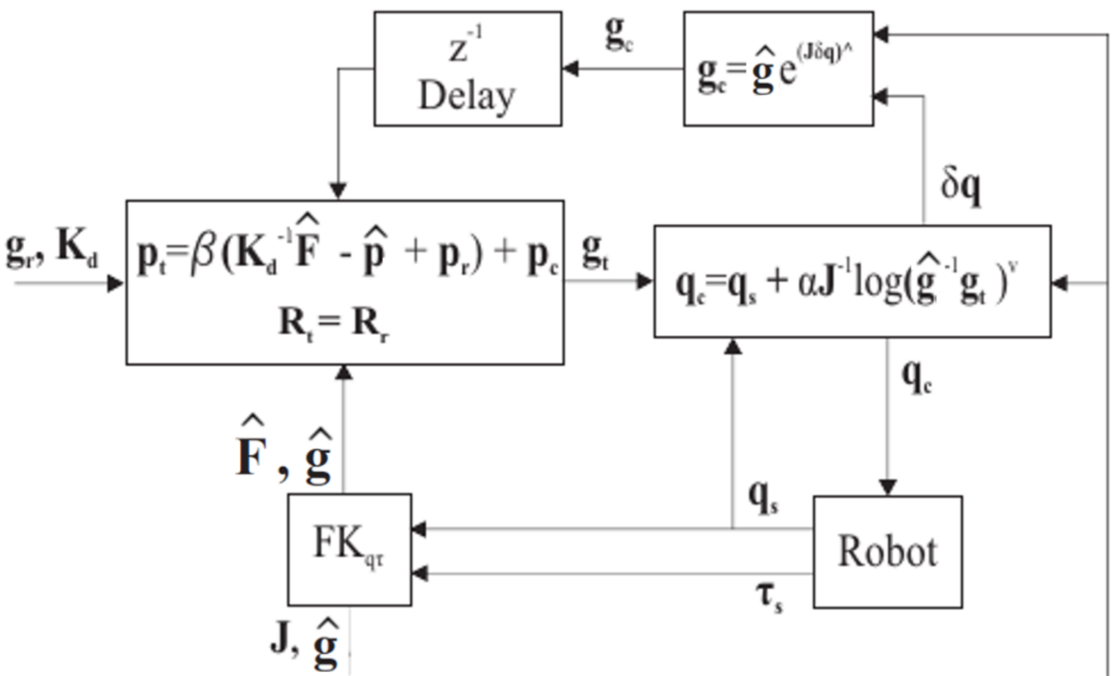


Figure 2.4: The full stiffness control block diagram consists of the stiffness control law, the model based estimator for the pose and the tip force, and uses a damped linear approximation to perform the inverse kinetostatics.

Newton step:

$$\boldsymbol{q}_c = \boldsymbol{q}_m + \alpha \boldsymbol{J}^{-1} \log(\hat{\boldsymbol{g}}^{-1}\boldsymbol{g}_t)^{\vee}, \tag{2.8}$$

where log() is the matrix logarithm and the Jacobian, $\boldsymbol{J}$, is calculated using equation 2.7. The target pose $\boldsymbol{g}_t$ is formed from the user-input reference pose, $\boldsymbol{g}_r$, and desired stiffness $\boldsymbol{K}_d$. The reference orientation is maintained by passing it along without modifications, while the target position, $\boldsymbol{p}_t$, is generated by the integral control law in equation 2.2. $\boldsymbol{p}_c$ is the position part of the pose $\boldsymbol{g}_c$ associated with the most recent commanded actuator configuration $\boldsymbol{q}_c$. Since the inverse kinetostatics is being damped, commanded values are approximated to first order using the manipulator Jacobian and the actuator update as:

$$\boldsymbol{g}_c = \hat{\boldsymbol{g}} e^{(\boldsymbol{J}(\boldsymbol{q}_c - \boldsymbol{q}_m))^{\wedge}}. \tag{2.9}$$

At steady state, equation 2.1 will be satisfied and the commanded pose will be equivalent to the target pose. When the robot is not in contact with the environment, the steady state commanded pose will be equivalent to both the target and reference pose.

2.3 Experimental Validation

The goal of these experiments is to show that the control scheme is capable of dictating the output stiffness of a parallel continuum robot. We present tests for multiple $\boldsymbol{K}_d$ matrices, with the manipulator in multiple poses, by applying known loads and measuring the resulting displacements while the robot is controlled with the approach outlined above.

A 6-link PCR, seen in Figure 2.2, was constructed for use in the experimental validation of the proposed stiffness controller. For the links, spring steel rods were used, 0.3 m in length and 1.4 mm in diameter. The distal ends of the rods were connected to the end-effector platform (0.125 inch clear acrylic) by using 3-D printed

PLA mounts. The rods were adhered to the mounts, and the mounts were adhered to the acrylic. The pattern for how the rods were affixed can also be seen in Figure 2.2. The rods were arranged around a radius of 0.11 m and with an angular offset of 10°. The base of the rods were also adhered to PLA mounts. The mounts were then attached to the load cells using screws. Then, another PLA mount was used to connect the load cells to the actuators, again using screws. The actuators were placed around a radius of 0.12 m and with angular offsets of 18.6°. Each pair of rods is distanced equally around the circle at 120°. The pairs on the end-effector are rotated (relative to the base) by 60°. In this design, a base plate was used to minimize the compliance of the actuator mounts. It was located at the base of the actuators to avoid interfering with the rods. To obtain displacement measurements, a visual tracking marker was affixed to the center of end-effector. To apply the weights to the end-effector, a plastic bag was tied with thread to the visual tracking marker.

We used Omega Engineering Load Cells LC703-10, which are rated up to 10 lbs at the base of each actuator mount. A SparkFun Load Cell Amplifier HX711 was also used for each load cell. This amplifier is rated to amplify ±20 mV. Empirically, we found that the load cell readings were accurate to ±0.5 N. The prototype was actuated by Firgelli L16-140-35-12-P linear actuators (accurate to ±0.3 mm) with our low-level position controller. To measure the tip position of the manipulator, we used a Claron Technology Inc Micron Tracker 3, model XB3-BW-H360 with accuracy of ±0.2 mm.

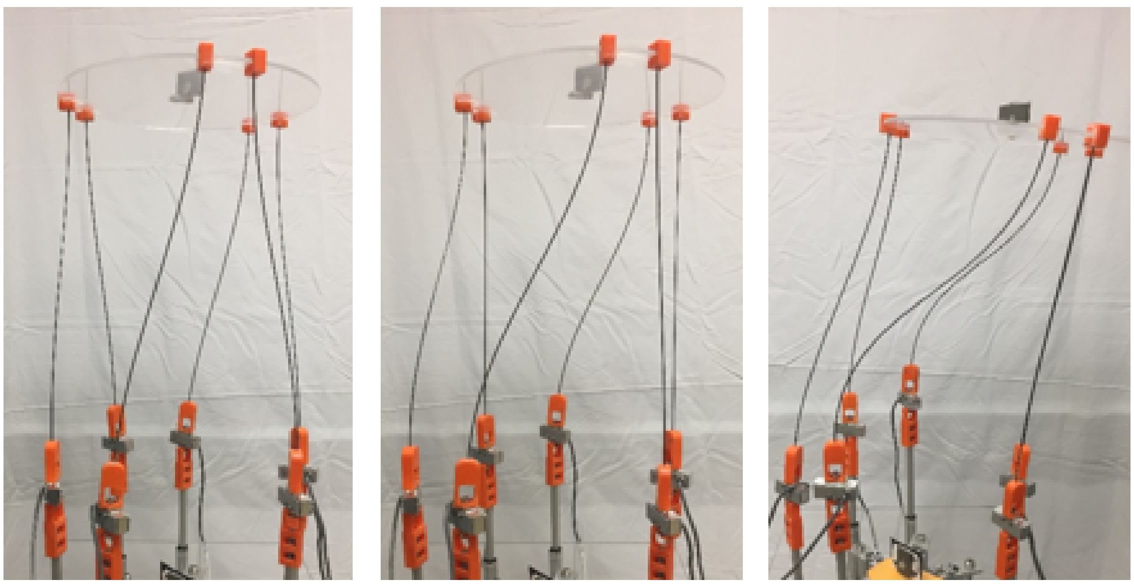
We calibrated the kinetostatic model parameter E (Young's modulus) in order to account for any unmodeled compliance in the device's structure. We applied known weights to the manipulator in the transverse directions and measured the resulting deflection, to get a set of test data. Then, using our forward kinetostatics model, we simulated the deflections of the manipulator under the same loads and performed a least-squares calibration of E using Matlab's `fminbnd()` to fit the data. The calibrated modulus was $E = 96.86$ GPa.

We tested the proposed stiffness control approach by specifying three desired diagonal stiffness matrices with equivalent stiffness in all 3 Cartesian directions. For the integral control gains, we set $\beta = 1$. Three stiffnesses were tested: 500, 200, and 166.7 N/m. For each, loads were applied to the end-effector in both axial and transverse directions.

For tests with loads in the axial direction (the z-direction in Figure 2.2), two poses were used: one in the nominal configuration and one where the end-effector was translated 4 cm in the y-direction without rotation. For each of these poses, the tests masses ranged from 0 to 0.6 kg with 0.1 kg increments.

For the tests with loads in transverse directions, three poses were used: the nominal pose, the translated pose, and an additional bent pose (Figure 2.5). In the "bent" configuration, the end-effector was both translated and rotated. To apply loads in the transverse directions, the manipulator was laid on its side, and weights were hung from the center of the end-effector so that gravity would apply a force in the xy-plane (labeled in Figure 2.2). The masses were applied 195° from the x-axis of the base. For each of the transverse pose tests, the masses ranged from 0 to 0.6 kg with 0.05 kg increments.

Figure 2.6 shows the resulting force/displacement plots of the experimental dataset overlaid with the desired stiffness lines. The controller was successful in dictating stiffness in both compliant and rigid directions with a maximum error of only 5 mm; however, there is a clear distinction between the algorithm performance with axial and transverse loads. This can be attributed to modeling errors. Since there is very little natural compliance in the axial direction, axial end-effector forces are more directly transferred to the actuators with little structural deflection, and the effect of deflection modeling error is reduced. The transverse case entails larger deflections and a higher dependence on model accuracy.



(a) Nominal Pose (b) Translated Pose (c) Bent Pose

Figure 2.5: Poses (a) and (b) were used for the axial tests. Poses (a), (b), and (c) were used for the transverse tests.

