UC San Diego
UC San Diego Electronic Theses and Dissertations

Title

Tactile Perception for Growing Robots via Discrete Curvature Measurements

Permalink

https://escholarship.org/uc/item/7770v9w6

Author

Bryant, Micah Andrew

Publication Date

2022

Peer reviewed|Thesis/dissertation

UNIVERSITY OF CALIFORNIA SAN DIEGO

Tactile Perception for Growing Robots via Discrete Curvature Measurements

A thesis submitted in partial satisfaction of the
requirements for the degree of Master of Science

in

Engineering Sciences (Mechanical Engineering)

by

Micah Bryant

Committee in charge:

Professor Tania Morimoto, Chair
Professor Nick Gravish
Professor Michael Tolley

2022

The Thesis of Micah Bryant is approved, and it is acceptable in quality and form

for publication on microfilm and electronically.

University of California San Diego

2022

DEDICATION

For my family and friends who supported me through the toughest moments.

TABLE OF CONTENTS

LIST OF FIGURES

ACKNOWLEDGEMENTS

I would like to first say thank you to Professor Tania Morimoto for allowing me to join her research lab and helping me every step of the way. Without her guidance on this project I would never have been able to complete it.

I would also like to say thank you to Connor Watson for his assistance and advice, without which this project would have taken triple the amount of time that it did. Without his ideas and help this project would not be what it is today.

Additionally, I would like to thank all the members of Morimoto Lab for their support and advice during this entire project.

This thesis, in part, has been submitted to the IROS 2022 Conference, Micah Bryant; Connor Watson; Tania K. Morimoto. The thesis author was the primary investigator and author of this paper.

ABSTRACT OF THE THESIS

Tactile Perception for Growing Robots via Discrete Curvature Measurements

by

Micah Bryant

Master of Science in Engineering Sciences (Mechanical Engineering)

University of California San Diego, 2022

Professor Tania Morimoto, Chair

Soft, growing robots have the ability to conform to their environment and traverse highly curved paths that would typically prove challenging for other robot designs. As they navigate through these constrained and cluttered environments, there is often significant interaction between the robot and its surroundings. In this work, we propose a method to enable tactile perception for growing robots, which utilizes commercially available, flexible sensors that measure the curvature of the robot shape at multiple locations. Our method consists of both a pouch design to enable seamless integration of the sensors with the material of the growing robot, as well as an algorithm for determining the location of point contacts along the robot body. We validate our proposed approach experimentally and show that we can localize a force applied

to various locations along the length of a growing robot with an average error of 3.44 ± 1.38 cm when the robot is unactuated and 4.62 ± 0.95 cm when the robot is actuated. Additionally, we characterize the minimum distance required for our tactile sensing approach to discriminate between two separate contact points along the robot body. Finally, we apply our method to a growing robot exploring an unknown environment and show that we are able to effectively determine when and where the growing robot collides with an unknown obstacle.

Chapter 1

Introduction

Over the past decade, there has been a significant advancement in the study of soft robotics. There are several definitions for what a soft robot is, but soft robotics refers typically to a robot constructed out of materials that can be characterized by a relatively low Young's modulus, around the scale of biological tissues. While traditional robotics materials have a Young's modulus on the scale of $10^9 - 10^{12}$ pascals, biological tissues are typically much less rigid with a modulus on the scale of $10^4 - 10^9$ pascals [5]. This decrease in stiffness in the materials naturally causes an increase in flexibility and adaptability of the overall robot.

Research in soft robotics grew rapidly due to its potential use cases and benefits compared to more rigid robots. Soft material means it is safer to use around humans. The material acts as a cushion when colliding with a human thereby decreasing any damage from accidents arising from interactions around humans [6]. This soft material is also desirable for medical applications. It can help decrease any damage dealt to the soft tissues inside the human body during surgery and make direct human interaction safer [7]. Soft robots also have increased flexibility and adaptability. These properties give the robot the ability to be used in confined spaces and in some cases even change the shape of its body to squeeze through gaps in the environment [1]. These are desirable properties for use in search and rescue operations [8], archaeological exploration [9], or underwater observation where the environment is unknown [10]. The cushioning property is especially desirable in the case where the unknown environment may

1

contain objects that can be damaged by a moving rigid body such as an injured person or a coral reef [10].

However, with these benefits comes a trade-off. An increase in flexibility and adaptability also means an increased number of degrees of freedom, which can pose a challenge for control and modeling [5]. Many of the actuation methods for soft robots involves bending, twisting, stretching, or compressing the material of the actuator, which often follows a nonlinear trend [5]. These challenges are further complicated when the environment is constrained or highly cluttered which causes significant interaction between the robot and its surroundings [11]. These interactions need to be considered when looking at modeling, planning, and control.

There has been significant work on modeling soft robots' response to actuation in free-space and soft robots' response to interacting with an obstacle [11][2]. These planning algorithms are used to perform planning of the path a robot can take to reach a specified configuration or endpoint, however, to our knowledge all these cases to date assume that the robot knows the locations of all the obstacles in advance. As this is not a feasible assumption in many use cases where the robot is in an unknown environment, there is a need to develop methods for touch localization to determine when an obstacle collision has occurred and where along the robot length it has occurred. Tactile sensing has the potential to enable soft robots to locate obstacles and support structures, map their environments in real-time, and plan their movement even in previously unexplored environments. The method for sensing also needs to be integrated in such a way that it does not sacrifice many of the qualities that are desirable in a soft robot.

1.1 Everting Robots

This thesis focuses on everting robots. Everting robots, also known as vine or growing robots, are a form of bio-inspired soft robot that derive their inspiration from the growth mechanism of vines. This form of growth is called tip-extension where material is added to

the tip. As the only forward movement occurs locally at the tip there is drastically decreased movement relative to the environment from the rest of the robot body. There are several methods for replicating this form of growth such as tip-localized 3D printing [12], unspooling material at the tip [13], and feeding material from the base to the tip [11]. The method that is focused on here is the method of storing material at the base of the robot. The material is pushed out of the tip of the robot to perform growth. This method is performed by creating an invertible tube from a thin-walled material[11]. The material is inverted, and the inside of the tube is pressurized either by fluid or air. This pressure applies a force at the tip of the robot which drives additional material to evert out into the environment (see Fig. 1.1).