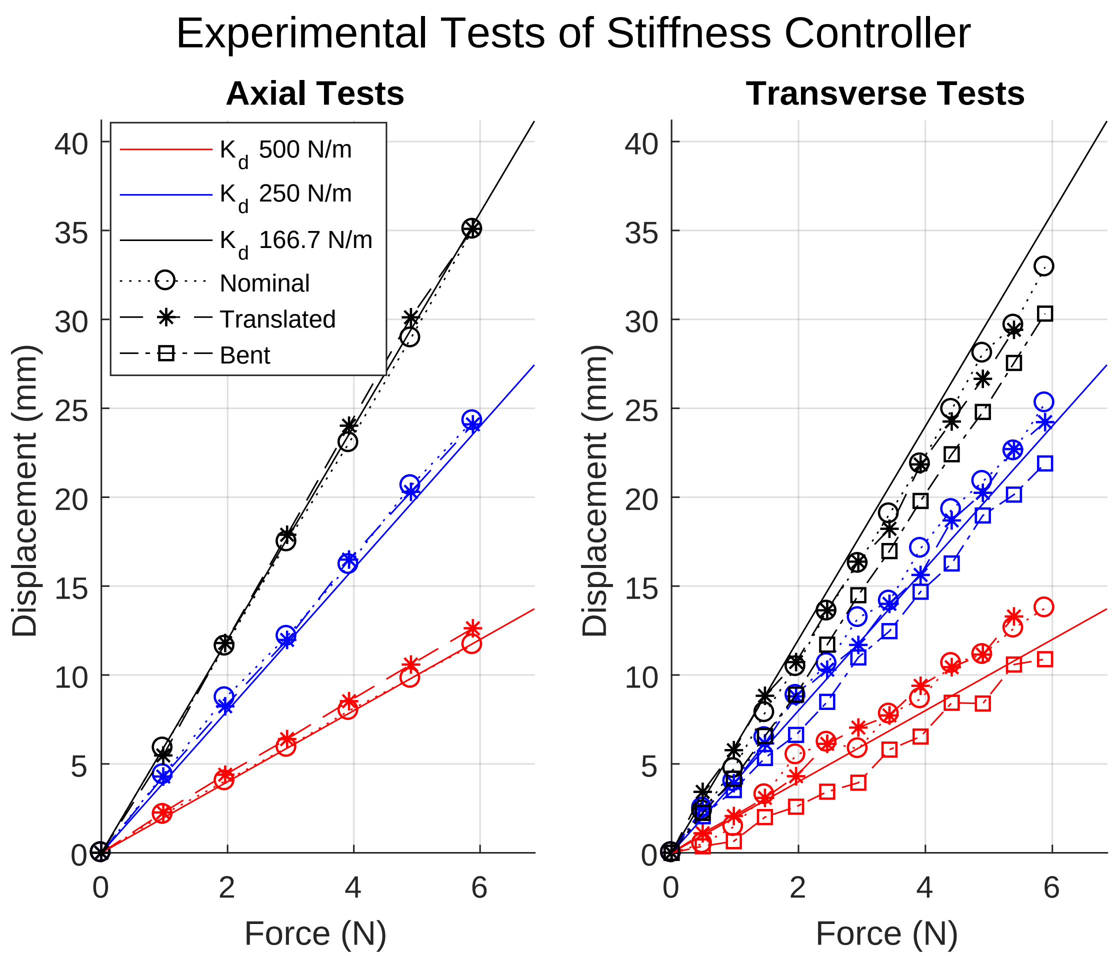


Figure 2.6: Three desired stiffnesses ($\boldsymbol{K}_d$) were tested on a parallel continuum robot. These tests include forces placed in the axial and transverse directions of the end-effector. These stiffnesses were also tested with the manipulator actuated into a bent and translated pose. This figure shows the linear stiffness behavior measured from the experiments compared to the desired stiffness. With the x-axis representing the forces applied and the y-axis representing the total displacement, the slope of the line is the stiffness.

Chapter 3

Estimating Body Loads Applied to an Elastic Rod

In this chapter, we will present a general force estimation algorithm that is capable of estimating multiple point contacts and distributed loads applied at arbitrary locations along the length of a flexible rod. It was assumed that no axial loading was applied along the body or at the tip of the rod. Experiments with planar point contact loading were preformed for partial validation and a qualitative example of a distributed load with out of plane bending was included as well. A summary of the notation for this chapter is include in table 3.1.

3.1 Shape Based Force Estimation

The force estimation was formulated such that the only necessary input to the algorithm was shape data. Using a machine vision based approach, N_p number of points along the rod were collected. Each point is notated as $\hat{\boldsymbol{p}}_i$ and is located at arc length s_i. This algorithm was formulated to find a distributed load along the arc length s, parameterized as $\boldsymbol{f}(s)$, that would fit the model based prediction of the rod shape to the experimentally measured shape as depicted in Figure 3.1. With the

Table 3.1: Notation for Rod Force Estimation

Symbol	**Definition**
$\boldsymbol{p}$	Position of a point along the rod
$\boldsymbol{R}$	Orientation of a point along the rod
$\boldsymbol{n}$	Internal force
$\boldsymbol{m}$	Internal moment
$\boldsymbol{f}$	Distributed force
$\boldsymbol{c}$	Vector of boundary conditions
$\boldsymbol{x}$	Vector of unknown inputs
$\boldsymbol{\gamma}$	Vector of parameters that define $\boldsymbol{f}(s)$
$\boldsymbol{U}$	Vector of uncertainty parameters
F	Magnitude of point force
μ	Location of point force
δ	Euler beam deflection
N_p	Number of data points
$\hat{\cdot}$	Model estimated value
Subscript ext	External load
Superscript l	Local value
Superscript x or y	x or y component of vector

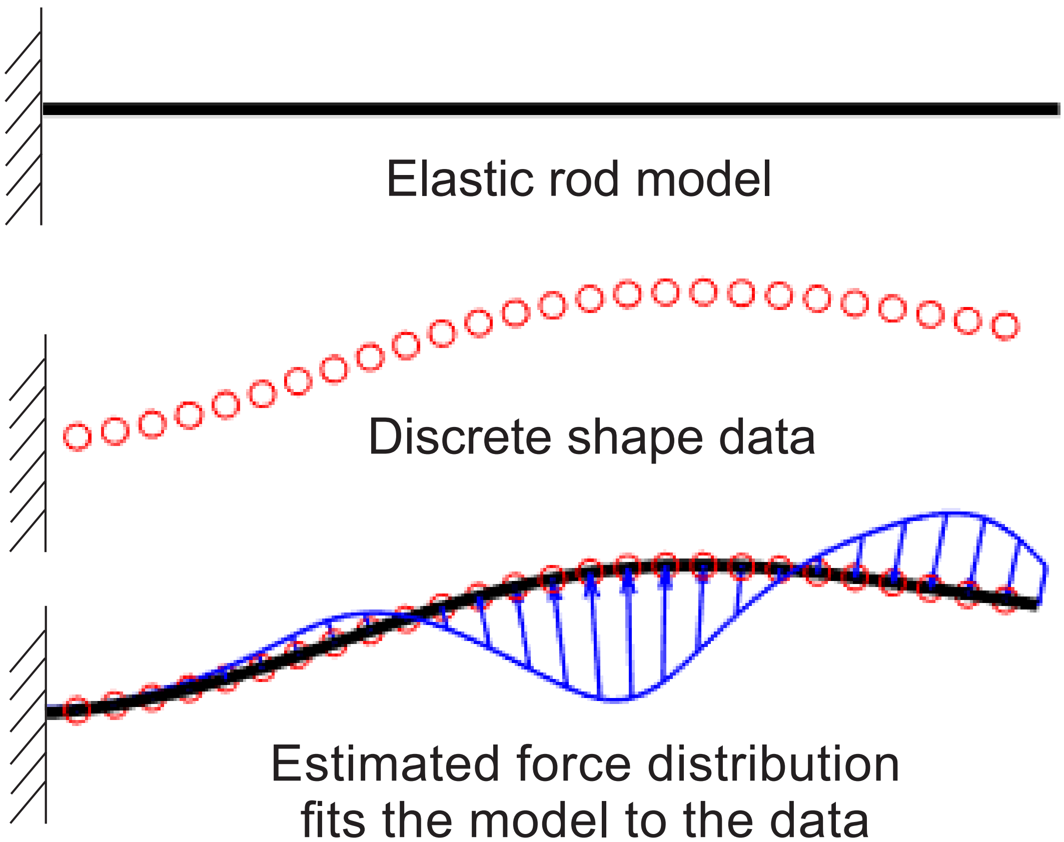


Figure 3.1: Our approach to the force estimation problem is to find set of forces which minimizes the error between discrete shape data and a continuous rod model

assumption that the tip of the rod is not loaded, the problem is constrained to have zero internal forces or moments at the tip:

$$\boldsymbol{c}(\boldsymbol{x}) = \begin{bmatrix} \boldsymbol{n}(L) \\ \boldsymbol{m}(L) \end{bmatrix} = \mathbf{0} \tag{3.1}$$

where L is the length of the rod.

We formulated this algorithm as a constrained nonlinear minimization problem based on the root-mean-squared error of the shape fit:

$$\begin{gathered} \underset{\mathbf{x}}{\text{minimize}} \quad \boldsymbol{p}_{rms}(x) = \sqrt{\frac{1}{N_p}\sum_{i=1}^{N_p} \|\hat{\mathbf{p}}_i(\mathbf{x}) - \mathbf{p}_i\|_2^2} \\ \text{subject to} \\ \mathbf{c}(\mathbf{x}) = \mathbf{0} \end{gathered} \tag{3.2}$$

where $\hat{\boldsymbol{p}}_i(\boldsymbol{x})$ is the 3D position generated by the Cosserat rod model evaluated at arc length s_i and $\boldsymbol{x}$ is the vector of unknown parameters:

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{n}_0^\top & \boldsymbol{m}_0^\top & \boldsymbol{\gamma}^\top & \boldsymbol{U}^\top \end{bmatrix}^\top \tag{3.3}$$

where $\boldsymbol{n}_0$ is the base force, $\boldsymbol{m}_0$ is the base moment, $\boldsymbol{\gamma}$ are the parameters that define $\boldsymbol{f}(s)$, and $\boldsymbol{U}$ are parameters to account for uncertainty in the measured data and registration error.

3.1.1 Ill-posedness of the Inverse Problem

It is easy to map from distributed loading to a unique deformation using a forward kinetostatic model; however, solving the inverse of this problem leads into the common issue of being ill-posed. Theoretically, there is a unique force distribution that results in a certain shape of the rod. Finding a load that can recreate the shape should, therefore, be sufficient to sense the loading. This, however, is limited in real world applications due to finite sensor resolutions. Therefore, special consideration must be made when choosing a parameterization for the loading and interpretation of the results. To improve the posedness, it is necessary to place constraints on the solution space and, in some cases, introduce additional sensed information into the problem.

As an analytical example of this problem, lets consider an Euler-Bernoulli beam. Given a straight beam, we apply a distributed load in the shape of a sine wave. We will be using a continuous root mean square (RMS) of the deflection to quantify the total change in shape. Using the differential equation relating shape and load, from Euler-Bernoulli beam theory, we can calculate the continuous RMS of the deflection:

$$
\begin{gathered}
f(s) = A\sin(\omega s) = \frac{1}{EI}\frac{d^4\delta(s)}{ds^4}, \\
RMS = \sqrt{\frac{1}{L}\int_0^L \delta(s)^2 ds} = \frac{\sqrt{2}AEI}{2\omega^4}.
\end{gathered}
\tag{3.4}
$$

where A is the amplitude of the cosine wave, ω is the frequency, s is the position along the beam, L is the total length, EI is the flexural rigidity, and δ is the deflection of a point. From this we can see that as the frequency of the distributed load increases, the total RMS of the deflection decreases by a power of four. This effect is graphically represented in Figure 3.2. This implies that there is an infinite combination of frequencies and amplitudes that can produce a change in shape that is less than some arbitrary minimum. In practical application, this can stall a minimization algorithm or cause convergence to a local solution that does not effectively describe the loading.

Furthermore, the design of the manipulator itself can also effect what can be sensed using shape data alone. As cited in [4], a robot with large axial stiffness will experience strains less than that of any viable sensor resolution. This means force estimation methods, based on deformation alone, cannot sense axially applied loads. This enforces special considerations on how the distributed loading is parameterized. This is because the shape of the manipulator is vastly dominated by the resultant curvature due to the internal moments as opposed to the shear due to internal forces.