This method of growth gives several benefits to the robot, such as increased ability to travel tortuous paths [14] and reduced friction when moving in the environment [10]. These robots then have found use cases in minimally-invasive surgery [14], search and rescue [8], archaeological exploration [9], and ocean exploration [10] due to these properties, in addition to the flexibility and adaptability that is characteristic of soft robots in general.

Control of everting robots presents a unique challenge as the robot is constantly changing its overall length. Control of everting robots typically refers to actuating the robot to steer the direction of tip growth. Furthermore, the actuation preferably would sacrifice as little of the properties that make using an everting robot desirable, such as adaptability, ability to travel tortuous paths, and reduced friction when moving in the environment [15]. Typically the way to actuate an everting robot is by shortening or lengthening one side of the robot by applying a distributed strain to induce a distributed curvature along the length. In an inextensible everting robot this distrubuted strain would result in sinusoidal wrinkling along the inner edge of the bend (see Fig. 1.2a). Actuation can be done multiple ways such as attaching Serial Pneumatic Artificial Muscles (SPAMs) [16], Inverse Pneumatic Artificial Muscles (IPAMs) [17], fabric Pneumatic Artificial Muscles (fPAMs) [18], or tendon actuation [19].An alternative is to create a concentrated curvature along the length by preforming curves in the material [1]. This preformed

curve will take the form of a single or several strong wrinkles in the desired region to achieve the desired curvature (see Fig.1.2b).

For control and planning algorithms used on everting robots, the modeling of the robot shape change caused by an actuator in free space can be done by assuming a constant-curvature
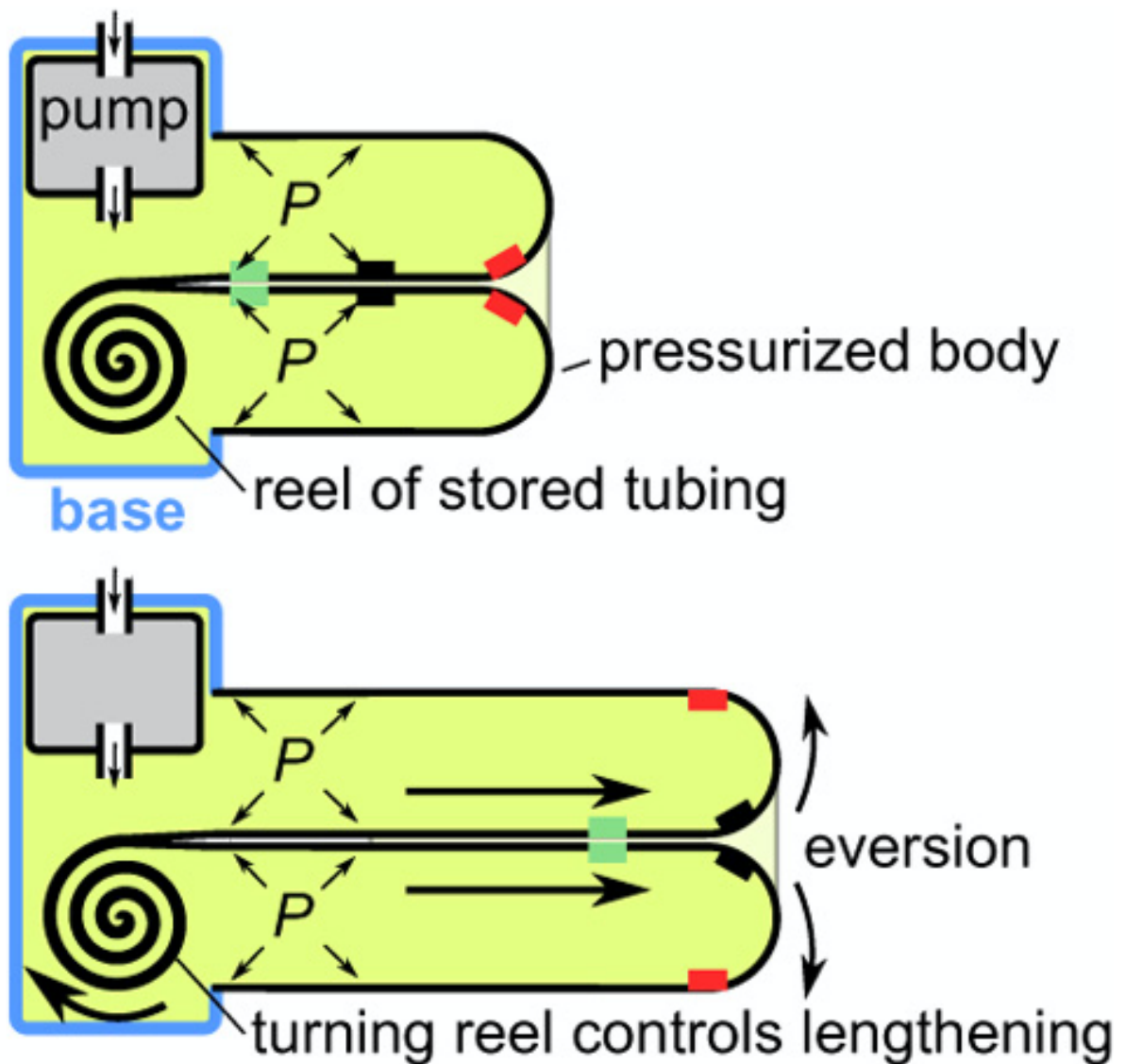
Figure 1.1. This figure which originally appeared in EW Hawkes et al [1] illustrates the mechanism for growth in an everting robot when the material is stored at the base. The material is spooled at the base and the inside of the robot is pressurized by a pump. This material then is pushed out through the tip by the pressure in the body. The length can be controlled by controlling the rotation of the spool.
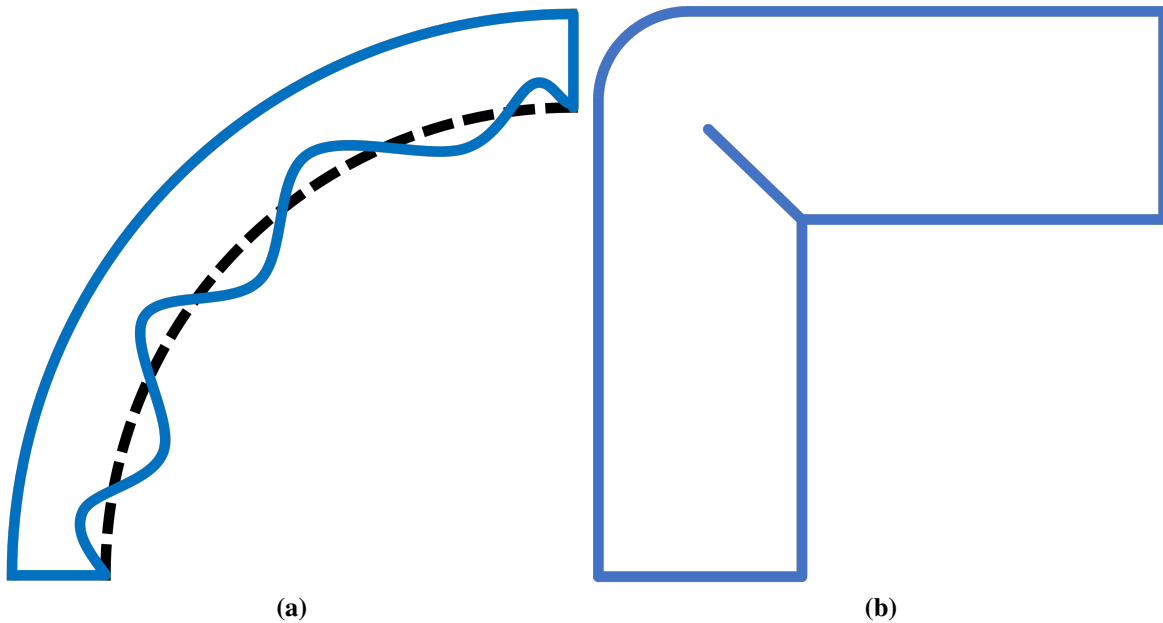
Figure 1.2. (a) Effect of applying a distributed strain load along the length of a relatively non-extensible everting robot body. The length of the material on the outer edge of the bend does not change in length, but the inside material undergoes wrinkling. (b) The effect of applying a concentrated strain load to a relatively non-extensible everting robot body. The material forms a single strong wrinkle and the remainder of the material does not undergo a shape conformation.

deformation model [2] (see Fig. 1.3). The shape is typically estimated by generating a mapping of curvature to pressure in the actuator for the pneumatic style actuators [20]. Object interaction when a robot is actuated has been analyzed by utilizing various analytical models such as an Euler Bernoulli (EB) model [2] or differential kinematics [11].

1.2   Sensor Integration

Sensor feedback about the current robot state and environment is important because it allows for more precise control in a workspace and understanding of the environment the robot is traveling into. It allows for active identification of obstacles [4], identification of tip position [21], and general information about the robot's surroundings [3].

However for an everting robot, the attachment of a sensor has many difficulties. These difficulties arise due to the unique form of movement that an everting robot performs. The
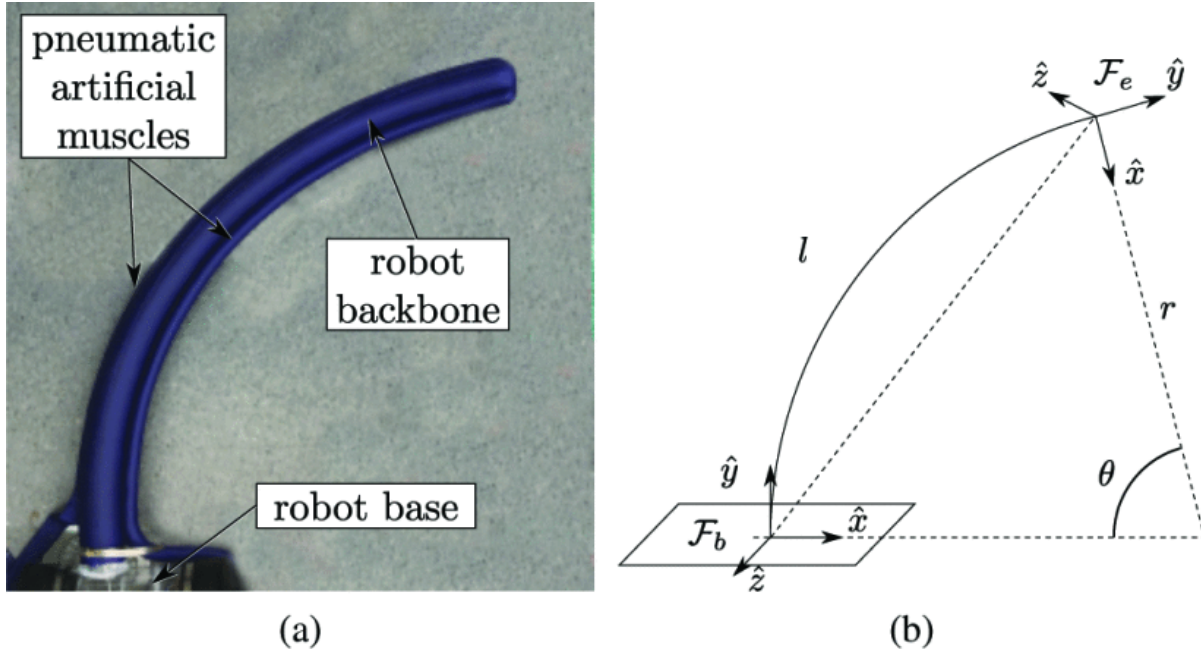
Figure 1.3. This figure which originally appeared in M Selvaggio et al [2] displays a side by side comparison of an actuated robot (a) and the associated constant curvature deformation (b) where r respresents the radius of curvature.

deployed material does not move laterally to the environment once deployed, the material at the tip has to be able to invert and evert, and the tip material constantly changes as the robot grows [15]. The main attachment areas for sensors to an everting robot is along the wall and at the tip of the robot.