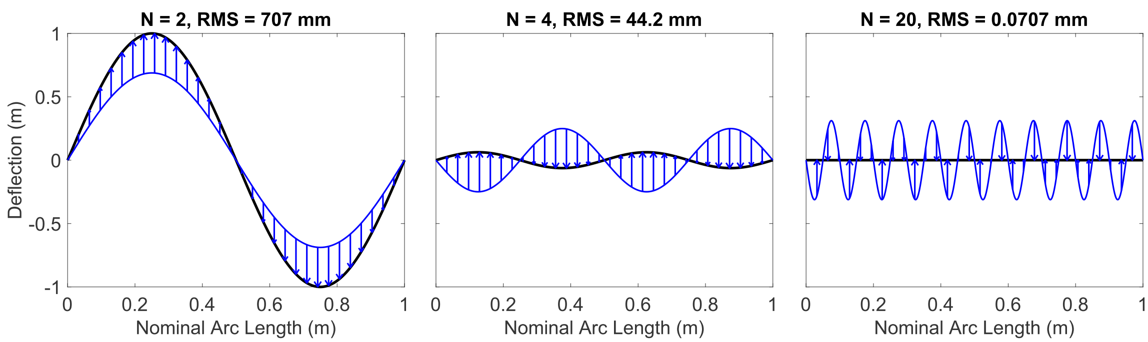


Figure 3.2: As the frequency of a sinusoidal distributed load is increased, the resultant deformation of a beam decreases. The beam in this example is $1m$ in length and has a flexural rigidity of $1\,\text{N-m}^2$. The load is a harmonic sinusoid with a constant amplitude of $A = 16\pi^4 \frac{N}{m}$ and an increasing frequency of $\omega = N\pi \frac{rad}{s}$.

Given a single point force applied to the backbone, there is an infinite combination of magnitudes and angles that can produce approximately the same reaction moment at the base if the axial component is unknown. An example of this is given in Figure 3.3.

3.1.2 Uncertainty of the Force Estimation

To further understand this relationship between force estimation and shape sensing, let us consider sensitivity analysis. Given an Euler Bernoulli beam, lets apply a load of magnitude F at point μ as in Figure 3.4. The resultant deflection at the tip is:

$$\delta(L) = \frac{F(3L\mu^2 - \mu^3)}{6EI}. \tag{3.5}$$

If a measured tip deflection was used to estimate a force, errors would manifest in both magnitude and location. We can directly calculate this relationship by taking partial derivatives of equation 3.5:

$$\frac{\partial F}{\partial \delta} = \frac{6EI}{(3L\mu^2 - \mu^3)} \qquad \frac{\partial \mu}{\partial \delta} = \frac{EI}{F(L\mu - \frac{1}{2}\mu^2)}. \tag{3.6}$$

In Figure 3.5, the sensitivity of a point load estimation with respect to tip deflection is plotted for varying force magnitudes and locations. This shows that an estimation of magnitude will become more uncertain as the force approaches the base. Intuitively, this is because base forces result in overall less deflection. This effect can also be seen in the estimation of force locations. It is important to note, that the location uncertainty is also inversely proportional to the force magnitude. Given two forces, it would not be directly apparent which estimation is more certain based only on their locations.

When using a nonlinear model, such as Cosserat Rod Theory, we can compute these partial derivatives numerically. In Figure 3.6 we can see that the Cosserat Rod Theory agrees with the equations from 3.6 in the linear region; however, as the force

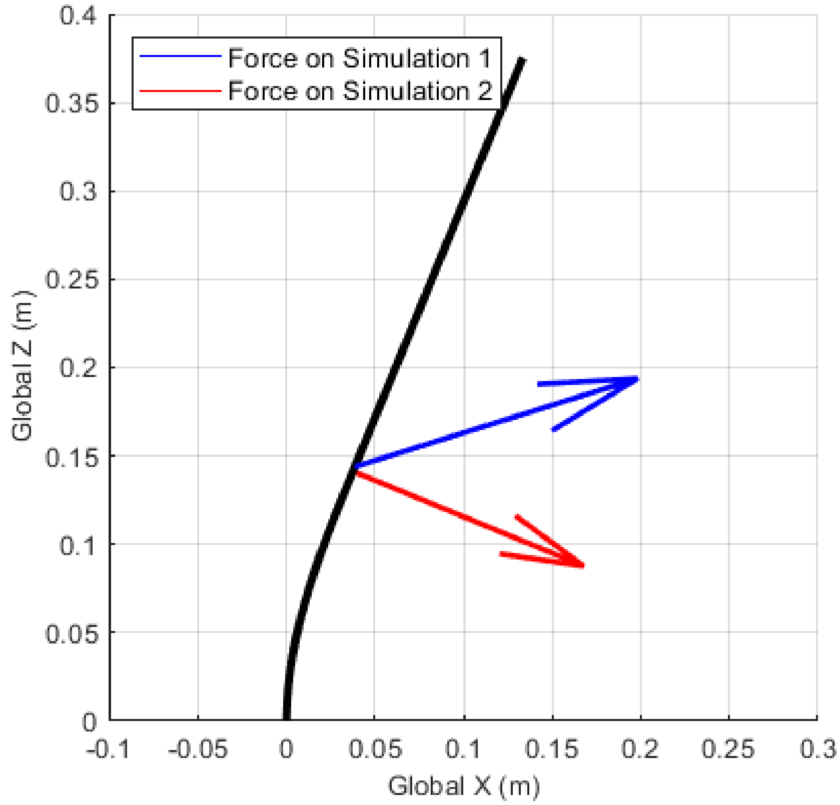


Figure 3.3: Two Cosserat rod simulations were ran. One with a load applied normal to the rod surface and one with a load that had an axial component. Due to the rod being axially stiff, both scenarios resulted in a rod shape that was nearly identical.

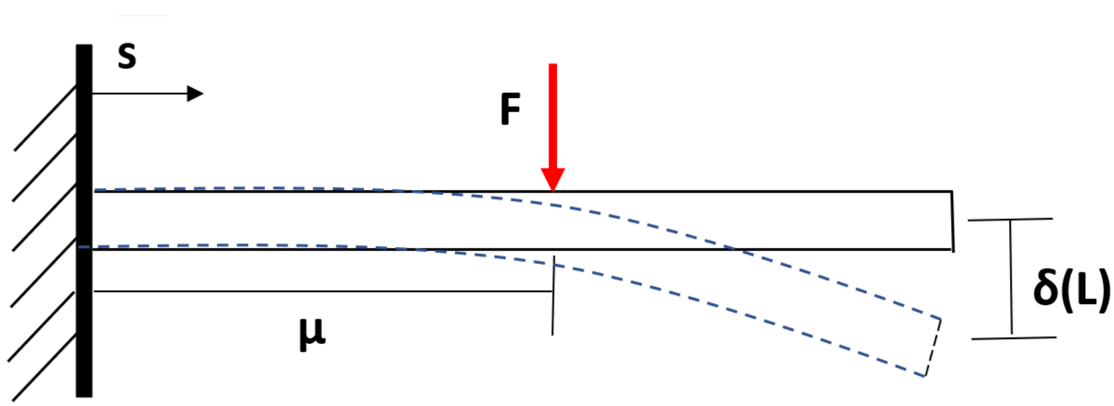


Figure 3.4: In traditional beam theory, a force F applied at point μ will create a deflection of δ at the tip.

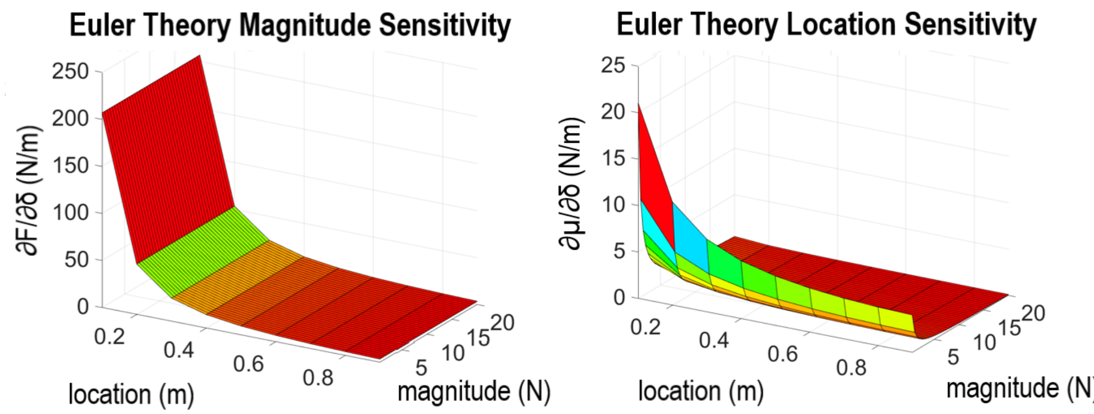


Figure 3.5: When estimating a force based off of a deflection, the certainty of its magnitude is only related to the point of application. The certainty of its location; however, is a function of both location and magnitude. For this example, a combination of load locations and magnitudes were applied to a beam of length $L = 1m$ with a flexural rigidity of $EI = 1Nm^2$.

.

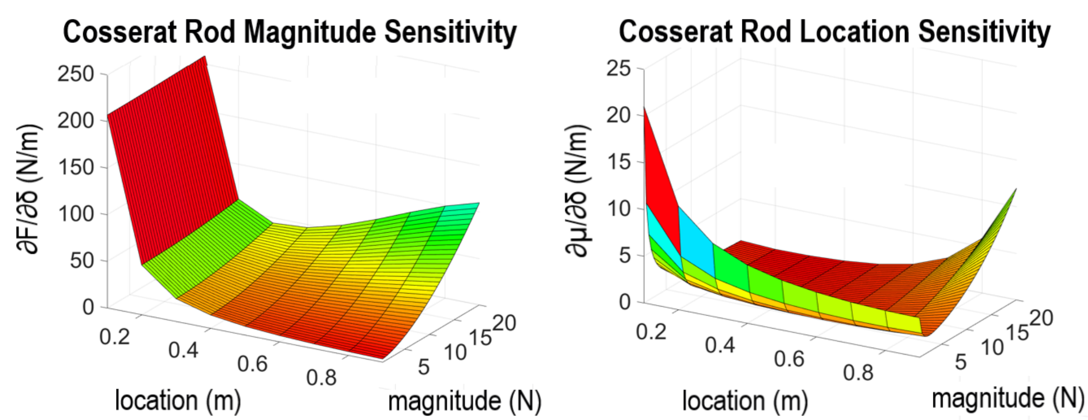


Figure 3.6: We can see that as forces deflect a cosserat rod into nonlinear deflections, the uncertainties begin to quickly increase as well. In this example, a combination of load locations and magnitudes were applied to a beam of length $L = 1m$ with a flexural rigidity of $EI = 1Nm^2$.

.

increases, eventually the uncertainty starts to rise. This is completely counter to what the linear theory predicts for both the location and magnitude estimates. Measuring more points along the rod will decrease the uncertainty but it does not remove the nonlinearity. These unpredictable effects become compounded once additional forces are introduced. This motivates calculating these uncertainties, simultaneously with the estimates, in order to avoid misinterpreting results.

3.1.3 Parameterization of Loading

The main assumption made in this work is that all forces are applied normal to the rod. This is necessary for materials with a large axial stiffness. This is justified as, axial forces along the body would only appear through friction which can be assumed to be negligible. Therefore, our distributed load is defined locally:

$$\boldsymbol{f}_{ext}(s) = \boldsymbol{R}(s) \begin{bmatrix} f_{ext}^{x,l}(s) \\ f_{ext}^{y,l}(s) \\ 0 \end{bmatrix}, \tag{3.7}$$

where the subscript ext denotes it is an external load.

In this chapter we will look at two different parameterizations of the loading. The first defines the distributed load as a truncated Fourier Series:

$$\begin{aligned} f_{ext}^{x,l}(s) &\approx a_0^x + \sum_{k=1}^{N} a_k^x \sin\frac{k\pi s}{L} + \sum_{k=1}^{N} b_k^x \cos\frac{k\pi s}{L} \\ f_{ext}^{y,l}(s) &\approx a_0^y + \sum_{k=1}^{N} a_k^y \sin\frac{k\pi s}{L} + \sum_{k=1}^{N} b_k^y \cos\frac{k\pi s}{L} \\ \boldsymbol{\gamma} &= [\boldsymbol{a}^{xT} \quad \boldsymbol{a}^{yT} \quad \boldsymbol{b}^{xT} \quad \boldsymbol{b}^{yT}]^T \end{aligned} \tag{3.8}$$

where a are the coefficients associated with the odd terms in the series and b are the coefficients associated with the even terms in the series, and the superscripts indicate the x or y component. Theoretically, this can describe any loading scenarios when enough terms are included. In practice, the series would often converge on the lower frequency terms and not quite have enough higher frequency terms to produce sharp peaks as seen in Figure 3.7 and discussed in section 3.1.1.

A second version of the parameterization is to use point loads (Dirac Delta Functions) that can slide along the length of the rod:

$$\begin{aligned} f_{ext}^{x,l}(s) &\approx \sum_{k=1}^{N} a_k^x \delta(s - \mu_k) \\ f_{ext}^{y,l}(s) &\approx \sum_{k=1}^{N} a_k^x \delta(s - \mu_k) \end{aligned} \tag{3.9}$$

where $\delta()$ is the Dirac Delta Function, a is the amplitudes of the force, and μ is the location of the force. To help with convergence of the algorithm, we can bound the load locations such that $\mu_k \in [0\ L]$. Our parameterization can then be written as:

$$\begin{aligned} \boldsymbol{\gamma} &= [\boldsymbol{a}^{xT} \quad \boldsymbol{a}^{yT} \quad \boldsymbol{\eta}^T]^T \\ \mu_k &= S(\eta_k) \in [0\ L], \end{aligned} \tag{3.10}$$

where $\boldsymbol{\eta}$ are the parameters that are mapped into the load locations. We found that this implementation can work; however, there are possible singularities that can arise. Allowing, the forces to change locations freely can allow the forces to overlap during the minimization iterations. If this happens, the Jacobian of the minimization algorithm will become singular as changing one force will have the same effect as changing another. This makes convergence dependent on initial guesses. For implementing this parameterization we assumed that we knew the number of forces but not their location or amplitudes.

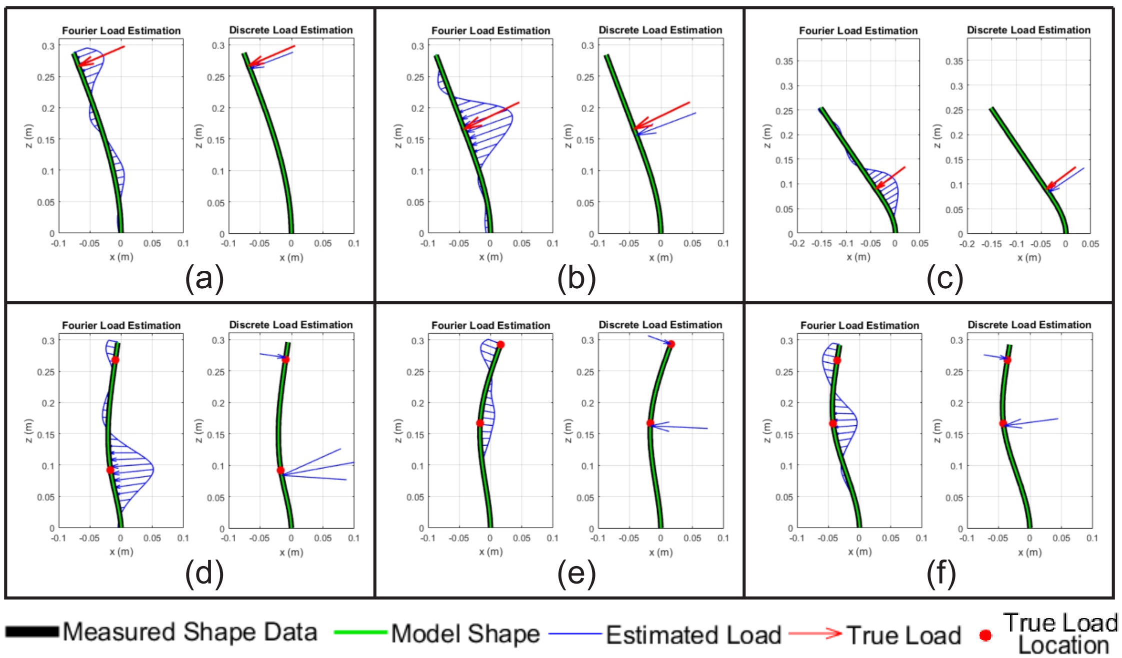


Figure 3.7: A subset of our experimental results is shown in the plots above, comparing the performance of our optimization approach with the two different force representations. In the single force cases on the top row, the estimated loads are compared to the ground truth load as measured by the base mounted sensor. Sub-figures (a), (b), and (c) correspond to tests 18, 14, and 12 in Table 3.2. On the second row, the ground-truth force locations are compared to the estimated loadings. Note that the magnitudes of the distributed forces and point forces on the plot should not be directly compared since they have different physical units and are scaled by different arbitrary amounts for visibility. Sub-figures (d), (e), and (f) correspond to tests 4, 5, and 6 in Table 3.3.

3.2 Rod Experiments

To test our proposed force estimation framework we performed a series of experiments which compare the performance of the Fourier series representation with the set of point loads representation. We consider planar experimental cases, but the full 3D Cosserat model and optimization framework is used to fit the planar data. For implementation of the algorithm, we used matlab's "fmincon" numerical solver. The bounding function used to determine the load locations was chosen to be:

$$\mu_k = S(\eta_k) = \frac{L}{2} + \frac{L}{2} sin(\eta_k). \tag{3.11}$$

For registration error, we chose to let the initial pose of the rod be an input to the guess vector. We also chose to let the arc length at which a data point is being measured be an input to the guess vector, with the initial guess being provided by the shape sensing method. This gives us:

$$U = [\boldsymbol{p}_0^T \quad \boldsymbol{\theta}_0^T \quad s_0 ... s_{N_p}]^T, \tag{3.12}$$

where $\theta_0 = [\theta_x \ \theta_y \ \theta_z]$ contains the Euler angles which define the base orientation $\boldsymbol{R}_0$.

3.2.1 Setup and Procedure

The experimental conditions included a set of 18 cases where a single point force was applied at various unknown locations along a rod and a set of 6 cases where two point forces were applied at various unknown locations. An appropriate number of terms for the Fourier series (for adequate resolution of point loads without over-fitting) was determined to be 21 using two cases from the data set. Point loads are the most difficult cases for the Fourier series to approximate. In contrast, the sliding Dirac deltas should be able to determine the appropriate number of forces and represent these cases exactly. We will see that both methods perform well in terms of attaining near-zero shape error and determining the location of high-force regions and load magnitudes.

The setup, shown in Figure 3.8, consisted of an elastic metal rod, a pegboard, an ATI 6-DoF force/torque sensor Nano 43 SI-36-0.5 (used only for validation), and an HD 1920x1080 Logitech digital camera held by a Noga Articulated Holder MG71003. We performed experiments on two different rods (of lengths 0.2 m and 0.3 m), both with a diameter of 1.4 mm ($I = J/2 = 1.89\text{x}10^{-13}\ m^4$) and made from ASTM A228 spring steel, which has a Young's modulus of 207 GPa and shear modulus of 79 GPa. The rods were painted green for enhanced visibility, mounted to the force sensor, and subjected to point loads by small plastic cylinders.

In each experiment, a set of discrete shape data points $\hat{p}_i$ is obtained by processing a single camera image (shown in Figure 3.7). We used color thresholding to isolate the pixels that correspond to the rod shape and obtain an ordered list of shape data points and arc lengths by employing a sliding window of pixels starting at the base of the rod and iteratively propagating the window forward along the approximate tangent vector. At each step we calculate the centroid of the rod pixels within the current window and then slide the window by a fixed distance in the the direction of the centroid until the end of the rod is reached. For the 30 cm and 20 cm rod experiments, $N_p = 134$ and $N_p = 98$.

In the first set of experiments, a single peg was used to create loading conditions at varying contact locations along each rod, which are listed in Table 3.2, some of which are shown in the top row of Figure 3.7. In a second set of experiments, we used two pegs to deform the rods by applying a set of two loads at various locations. Three different loading conditions were analyzed for each rod, which are listed in Table 3.3. The three two-peg experiments for the 30 cm rod are shown in the second row of Figure 3.7.

To measure the ground-truth arc-length location of the applied forces, the undeformed rod was marked with ticks, 1mm apart, using a ruler, and the locations were identified manually. In the single-load experiments we obtained the ground-truth applied force vector directly from the base mounted force sensor (since for a single peg, the measured base force should equal the applied force from the peg).

Table 3.2: Summary of Single Load Experimental Results

Rod Length	Test Number	RMS Shape Error (mm)		Arc Length Location (mm)			Magnitude (N)		
		Fourier Series	Sliding Dirac Delta	Ground Truth	Fourier Series	Sliding Dirac Delta	Ground Truth	Fourier Series	Sliding Dirac Delta
200 mm	1	0.085	0.086	67	62	65	4.17	5.44	4.44
	2	0.067	0.067	92	84	90	1.71	2.10	1.82
	3	0.093	0.091	99	90	94	4.29	5.39	4.60
	4	0.071	0.069	117	123	115	0.89	0.98	0.93
	5	0.079	0.075	123	127	118	2.32	2.46	2.38
	6	0.099	0.097	133	134	130	3.15	3.17	3.12
	7	0.082	0.079	167	179	159	0.32	0.36	0.36
	8	0.063	0.066	171	165	165	0.88	1.00	0.89
	9	0.086	0.082	180	186	177	1.34	1.13	1.26
300 mm	10	0.075	0.068	68	66	64	4.96	5.06	4.52
	11	0.124	0.076	93	91	83	2.24	2.52	2.74
	12	0.151	0.136	100	93	93	4.43	5.71	5.17
	13	0.130	0.078	169	161	156	0.40	0.46	0.45
	14	0.112	0.069	173	167	163	0.99	0.96	1.02
	15	0.148	0.118	182	185	179	1.35	1.28	1.29
	16	0.166	0.111	270	279	264	0.11	0.14	0.10
	17	0.125	0.082	272	258	270	0.27	0.19	0.24
	18	0.127	0.086	278	277	273	0.40	0.49	0.37

Table 3.3: Summary of Double Load Experimental Results

Rod Length	Test Number	RMS Shape Error (mm)		Arc Length Locations (mm, mm)			Base Force Magnitude (N)		
		Fourier Series	Sliding Dirac Deltas	Ground Truth	Fourier Series	Sliding Dirac Deltas	Ground Truth	Fourier Series	Sliding Dirac Deltas
200 mm	1	0.157	0.073	67, 194	72, 175	62, 198	8.60	7.87	9.61
	2	0.206	0.072	92, 168	87, 178	92, 166	5.16	6.10	5.18
	3	0.100	0.248	108, 193	125, 190	117, 197	3.62	3.18	3.53
300 mm	4	0.187	0.087	94, 272	90, 289	87, 276	4.01	4.18	4.48
	5	0.147	0.084	169, 300	179, 284	166, 300	1.89	2.21	1.85
	6	0.141	0.084	173, 275	175, 273	171, 277	2.67	2.21	2.58