For attachment at the tip, the unique challenge that the tip material constantly changes as the robot changes length must be overcome. This property of everting robots means that if an object wishes to be placed constantly at the tip location, it must be able to move relative to the material that forms the robot body. Tip attachment is desirable when attempting to deliver a payload to an object in the environment [22], for sensing using a camera to gain visual information about the environment [1], or for keeping a permanent magnet at the tip for localization [21]. There are a few methods that address how to deal with this issue. If the sensing mechanism is placed outside the robot, the mechanism can be moved forward by the motion of the material, and it can be constrained from going further than the robot location by
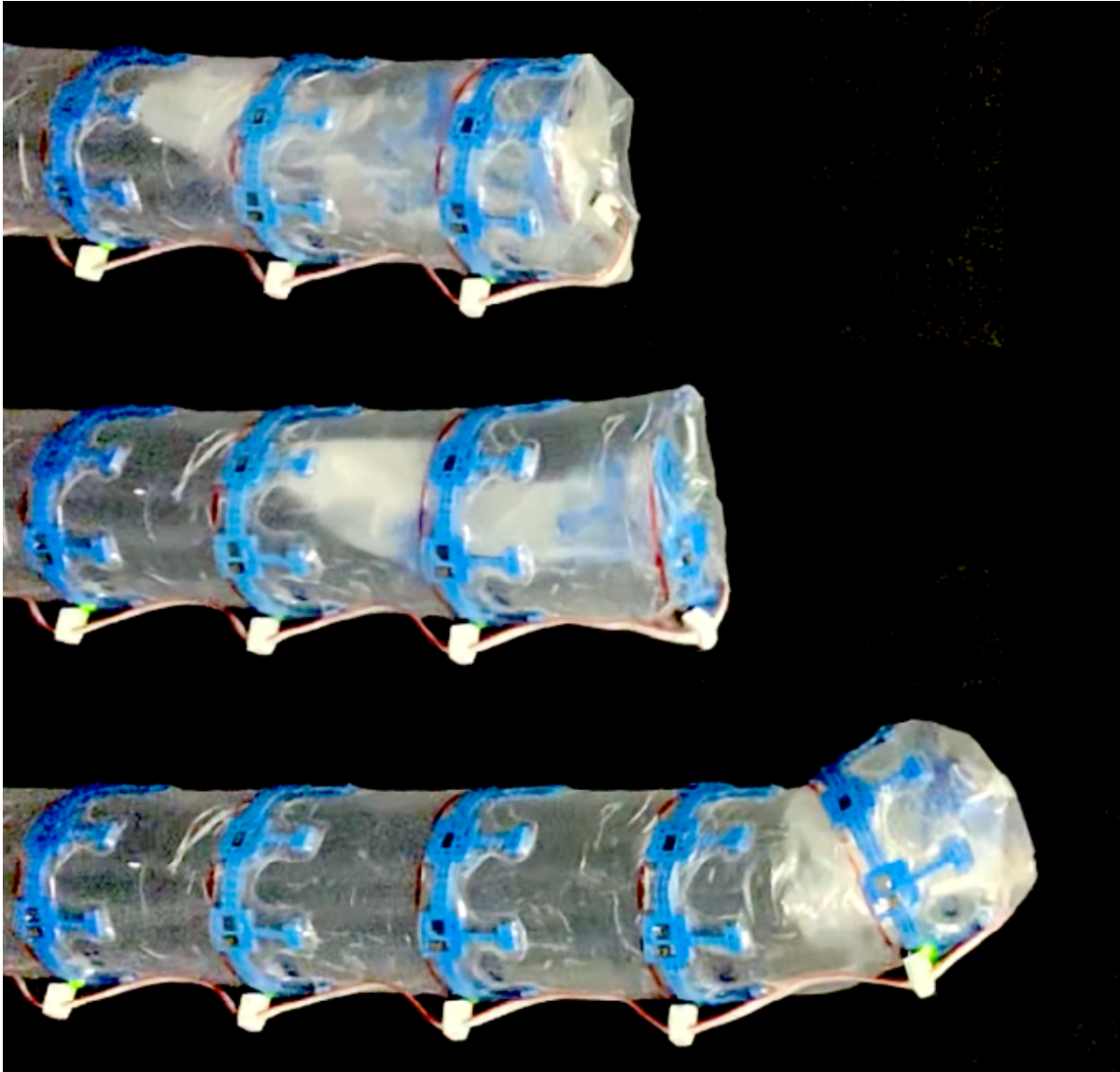
Figure 1.4. This figure which originally appeared in AM Gruebele et al [3] shows a method of attaching sensors to the outer wall of an everting robot.

having some constraining element such as applying a tension to the wire of a camera [1]. If the mechanism is placed inside the robot, the device can be pushed forward by the friction of the material moving inside the robot, and it will be constrained by the material itself [21].

The final major method for attaching sensors to an everting robot is to attach it along the body. The method here is to anchor a sensor or sensor package to the wall. They can be anchored either inside or outside of the robot (see Fig. 1.4). Attaching the sensors outside

allows for the sensors to interact with the environment, but also exposes the sensors and any associated wires to the environment as well which may allow them to sustain damage at their deployed location [3]. To protect the sensors, they can be placed inside of the robot. As this stops the sensors from being able to measure values outside of the robot, this is used when the sensors are only going to measuring parameters about the robot itself rather than the outside environment [15].

One major challenge that is still present in the integration of sensors to an everting robot lies with attaching sensors to the wall of the robot. Attaching sensors to the wall of the robot can create stiffness mismatches which can then make it difficult to accurately use a mechanical model for the robot as well as still allow the robot to move as required. In addition, if these sensors are used to gather shape estimates, the signal could be significantly reduced in comparison to the parts of the robot where the sensors are not placed.