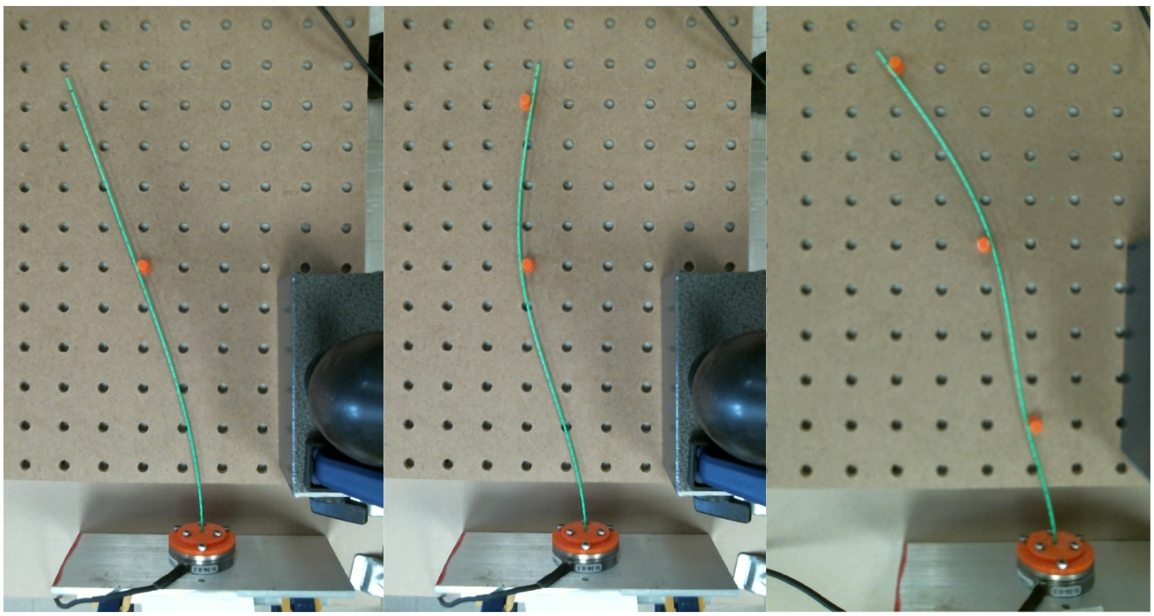


Figure 3.8: The experimental setup consisted of a painted green rod, an ATI 6-DoF force/torque sensor, a mounted camera, and pegboard. The images shown above were used by the estimation algorithms for the single load test 14 (in Table 3.2), the double load test 6 (in Table 3.3), and a three load test.

The sensor is used only for validation and does not provide any information to the estimation algorithms. In the two-load case, the individual ground-truth loads are not directly measurable using only the base measured force, but we can still compare the measured force at the base to the resultant of the estimated force distribution, computed by integrating it over the entire length (where the integral of a Dirac delta function is one by definition, so integrating merely sums the discrete loads).

3.2.2 Results

For the point-force representation, the ground-truth force vector and location can be directly compared to the estimated ones, as shown in Figure 3.7. For the Fourier series representation, we compare the ground-truth load location to the location of the highest peak on the force distribution, and we compare the ground-truth load magnitude to the integral of the Fourier force distribution over the length. Note that the size of the distributed forces and point forces shown in Figure 3.7 should not be directly compared since they have different units and are scaled by different amounts for visibility.

Table 3.2 lists the RMS shape error of both methods, the arc-length location, and the load magnitude for all the single force cases, comparing estimated values to the true applied values in each case. Over the 18 single load cases, the average error in load location was 6 mm (2-3% of total rod length) for the Fourier series approximation and 5 mm for the Dirac delta approximation. The average error in load magnitude was 0.30 N and 0.16 N, respectively. The 6 two loads cases are listed in Table 3.3. The average error in load location was 9 mm for the Fourier series and 4 mm for the for the Dirac delta. The average base force magnitude error was 0.51 N and 0.29 N, respectively.

The results indicate that the Dirac delta approximation provided an overall more accurate representation of the loads. This was expected as the Fourier series is fundamentally limited when describing point loads. We chose this set of experiments because we believe that the most important application of the distributed force estimation problem is to identify the locations and magnitudes of highly concentrated forces. Since this is also the most difficult case for the Fourier series to represent, these experiments are a good stress test for finding the limits of its ability.

3.2.3 Further Examples: Complex and 3D Load Cases

In addition to our main experimental dataset, we further demonstrate the generality of the approach by including two ad hoc experimental examples of more complex cases. First, Figure 3.9 shows an experiment with 3 point loads that create multiple inflection points. The results are consistent with our main dataset indicating that more complex loadings can be handled. Second, we show the results from a much more challenging 3D loading experiment in Figure 3.10. We used a 3D printed rod with a 4mm square cross section made of thermoplastic polyurethane. To achieve a 3D, non-planar deformed shape, the rod was printed into a precurved “U” shape, and gravity was used to apply an out of plane distributed load. In this large-deflection scenario, torsion is a significant contributor to the deformed shape. We used orthogonal camera views to determine the 3D shape, and applied our Fourier estimation method with $N = 21$ terms. Despite the fact that a global gravity load violates our modeling assumption, that the loading is orthogonal to the rod tangent vector, the results shown in Figure 3.10 display an estimated load of approximately the right magnitude and direction. The RMS shape error over the length is 2.64 mm (whereas the width of the rod is 4 mm, and the total length is 190 mm). This example indicates that the inverse problem is perhaps more ill-conditioned in 3D cases with significant torsion, but the method still gives useful information about loading conditions consistent with the observed shape. This work was published in the IEEE 2019 ICRA Proceedings [41]

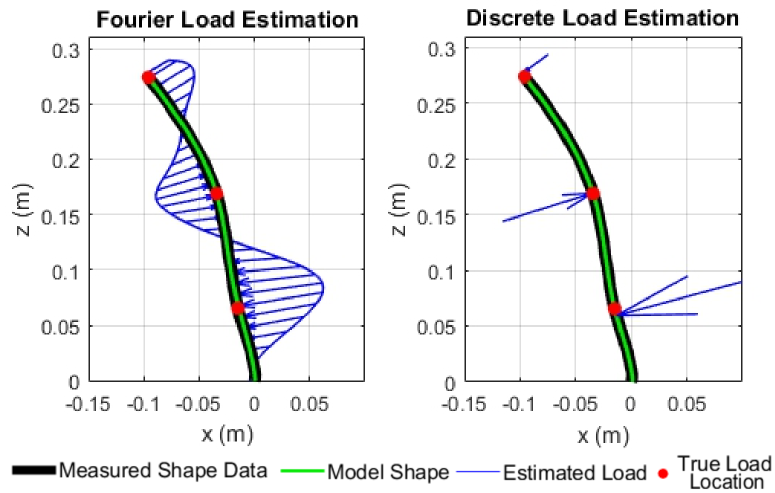


Figure 3.9: We include this additional ad hoc example of a three-load case to further demonstrate the generality of the approach. Even with alternating curvatures, both distributed load approximations are capable of achieving near-zero shape error, and the estimated distributions have peaks near the true contact points.

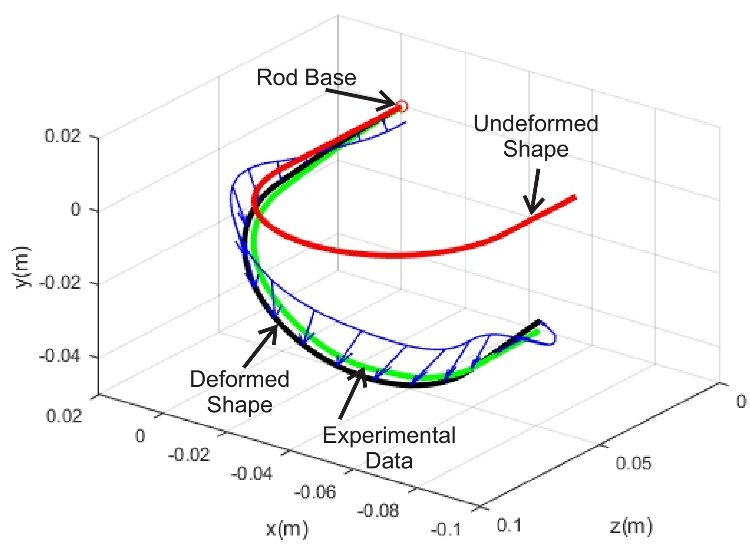


Figure 3.10: In theory, the algorithms presented are capable of force estimation for 3D rod shapes and distributed loading. Here we present a test example where a soft "U" shaped rod structure is deformed by gravity.

Chapter 4

Estimating Body Loads Along Continuum Robots

In this chapter, we will extend the current method for force estimation to continuum robots. In essence, we can directly apply the method from Chapter 3 by replacing the Cosserat Rod Model with a kinetostatic model of a desired robot. This can slightly complicate the problem as we now have actuators applying forces to the robot as well. These need to either be predicted from a model or measured directly. We will provide an example of how this problem can be formulated and applied to a multi-segment tendon driven continuum robot. Finally, experiments will be preformed on a single tendon robot and a multi-segment two tendon robot. The notation for this chapter is summarized in table 4.1.

4.1 Adapting the Estimation Method to Robots

Forward kinetostatic models are commonly formulated as boundary value problems where the distal boundary conditions are enforced via a shooting method which applies an optimization routine to the associated initial value problem [46]. In this context, it is efficient to combine the model optimization loop with the estimation

Table 4.1: Notation for Continuum Robot Force Estimation

Symbol	Definition
$\boldsymbol{p}$	Position of a point along the rod
$\boldsymbol{R}$	Orientation of a point along the rod
$\boldsymbol{n}$	Internal force
$\boldsymbol{m}$	Internal moment
$\boldsymbol{f}$	Distributed force
$\boldsymbol{c}$	Vector of boundary conditions
$\boldsymbol{x}$	Vector of unknown inputs
$\tilde{\boldsymbol{z}}$	Vector of terms to minimize
$\boldsymbol{W}$	Diagonal matrix of weights
$\boldsymbol{\alpha}$	Vector of unknown model inputs
$\boldsymbol{\gamma}$	Vector of parameters that define $\boldsymbol{f}(s)$
$\boldsymbol{\zeta}$	Twist vector for data registration
μ	Location of point force
$\boldsymbol{a}$	Amplitude of a gaussian function
σ	Steepness of gaussian function
N_p	Number of data points
$\hat{\cdot}$	Model estimated value
Subscript ext	External load
Superscript l	Local value
Superscript x or y	x or y component of vector

optimization loop to form a single problem that solves the boundary conditions of the kinetostatic model simultaneously with the force estimation and registration problems. Weighted least squares provides a natural way to satisfy all these simultaneously:

$$\underset{\mathbf{x}}{\text{minimize}} \quad \tilde{\boldsymbol{z}}^\top \boldsymbol{W} \tilde{\boldsymbol{z}}$$

$$\tilde{\boldsymbol{z}}(\boldsymbol{x}) = \begin{bmatrix} \hat{\boldsymbol{p}}_0(\boldsymbol{x}) - \boldsymbol{p}_0 \\ \vdots \\ \hat{\boldsymbol{p}}_{N_p}(\boldsymbol{x}) - \boldsymbol{p}_{N_p} \\ \\ \boldsymbol{c}(\boldsymbol{x}) \end{bmatrix}, \tag{4.1}$$

$$\boldsymbol{x} = \begin{bmatrix} \boldsymbol{\alpha}^\top & \boldsymbol{\gamma}^\top & \boldsymbol{\zeta}^\top \end{bmatrix}^\top,$$

where $\boldsymbol{c}(\boldsymbol{x})$ is the vector of distal boundary condition residuals, $\boldsymbol{W}$ is a chosen diagonal matrix of weights, $\boldsymbol{\zeta}$ is the twist vector that parameterizes the registration transformation, and $\boldsymbol{\alpha}$ is the vector of unknown initial conditions to be solved. To evaluate this problem, we will be using an Extended Kalaman Filter where the data is invariant in time:

$$\begin{aligned} \bar{\boldsymbol{\Sigma}}_i &= \boldsymbol{\Sigma}_i + \boldsymbol{Q} \\ \boldsymbol{K}_i &= \bar{\boldsymbol{\Sigma}}_{i-1} \boldsymbol{H}_i^\top (\boldsymbol{H}_i \bar{\boldsymbol{\Sigma}}_i \boldsymbol{H}_i^\top + \boldsymbol{W}^{-1})^{-1} \\ \boldsymbol{x}_{i+1} &= \boldsymbol{x}_i + \boldsymbol{K}_i \tilde{\boldsymbol{z}} \\ \boldsymbol{\Sigma}_{i+1} &= (\boldsymbol{I} - \boldsymbol{K}_i \boldsymbol{H}_i) \bar{\boldsymbol{\Sigma}}_i \end{aligned} \tag{4.2}$$

where $\boldsymbol{I}$ is the identity matrix, $\boldsymbol{\Sigma}$ is the co-variance matrix of $\boldsymbol{x}$, $\boldsymbol{Q}$ is conventionally the matrix of expected process noise, and $\boldsymbol{H}$ is the Jacobian of $\boldsymbol{z}$ with respect to $\boldsymbol{x}$. In the context of solving a nonlinear least-squares optimization problem, $\boldsymbol{Q}$ represents a tunable damping matrix, similar to a Levenberg-Marquardt type of damping. In

[43], they made note that using an Unscented Kalman Filter was more efficient and faster than using a Levenberg-Marquardt algorithm.

4.1.1 Gaussian Load Parameterization

For this chapter, we will be using Gaussian Distribution Functions. This choice was made because it is easy to implement and, by altering the steepness of the function, we can approximate a point load or a distributed load. Each Gaussian function is centered and fixed at a specific arc length μ_i with an amplitude in both the local x and y direction. The full parameterization can then be given as:

$$\begin{aligned} f_{ext}^{x,l} &= \sum_{k=0}^{N_l} a_k^x e^{-\sigma_k (s-\mu_i)^2} \\ f_{ext}^{y,l} &= \sum_{k=0}^{N_l} a_k^y e^{-\sigma_k (s-\mu_i)^2} \\ \mu_k &= S(\eta_k) \in [0 \;\; L] \\ \boldsymbol{\gamma} &= [\boldsymbol{a}^{xT} \quad \boldsymbol{a}^{yT} \quad \boldsymbol{\eta}^T \quad \boldsymbol{\sigma}]^T, \end{aligned} \tag{4.3}$$

where N_l is the total number of load terms, a_i^x and a_i^y are the amplitudes of the components for each term, and σ is the variable that determines how steep the peak of the Gaussian function is. We will be opting to use a sigmoid function in order to bound the locations of the loading as these are monotonic functions with well defined derivatives:

$$\mu_k = S(\eta_k) = \frac{\eta_k}{\sqrt{\epsilon + \eta_k^2}}, \tag{4.4}$$

where ϵ determines the slope of the sigmoid curve.

There is an added benefit of using Gaussian functions due to the ease of integration. Integrating the distributed load curve can give an approximation of the total load applied:

$$\|F_i\| = \sqrt{\frac{\pi}{\sigma_i}} \|a_i\| \tag{4.5}$$

4.1.2 Tendon Driven Continuum Robot Model

For validation of this method and an example of its use, we will applying it to a tendon driven continuum robot. We implemented the model presented in [48]. A basic tendon driven robot can be treated as a cosserat rod experiencing external loading where the tendons exert a distributed wrench along the backbone:

$$\begin{aligned} \boldsymbol{p}' &= \mathbf{R}\mathbf{v} \\ \mathbf{R}' &= \mathbf{R}\mathbf{u}^{\wedge} \\ \boldsymbol{n}' &= -(\boldsymbol{f}_{ext} + \boldsymbol{f}_t + \boldsymbol{f}_g) \\ \boldsymbol{m}' &= -\boldsymbol{p}' \times \mathbf{n} - \boldsymbol{l}_t, \end{aligned} \tag{4.6}$$

where subscript t indicates a load from a tendon and $\boldsymbol{f}_g$ is the gravitational load. It is worth noting, however, that $\boldsymbol{f}_t$ and $\boldsymbol{l}_t$ are nonlinear functions of the state and introduce additional derivative terms into the problem which is further elaborated on in [48].

Given the tension in the tendons, this system can be solved as a boundary value problem using a shooting method. Distal boundary conditions are derived from a static balance at the tip:

$$\boldsymbol{c}(\boldsymbol{x}) = \begin{bmatrix} \boldsymbol{n}(L) - \boldsymbol{F}_t \\ \boldsymbol{m}(L) - \boldsymbol{M}_t \end{bmatrix}, \tag{4.7}$$

where $\boldsymbol{F}$ and $\boldsymbol{M}$ are the tip forces and moments applied by the tendons. The unknown initial conditions are the base wrench:

$$\boldsymbol{\alpha} = \begin{bmatrix} \boldsymbol{n}(0) \\ \boldsymbol{m}(0) \end{bmatrix}. \tag{4.8}$$

4.2 Single DoF Robot Experiments

For the first set experiments, a robot with a single tendon was constructed, as seen in Figure 4.1. It consisted of a 1.4 mm diameter backbone, nine PLA spacer disks, and Kevlar thread. The length of the robot was 40cm and the disks were evenly spaced. The tendon terminates at the final disk and is connected to a load cell. This load cell is affixed to a linear actuator. The tendon runs parallel to the back bone at a distance of 7 mm. The backbone is made of ASTM A228 Spring steel, which has a Young's modulus of 207 GPa and shear modulus of 79 GPa. Finally, optical markers were attached to the spacer disks so we could record their 3D positions. This gives us $N_p = 9$ data points. The robot was oriented such that the tendon would move the end effector in the plane of gravity.

4.2.1 Estimating Gravitational Load

For the algorithm to present good results, it is not necessary to know the estimation of gravitational load before hand; however, the effects of $\boldsymbol{f}_g$ cannot be parsed from that of the other loads, $\boldsymbol{f}_{ext}$. Gravity also acts in the global frame, thus it cannot be parameterized in the same way as the other loads. For our robot, we assumed that the effect of gravity can be approximated as an evenly distributed load:

$$\boldsymbol{f}_g = \rho \boldsymbol{a}_g \tag{4.9}$$

where ρ is the mass per length, and $\boldsymbol{a}_g$ is the gravitational acceleration vector. To calibrate ρ, we can simply include it in the vector of unknowns, $\gamma = \rho$, and solve the force estimation problem from equation 4.1 assuming the only external load is gravity. We actuated the robot into seven unique poses and recorded the tendon tensions and maker frames. Using a Levenberg-Marquardt algorithm, we fit the shape for all 7 tests simultaneously. This gave us a final result of $\rho = 0.144\frac{kg}{m}$.

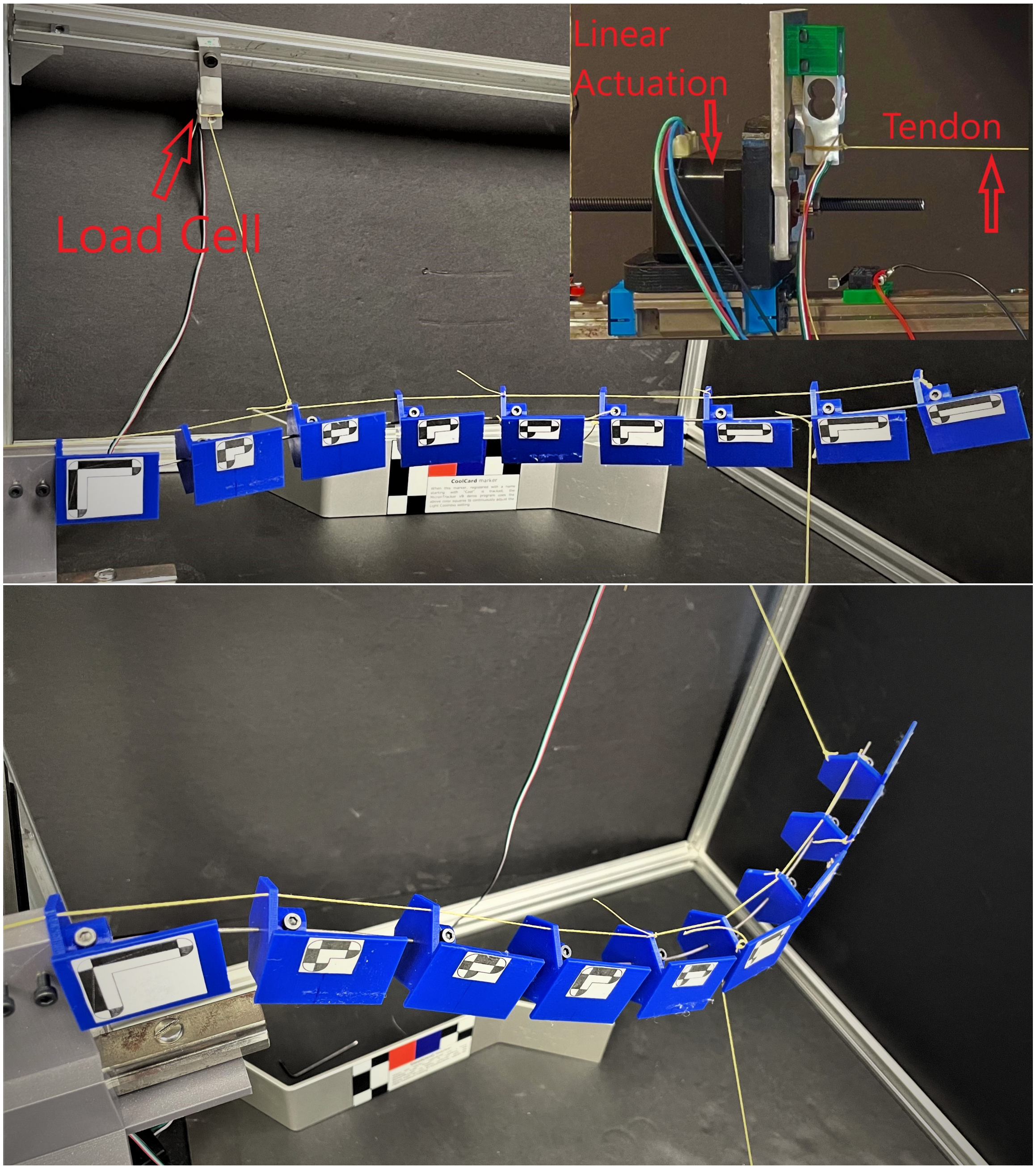


Figure 4.1: The robot was constructed using a spring steal rod and 9 PLA spacer disks. To measure tension in the tendon, a loadcell was placed at the actuator. To apply forces to the backbone, load cells were tied to the spacer disks and oriented such that they were normal.

4.2.2 Procedure and Results

In the first set of experiments, a load cell was used to create a single point force both in and out of plane. The loading was placed at varying contact locations and with the robot actuated into various poses. The cables were oriented such that they were normal to the rods backbone. An optical micron tracker was used to capture the shape data. The results are listed in Table 4.2. For the estimation, the loading was parameterized as a single gaussian curve.

For the single loading cases, the mean magnitude error was 0.32 N and the mean location error was 2.7 cm (7% of total robot length). Even with the extra loading from gravitational forces and actuation, the force magnitude was of same order of accuracy as the experiments performed on a cantilevered rod in Chapter 3 (0.29 N). The location error was slightly larger but only increased by a few percent of the total length (2-3%). Comparing the planar and non planar cases, the difference in accuracy is negligible.

In the second set of experiments, we used two load cells to apply two point forces both in and out of plane, such as in Figure 4.1. As before, we tested loads at various locations and different robot poses. These results are given in Table 4.3. For the estimation in these experiments, the loading was parameterized as two gaussian curves. Even with the robot loaded into large nonlinear deflections, the accuracy remained consistent. There was a mean error of 0.51 N and 2.64 cm for the magnitude and location, respectively.