1.3   Soft Robot Touch Localization Methods

Touch localization can be used when growing in an unknown environment to determine potential support structures in a manner similar to biological vines [23] or to map out obstacle locations to be used in either planning or avoidance algorithms [11] [2]. There are several methods that are used to perform touch localization. Robots such as continuum and everting robots have forces acting upon them influence their overall shape due to the force being distributed down the entire shape [4]. As they are also much longer than they are wide, they can also be estimated as a thin beam [24]. Touch localization methods have much in common with force estimation. Typically for force estimation on continuum robots, force is assumed to be applied at the tip [25][26][27]. There are several methods that relax this constraint by assuming that both the force and the location are unknown [4][20][28]. However, these methods have not been used on a growing robot and either constrain the movement to one degree-of-freedom, require images taken of the robot, or require continuous shape information for the robot.
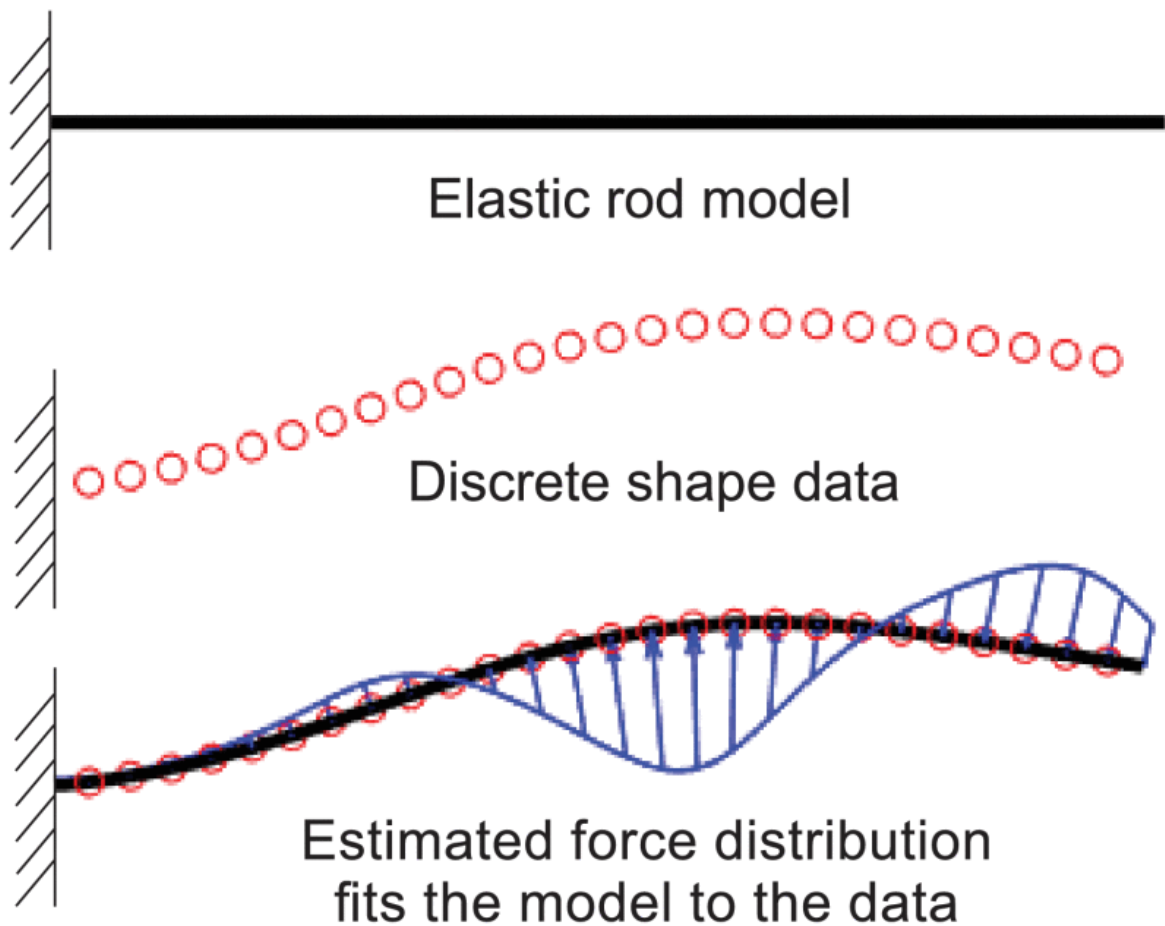
Figure 1.5. This figure which originally appeared in VA Aloi et al [4] displays the discrete shape data collected from the overhead camera and a predicted force distribution and shape estimate using a Cosserat-rod model.

For the methods that assume that both the force and location of a contact is unknown, there are several sensing modalities used to find these parameters. The first involves using an electromagnetic (EM) tracker. This tracker gives information with 5-degrees of freedom (DOF). Using this information a model can be fit such that the difference between the sensor readings and the model output are minimized [20]. In this method a joint kinematic model is used to compare against the sensor information [20].

Another method to determine contact force and location utilizes shape information about the robot. This can be found using an overhead camera to create a set of discrete data points

that return robot position [4] (see Fig. 1.5). Another method that has been done to collect the shape information is to use Fiber-Bragg Grating (FBG) shape sensors to reconstruct the shape [28]. The FBG generates a series of strain measurements which are converted to curvature measurements. These curvature measurements are finally integrated together to form a 3D shape of the robot [28]. For the shape information based estimations, a model is constructed using a mechanical model, however instead of having a single data point containing position and pose information, a distributed set of data points is used to compare a model prediction of shape to the measured shape [4].

However, touch localization has yet to be implemented on a growing robot. In addition, the methods to use distributed information about robot shape either require a camera to record robot shape 1.5, which is not feasible in many use cases, or requires an FBG [28], which is extremely cost prohibitive, can be difficult to incorporate into an everting robot, and occupies the space in the center of the robot which may need to be used to hold various other tools to be used depending on the robot application.

1.4   Contributions

Our contributions are as follows. (1) We present a method to localize contact location on an everting robot using discrete curvature measurements in conjunction with a mechanical model. This localization is shown to work in a passive, multiple contacts, actuated, and growing case. (2) We present a method to attach sensors to the walls of an everting robot that give shape information without disrupting the mechanical and analytical models used in analysis of an everting robot.

Chapter 2

System design

2.1   Growing Robot

Our growing robot (Fig. 2.1) is fabricated using a lightweight, silicone-urethane impregnated ripstop fabric (Seattle Fabrics) that is flexible, but relatively inextensible. The main body of the robot is fabricated by forming this material into an invertible tube. Fabrication is completed by forming a lap joint with the fabric that is glued together with Sil-poxy Silicone
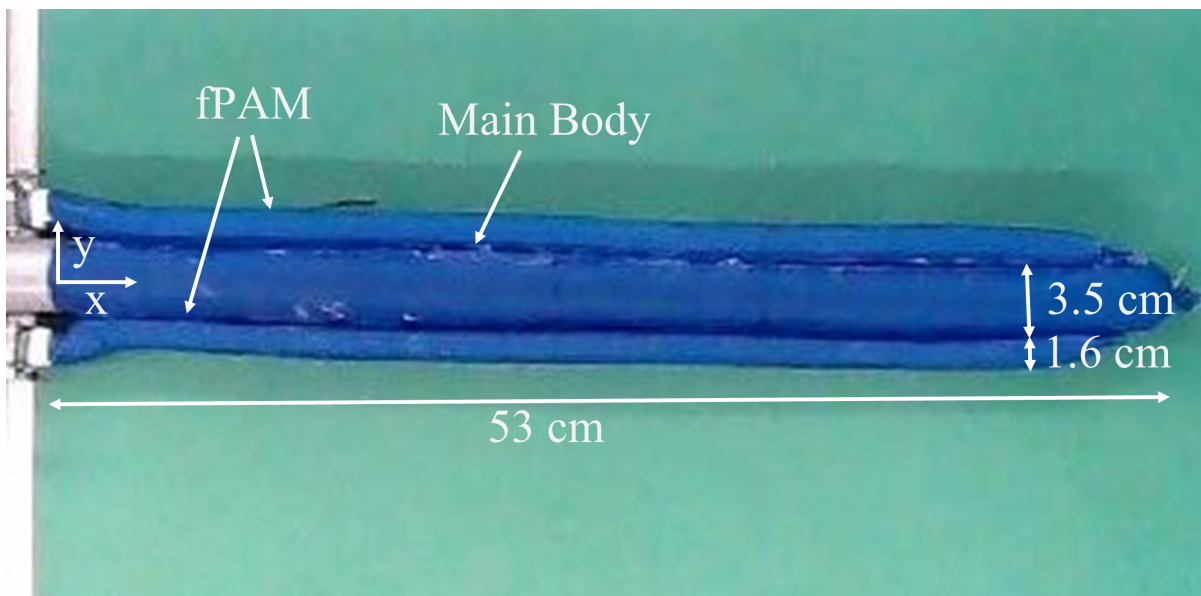
Figure 2.1. The fully deployed growing robot, along with critical dimensions for both the main robot body and the fabric pneumatic artificial muscles (fPAMs). The coordinate axes are also defined, where *x* points in the direction of the straight robot configuration and *y* is orthogonal to *x*.

adhesive (Smooth-On). After the lap joint is glued together, the length of the robot is 53 cm. The diameter of the robot, when inflated, is 3.5 cm, which was chosen to ensure that the selected sensors, explained in detail in the subsequent section, could be easily everted. The distal end of the robot is inverted through the center of the robot and connected using a string to a motorized spool housed inside the pressure vessel. The proximal end of the robot body is attached to an outlet of a pressure vessel, whose internal pressure can be controlled to induce robot growth. The robot grows by having an internal pressure that exerts a driving force on the tip of the robot (see Fig. 1.1). In order for growth to occur, the driving force must overcome friction forces present in the robot due to various materials moving within the robot such as wires or the robot material itself.

The pressure vessel consists of a cylindrical plastic housing that contains an air inlet port connected to a mechanical pressure regulator (SMC AR20K-N02-Z-B) as seen in Fig. 2.2. A rubber cap is used with a hose clamp to form an airtight seal over the top of the housing. The length of the robot is controlled by using a motor in coordination with an encoder (Cytron 12V 225RPM DC Geared Motor with Encoder) to determine how much of the string is unspooled.

2.2   Robot Actuation

We chose to use fabric pneumatic artificial muscles (fPAMs) [18] for steering of the robot. These were chosen due to the ease of attachment, the ability to get adequate curvatures when activated, and the ability to generate predictable curvatures in the robot body. Two fPAMs are fabricated with a diameter of 1.6 cm using the same material as the main body. They are attached on opposite sides of the growing robot body using the same Sil-poxy silicone adhesive as was used to fabricate the main body (see Fig. 2.1) and are used to bend, or steer, the robot in the plane, in a manner analogous to tendons.