Table 4.2: Single Load Cases for 1 DoF Tendon Robot

In Plane				
Actuation (cm)	Load Magnitude		Load Location	
	Load Cell Reading (N)	Estimated (N)	True Location (cm)	Estimated (cm)
-11.5	0.06	0.04	30	25
	1.2	0.7	10	10
-5	3.9	4.8	10	7
	0.5	0.4	30	28
5	0.7	0.6	10	12
	2.5	2.0	10	10
Out of Plane				
-11.5	0.6	0.7	30	29
	1.4	2.3	10	8.5
0	4.2	4.2	10	7.7
	0.7	0.8	30	27

Table 4.3: Double Load Cases for 1 DoF Tendon Robot

In Plane				
Actuation (cm)	Load Magnitude		Load Location	
	Load Cell Reading (N)	Estimated (N)	True Location (cm)	Estimated (cm)
0	3.3 1.9	3.9 1.4	20 30	15 30
	5.0 2.9	5.4 2.2	30 20	29 15
5	2.2 1.6	3.8 1.1	10 30	12 27
	9.6 2.5	8.6 3.2	10 30	10 26
Out of Plane				
0	3.2 1.9	3.8 2.4	25 40	24 36
	1.52 0.72	1.86 0.92	25 40	12 23
5	5.6 0.9	4.2 0.4	10 35	9 28
14.5	0.8 0.4	0.9 0.5	40 25	39 22

4.3 Two DoF Robot Experiments

To cap off this dissertation, we finally constructed a multi-segment robot with two tendons as seen in Figure 4.2. Similar to the one before, we used a spring steel rod backbone and nine spacer disks. This robot was significantly larger though, with a length of 55 cm and a calibrated density $\rho = 0.252\frac{kg}{m}$. The tendons run parallel to the backbone at an offset of 2.5 cm. The tendons were placed on orthogonal sides such that actuation of both moves the robot out of plane. The tendon that actuates in the plane of gravity terminates at end effector and the second tendon terminates at the disk halfway up the robot.

For the experiments, eight poses were tested and the results are summarized in Table 4.4. We did six single load cases and two double load cases. The overall results were similar to that of the 1 DoF experiments with there being a 0.65 N error and 3.98 cm (7.2 %) error in magnitude and location, respectively. Even though expected, this shows that the increased complexity of the loading from the second tendon and large out of plane poses do not limit the estimations. In Figure 4.3, we provide a showcase of four examples comparing the model estimation and actual loading.

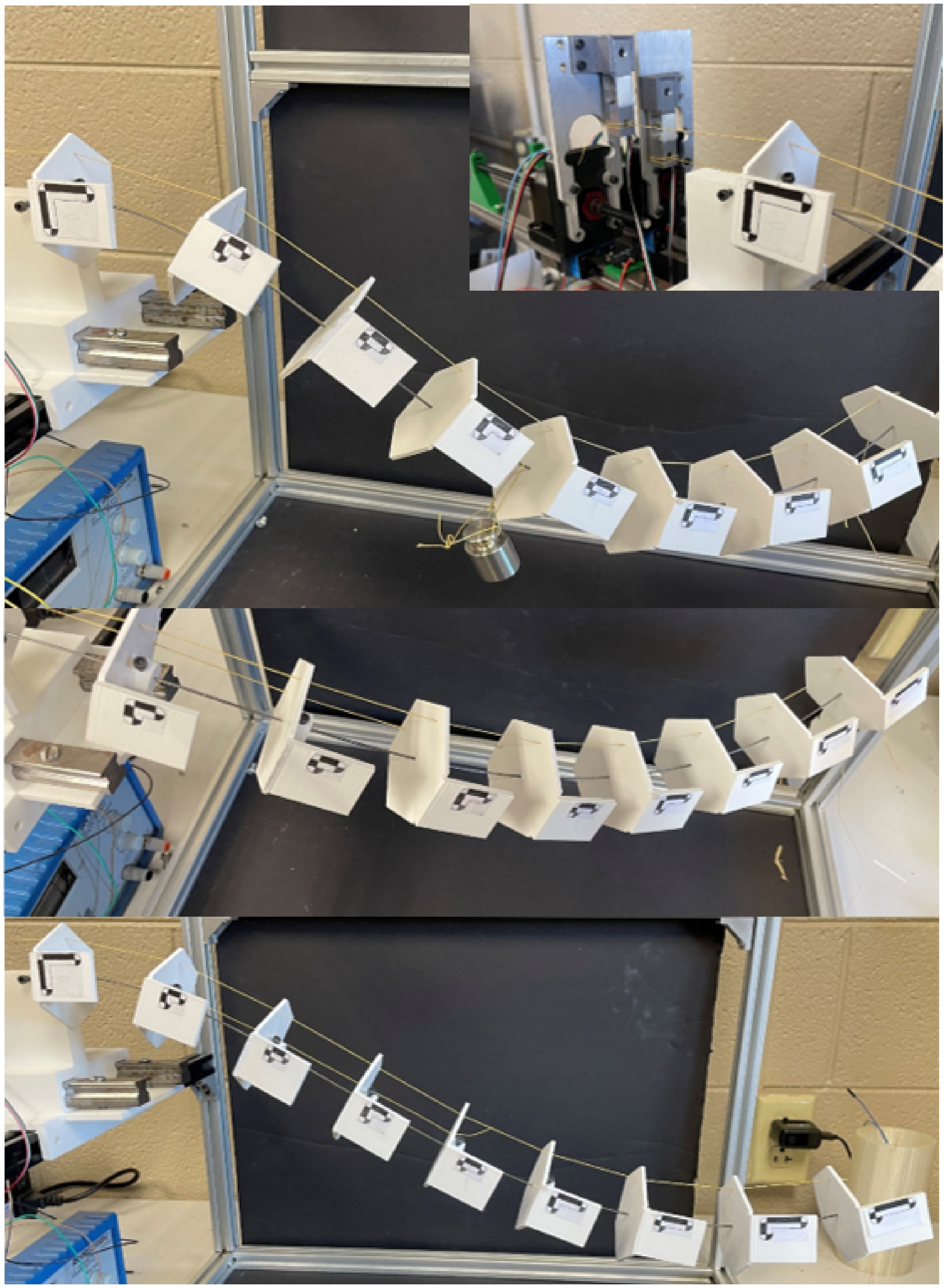


Figure 4.2: A second robot with two tendons was constructed. The tendons were placed orthogonal to eachother such that actuating both pulls the robot out of plane. One tendon terminates at the end effector and the other tendon terminates at spacer disk halfway up the robot.

Table 4.4: 2 DoF Tendon Robot Experiments

Actuation		Load Magnitude		Load Location	
DoF 1 (cm)	DoF 2 (cm)	Load Cell Reading (N)	Estimated (N)	True Location (cm)	Estimated (cm)
25	0	1.17	1.44	14	18.9
15	5	1.39	1.58	28	20.6
30	0	5.21	3.71	21	19.8
20	20	1.59	2.68	21	23.5
30	5	5.57	3.62	21	20.0
20	10	1.01	1.99	21	21.6
30	5	2.87	3.02	21	19.4
		0.49	0.60	52	46.0
10	0	1.96	2.11	35	22.7
		1.09	0.93	55	52.7

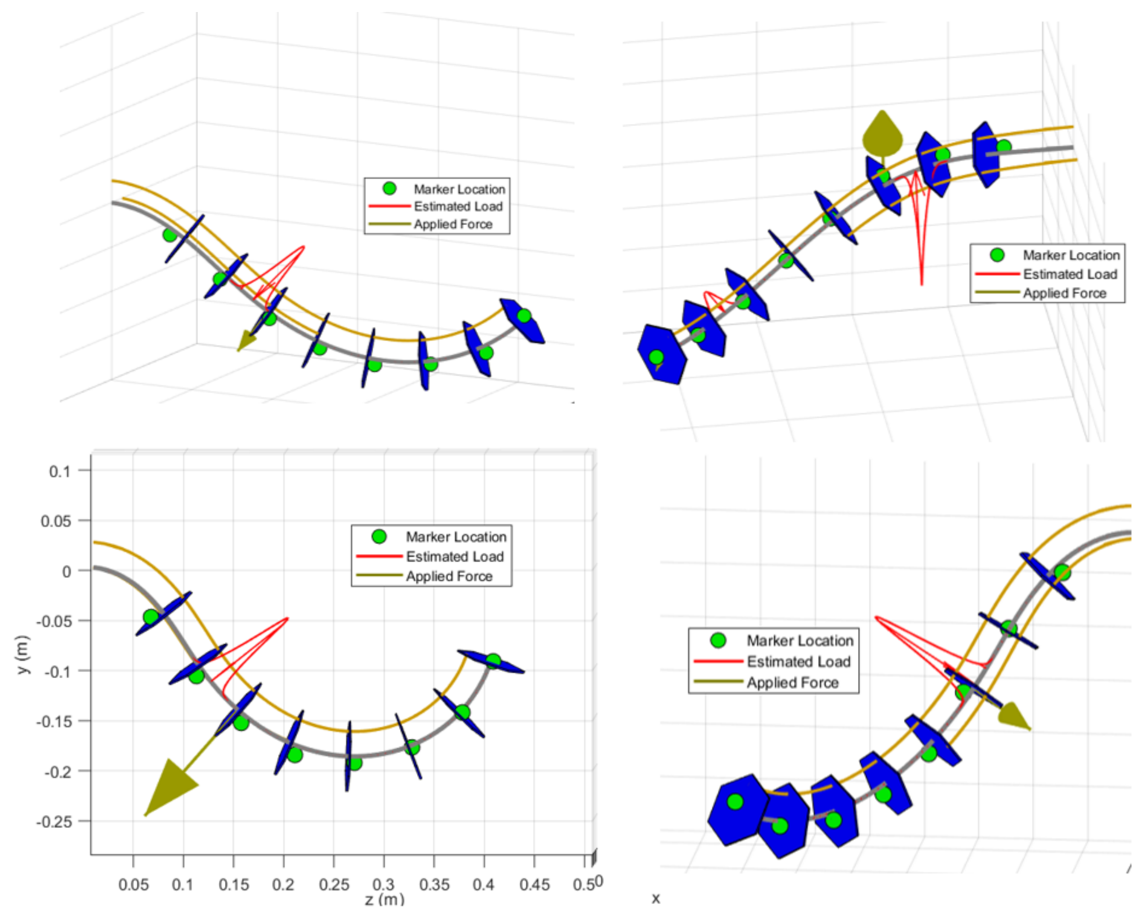


Figure 4.3: Using a 2 DoF robot, experiments were preformed to validate non planar loading and actuation. Above are 4 example cases from the experiment set. Using shape data and known tendon tensions, the algorithm is able to find appropriate forces necessary to recreate that shape.

Chapter 5

Conclusions

The goal of this dissertation was to improve the control of continuum robots by using existing kinetostatic models in the field. With a model based approach, it was already known that we could indirectly sense tip forces using other forms of data such as force measurements at the actuators or tip displacement. The first thrust of this work was to use this method to control the apparent stiffness of an end effector. We provided an integral based control law, that was capable of dictating a full 3x3 desired stiffness matrix for a continuum robot interacting with both compliant or rigid environments.

In the second thrust, we tried to extend this idea for tip force estimation to more general loading scenarios. We found that using shape data, we can use a model to predict a loading scenario that can recreate the shape. Due to uncertainty propagation, we know that a predicted load distribution will not be exact and there are actually large families of loads that can recreate the same shape within a small window of shape error. This uncertainty is effected by loading and geometry, and can change during actuation. That said, by making good choices for the parameterization of the loading, we can mitigate these problems. As an example, in this work we assumed all loads would be applied normal to the back bone and primarily focused on estimating point forces. We were able to validate this method on two separate tendon robots with similar levels of accuracy between them. There was a total magnitude error of 0.58 N and location error of 7% (of the total robot length).