An fPAM is based off of the working mechanism of a McKibben muscle. There is a material that has two inextensible threads woven throughout that are orthogonal to the other
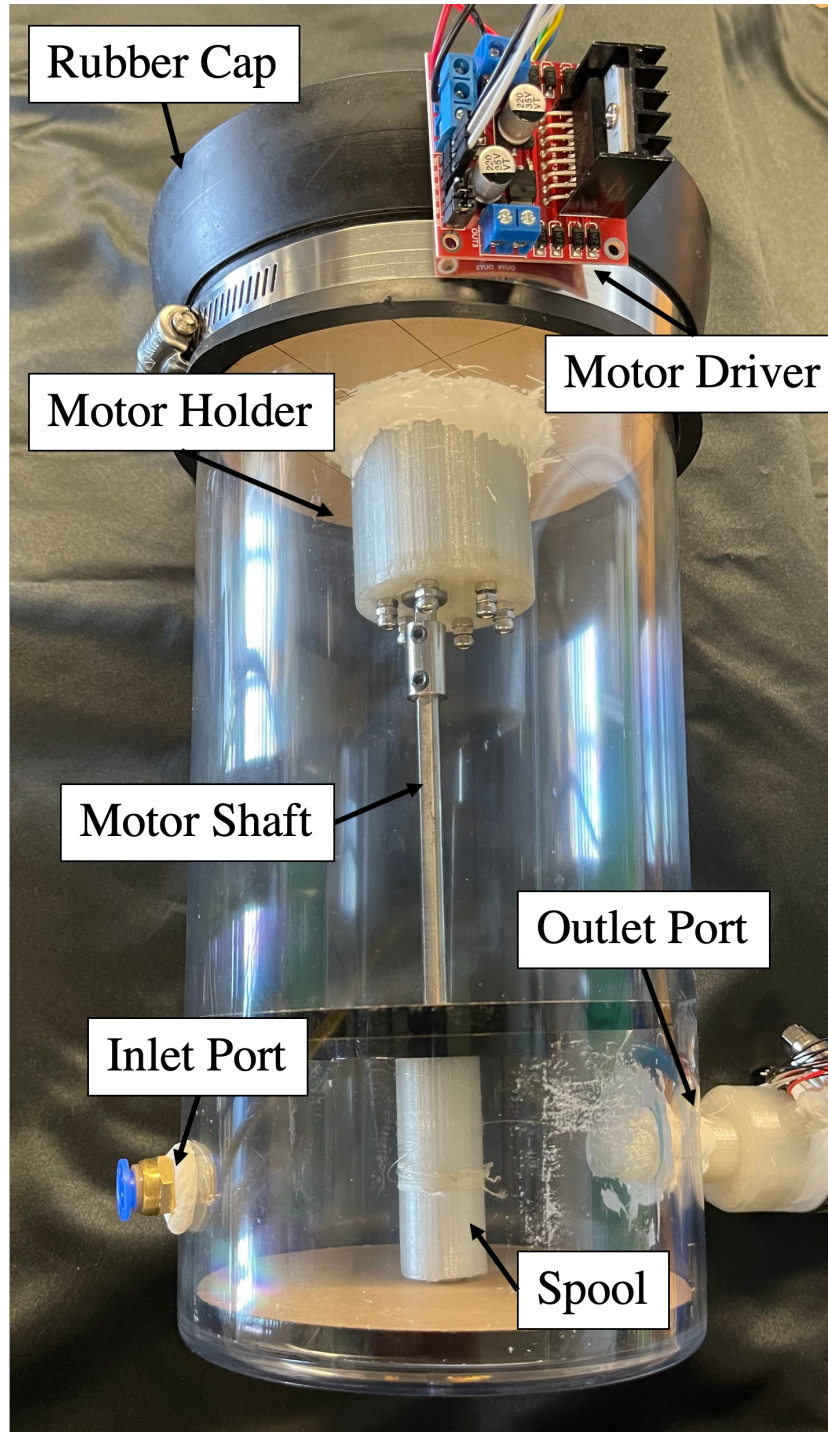
Figure 2.2. The pressure vessel used for length control of the everting robot. A spool connected to a motor with encoder is controlled in via an Arduino Mega through a motor driver. The vessel is made airtight with a rubber cap, is pressurized by an inlet port, and is able to pressurize the main body of the everting robot through an outlet port.
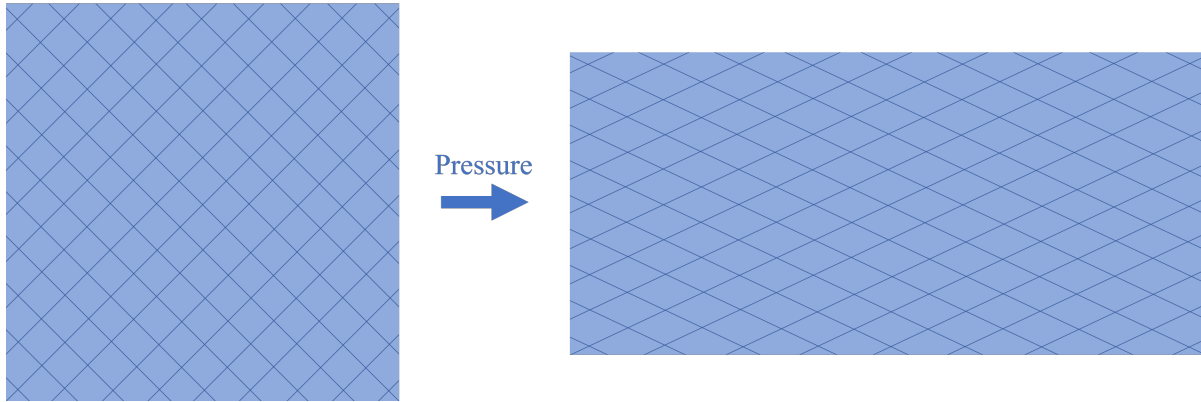
Figure 2.3. The shape deformation of the material used to fabricate a fabric pneumatic artificial muscle (fPAMs) when going from unpressurized to pressurized. As the material inflates, the material expands out radially causing a shift in the inextensible fibers. The inextensible fibers tie a radial change to a length change. This then causes a contraction of the overall material in the actuator.

(see Fig. 2.3). The material is cut such that these threads are at a 45 degree angle relative to the longest and shortest length of the actuator. When the pressure inside an fPAM is increased, the material expands radially. This radial expansion causes an angular change between the two threads thereby causing a contraction of the overall actuator. Since the fPAM is coupled to one side of the robot, a contraction of the fPAM causes a shortening of that side of the robot, which causes the entire robot body to bend. It is important to note that due to the relatively inextensible nature of the robot body, the shortening of that side of the robot is primarily caused by wrinkling in the main body material [29]. The pressure in each fPAM is controlled by a separate NITRA Current to Pneumatic Transducer (NCP1-20-260N and NCP2-20-260N). The current sent to the regulators is maintained in a pressure feedback loop established using a PID controller and two Honeywell pressure sensors (SSC-series 60 PSI). Both the actuator pressures and the motorized spool position are set by an Arduino Mega 2560, which allows for control of two independent degrees of freedom — the deployed length of the robot and its tip orientation in the plane.