Moving beyond this work, understanding the uncertainty will become more necessary as the estimation methods are applied to control algorithms. We will never be able to guarantee we know the exact load distribution; however, it could be possible to gather a probability distribution that could be equally useful. This could help reduce the number of assumptions that have to be made about the loading overall. As the dissertation comes to a close, it is still apparent how many questions have been left unanswered. It is my genuine hope that the community continues to show love for this problem and push onward to improving the utility of these amazing robots.

Bibliography

[1] G. Robinson and J. Davies, "Continuum robots - a state of the art," vol. 4. IEEE, pp. 2849–2854. 1

[2] J. Burgner-Kahrs, D. C. Rucker, and H. Choset, "Continuum robots for medical applications: A survey," *IEEE Transactions on Robotics*, vol. 31, pp. 1261–1280, 12. 1

[3] G. Chirikjian and J. Burdick, "Hyper-redundant robot mechanisms and their applications." IEEE, pp. 185–190. 1

[4] F. Khan, R. J. Roesthuis, and S. Misra, "Force sensing in continuum manipulators using fiber bragg grating sensors." IEEE, 9, pp. 2531–2536. 2, 9, 12, 31

[5] M. Khoshnam, A. C. Skanes, and R. V. Patel, "Modeling and estimation of tip contact force for steerable ablation catheters," *IEEE Transactions on Biomedical Engineering*, vol. 62, pp. 1404–1415, 5. 2, 6, 9, 11

[6] C. M. Heunis, V. Belfiore, M. Vendittelli, and S. Misra, "Reconstructing endovascular catheter interaction forces in 3d using multicore optical shape sensors." Institute of Electrical and Electronics Engineers Inc., 11 2019, pp. 5419–5425. 2, 9, 12

[7] R. Xu, A. Yurkewich, and R. V. Patel, "Curvature, torsion, and force sensing in continuum robots using helically wrapped fbg sensors," *IEEE Robotics and Automation Letters*, vol. 1, pp. 1052–1059, 7. 2, 7, 9, 11

[8] J. Back, L. Lindenroth, K. Rhode, and H. Liu, "Three dimensional force estimation for steerable catheters through bi-point tracking," *Sensors and Actuators, A: Physical*, vol. 279, pp. 404–415, 8 2018. 2

[9] L. Lindenroth and C. Duriez, "Intrinsic force sensing capabilities in compliant robots comprising hydraulic actuation development of a robotic trans-esophageal ultrasound system view project robotic ultrasound systems for intelligent fetal imaging and diagnosis view project." 2, 9, 12

[10] H. Yuan, P. W. Y. Chiu, and Z. Li, "Shape-reconstruction-based force sensing method for continuum surgical robots with large deformation," *IEEE Robotics and Automation Letters*, vol. 2, pp. 1972–1979, 10. 2, 6, 9, 12

[11] M. Razban, J. Dargahi, and B. Boulet, "A sensor-less catheter contact force estimation approach in endovascular intervention procedures." Institute of Electrical and Electronics Engineers Inc., 12 2018, pp. 2100–2106. 2, 9, 11

[12] C. D. Santina, R. L. Truby, and D. Rus, "Data–driven disturbance observers for estimating external forces on soft robots," *IEEE Robotics and Automation Letters*, vol. 5, pp. 5717–5724, 10. 2, 9, 10

[13] A. Shiva, S. M. Sadati, Y. Noh, J. Fraś, A. Ataka, H. Würdemann, H. Hauser, I. D. Walker, T. Nanayakkara, and K. Althoefer, "Elasticity versus hyperelasticity considerations in quasistatic modeling of a soft finger-like robotic appendage for real-time position and force estimation," *Soft Robotics*, vol. 6, pp. 228–249, 4. 2, 9, 12

[14] D. C. Rucker and R. J. Webster, "Deflection-based force sensing for continuum robots: A probabilistic approach." IEEE, 9, pp. 3764–3769. 2, 9, 12

[15] J. Back, T. Manwell, R. Karim, K. Rhode, K. Althoefer, and H. Liu, "Catheter contact force estimation from shape detection using a real-time cosserat rod model," vol. 2015-Decem. Institute of Electrical and Electronics Engineers Inc., 12 2015, pp. 2037–2042. 2

[16] C. B. Black, J. Till, and D. C. Rucker, "Parallel continuum robots: Modeling, analysis, and actuation-based force sensing," *IEEE Transactions on Robotics*, vol. 34, pp. 29–47, 2. 2, 9, 12, 17

[17] V. Aloi, C. Black, and C. Rucker, "Stiffness control of parallel continuum robots," vol. 1. American Society of Mechanical Engineers (ASME), 11 2018. 2, 17

[18] S. Antman, *Nonlinear Problems of Elasticity*, 2nd ed. Springer-Verlag New York Inc., 9 2005. 2

[19] K. Reed, A. Okamura, and N. Cowan, "Modeling and control of needles with torsional friction," *IEEE Transactions on Biomedical Engineering*, vol. 56, pp. 2905–2916, 12. 3

[20] R. M. Murray, Z. Li, and S. Sastry, *A mathematical introduction to robotic manipulation*. CRC Press, 1994. 3

[21] Z. Zhang, J. Dequidt, and C. Duriez, "Vision-based sensing of external forces acting on soft robots using finite element method," *IEEE Robotics and Automation Letters*, vol. 2018. 6, 9, 10

[22] ——, "Vision-based sensing of external forces acting on soft robots using finite element method," *IEEE Robotics and Automation Letters*, vol. 3, pp. 1529–1536, 7 2018. 6, 9

[23] Z. Zhang, A. Petit, J. Dequidt, and C. Duriez, "Calibration and external force sensing for soft robots using an rgb-d camera," *IEEE Robotics and Automation Letters*, vol. 4, pp. 2356–2363, 7 2019. 6, 9, 10

[24] S. Song, Z. Li, H. Yu, and H. Ren, “Shape reconstruction for wire-driven flexible robots based on bézier curve and electromagnetic positioning,” *Mechatronics*, vol. 29, pp. 28–35, 8. 6

[25] E. Coevoet, A. Escande, and C. Duriez, “Optimization-based inverse model of soft robots with contact handling,” *IEEE Robotics and Automation Letters*, vol. 2, pp. 1413–1419, 7. 6

[26] M. C. Yip and D. B. Camarillo, “Model-less hybrid position/force control: A minimalist approach for continuum manipulators in unknown, constrained environments,” *IEEE Robotics and Automation Letters*, vol. 1, pp. 844–851, 7 2016. 7

[27] S. Xu, B. He, Y. Zhou, Z. Wang, and C. Zhang, “A hybrid position/force control method for a continuum robot with robotic and environmental compliance,” *IEEE Access*, vol. 7, pp. 100 467–100 479, 2019. 7

[28] L. Toscano, V. Falkenhahn, A. Hildebrandt, F. Braghin, and O. Sawodny, “Configuration space impedance control for continuum manipulators.” Institute of Electrical and Electronics Engineers Inc., 4 2015, pp. 597–602. 7

[29] G. He, Y. Fan, T. Su, L. Zhao, and Q. Zhao, “Variable impedance control of cable actuated continuum manipulators,” *International Journal of Control, Automation and Systems*, vol. 18, pp. 1839–1852, 7. 7

[30] T. Su, L. Niu, G. He, X. Liang, L. Zhao, and Q. Zhao, “Coordinated variable impedance control for multi-segment cable-driven continuum manipulators,” *Mechanism and Machine Theory*, vol. 153, p. 103969, 11 2020. 7

[31] R. E. Goldman, A. Bajo, and N. Simaan, “Compliant motion control for multisegment continuum robots with actuation force sensing,” *IEEE Transactions on Robotics*, vol. 30, pp. 890–902, 2014. 7

[32] M. Mahvash and P. E. Dupont, "Stiffness control of a continuum manipulator in contact with a soft environment," 2010, pp. 863–870. 7

[33] ——, "Stiffness control of surgical continuum manipulators," *IEEE Transactions on Robotics*, vol. 27, pp. 334–345, 4 2011. 7

[34] S. M. Mustaza, C. M. Saaj, F. J. Comin, W. A. Albukhanajer, D. Mahdi, and C. Lekakou, "Stiffness control for soft surgical manipulators," *International Journal of Humanoid Robotics*, vol. 15, 10 2018. 8

[35] J. Back, T. Manwell, R. Karim, K. Rhode, K. Althoefer, and H. Liu, "Catheter contact force estimation from shape detection using a real-time cosserat rod model." IEEE, 9, pp. 2037–2042. 9, 10

[36] J. Back, L. Lindenroth, R. Karim, K. Althoefer, K. Rhode, and H. Liu, "New kinematic multi-section model for catheter contact force estimation and steering." IEEE, 10, pp. 2122–2127. 9, 10

[37] J. Back, L. Lindenroth, K. Rhode, and H. Liu, "Three dimensional force estimation for steerable catheters through bi-point tracking," *Sensors and Actuators A: Physical*, vol. 279, pp. 404–415, 8. 9, 10

[38] A. Bajo and N. Simaan, "Finding lost wrenches: Using continuum robots for contact detection and estimation of contact location." IEEE, 5, pp. 3666–3673. 9, 10

[39] ——, "Kinematics-based detection and localization of contacts along multisegment continuum robots," *IEEE Transactions on Robotics*, vol. 28, pp. 291–302, 4. 9, 10

[40] J. E. Bernth, J. Back, G. Abrahams, L. Lindenroth, B. Hayee, and H. Liu, "Endoscope force generation and intrinsic sensing with environmental scaffolding," vol. 2019-May. Institute of Electrical and Electronics Engineers Inc., 5 2019, pp. 1940–1946. 9, 11

[41] V. A. Aloi and D. C. Rucker, "Estimating loads along elastic rods," vol. 2019-May. Institute of Electrical and Electronics Engineers Inc., 5 2019, pp. 2867–2873. 9, 13, 44

[42] Q. Qiao, D. Willems, G. Borghesan, M. Ourak, J. D. Schutter, and E. V. Poorten, "Estimating and localizing external forces applied on flexible instruments by shape sensing," *2019 19th International Conference on Advanced Robotics, ICAR 2019*, pp. 227–233, 12 2019. 9, 13

[43] Q. Qiao, G. Borghesan, J. D. Schutter, and E. V. Poorten, "Force from shape–estimating the location and magnitude of the external force on flexible instruments," *IEEE Transactions on Robotics*, 2021. 9, 13, 49

[44] O. Al-Ahmad, M. Ourak, J. Vlekken, and E. V. Poorten, "Fbg-based estimation of external forces along flexible instrument bodies," *Frontiers in Robotics and AI*, vol. 0, p. 232, 7. 9, 13

[45] C. E. Bryson and D. C. Rucker, "Toward parallel continuum manipulators." Institute of Electrical and Electronics Engineers Inc., 9 2014, pp. 778–785. 17

[46] J. Till, C. E. Bryson, S. Chung, A. Orekhov, and D. C. Rucker, "Efficient computation of multiple coupled cosserat rod models for real-time simulation and control of parallel continuum manipulators." IEEE, 5, pp. 5067–5074. 17, 46

[47] A. L. Orekhov, C. B. Black, J. Till, S. Chung, and D. C. Rucker, "Analysis and validation of a teleoperated surgical parallel continuum manipulator," *IEEE Robotics and Automation Letters*, vol. 1, pp. 828–835, 7 2016. 17

[48] D. C. Rucker and R. J. W. III, "Statics and dynamics of continuum robots with general tendon routing and external loading," *IEEE Transactions on Robotics*, vol. 27, pp. 1033–1044, 12. 50

Vita

Vincent Aloi was born in Oak Ridge, Tennessee to parents Katie and Timothy. Motivated by a passion for physics and mathematics, in 2012 Vince started their 9.5 year journey into higher academia at the University of Tennessee, Knoxville. By 2016, they received a Bachelors of Science in Mechanical Engineering. Starting in UT's graduate program immediately afterwards, Vince became a member of Caleb Rucker's REACH lab where they would spend the remainder of their time during graduate education.