A mapping was created to determine the robot curvature as a function of actuator pressure. This mapping is possible because an fPAM applies an approximately constant moment along the length of the robot [18]. This creates an approximately constant curvature deformation
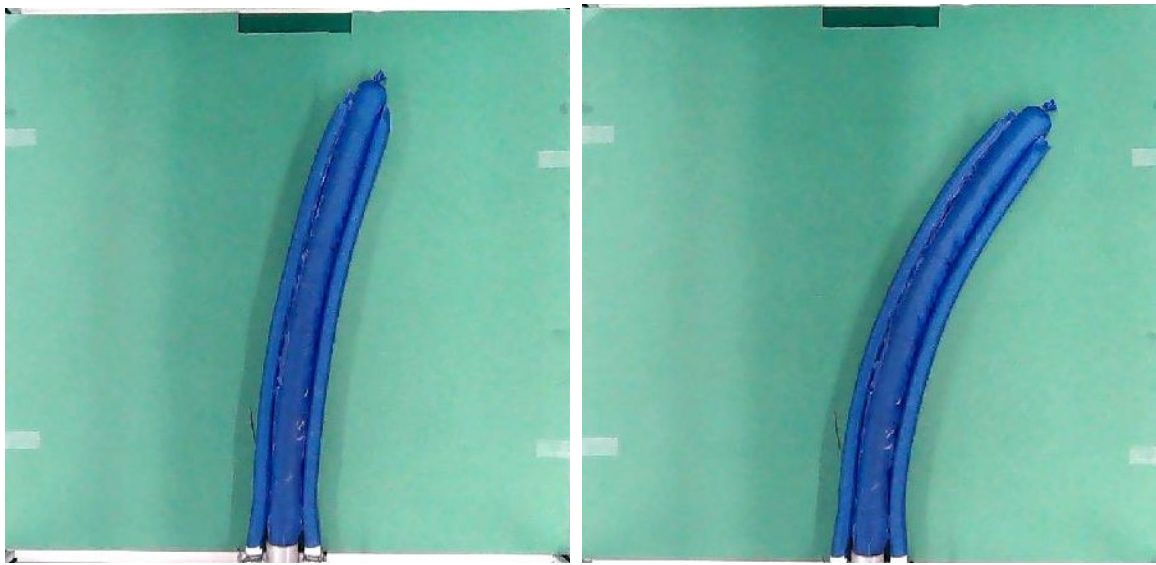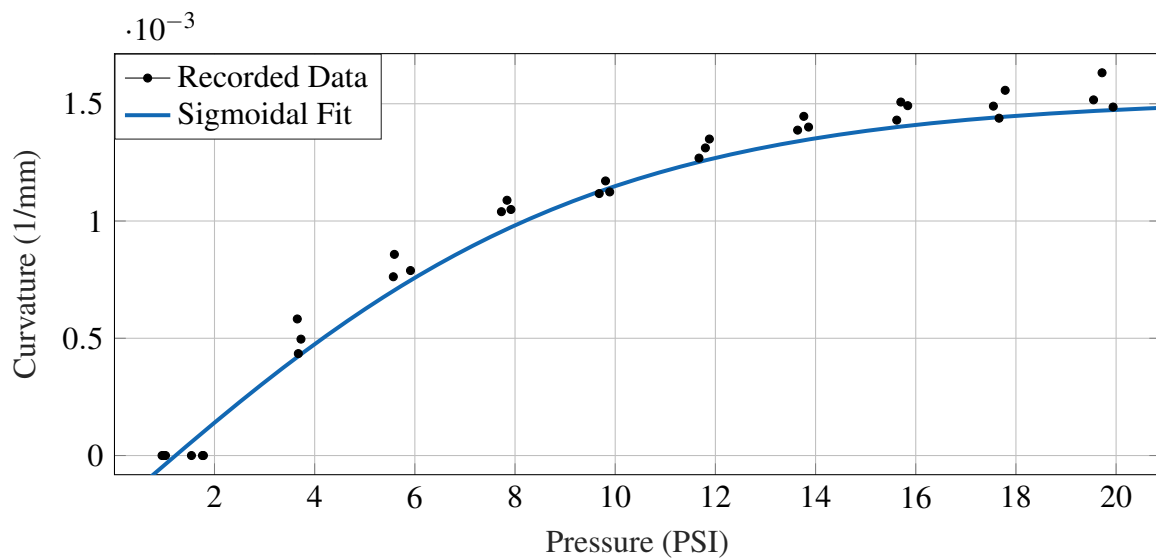
Figure 2.4. (a) This is the deformation the robot took when 6 PSI was applied to the actuator on the right side. (b) This is the deformation that the robot took when 20 PSI was applied to the actuator on the right side. (c) A graph showing the fit line generated with a sigmoidal function (hyperbolic tangent) when supplied pressures applied to the actuator on the right side and the associated measured curvature from analyzing the overhead image.

through uniform wrinkling of the material that the actuator is attached to (see Fig. 1.2). This

allows for a mapping of a single curvature to a single pressure in the actuators [2]. The mapping

was created by applying 11 different pressures (0 to 20 psi in 2 psi increments) to each actuator across 3 trials, recording the curvature using an overhead camera, and fitting the data to a hyperbolic tangent function (see Fig. 2.4).

2.3   Sensor Selection

Integrating sensors onto the flexible, thin-walled body of a growing robot without adversely affecting the desirable attributes of the robot (e.g. eversion and compliance) is difficult. Many of these challenges stem from the significant stiffness mismatch that typically exists between the sensor materials and robot materials. In addition, the robot can experience uncontrolled wrinkling and the eversion process induces tight curvatures, all of which can lead to significant stress on any integrated sensors. Further, most of the sensing options that do have similar mechanical properties to growing robots — such as fabric-based strain sensors [30] or soft force sensors [31] — are still in the research phase of development, hindering wider adoption. For these reasons, we seek a solution for tactile perception utilizing sensors that both have flexibility similar to growing robots and are available for purchase off-the-shelf. We find that flexible, resistance-based sensors (Spectra Symbol 2.2") are an appropriate choice given these design requirements, provided they are properly coupled to the growing robot body. As seen in Fig. 2.5, the sensor is flexible enough to undergo eversion.

Each sensor is 5.5 cm long and has a resistance that is linearly proportional to the total change in bending angle across the entire sensor which is represented by the following equation:

$$R_i = C_i * \Delta\theta_i + R_{o,i}, \tag{2.1}$$

Where $\Delta\theta_i$, R_i, $R_{o,i}$, and C_i refer to the total change in bending angle, measured resistance, resistance in sensor at rest, and proportionality constant relating bending angle to resistance in the sensor respectively. It is important to note that for each sensor there is a different proportionality constant and resting resistance. Using the EB beam model to relate the bending

16

| Inverted Sensor | Partially Everted Sensor | Everted Sensor |
| (a) | (b) | (c) |
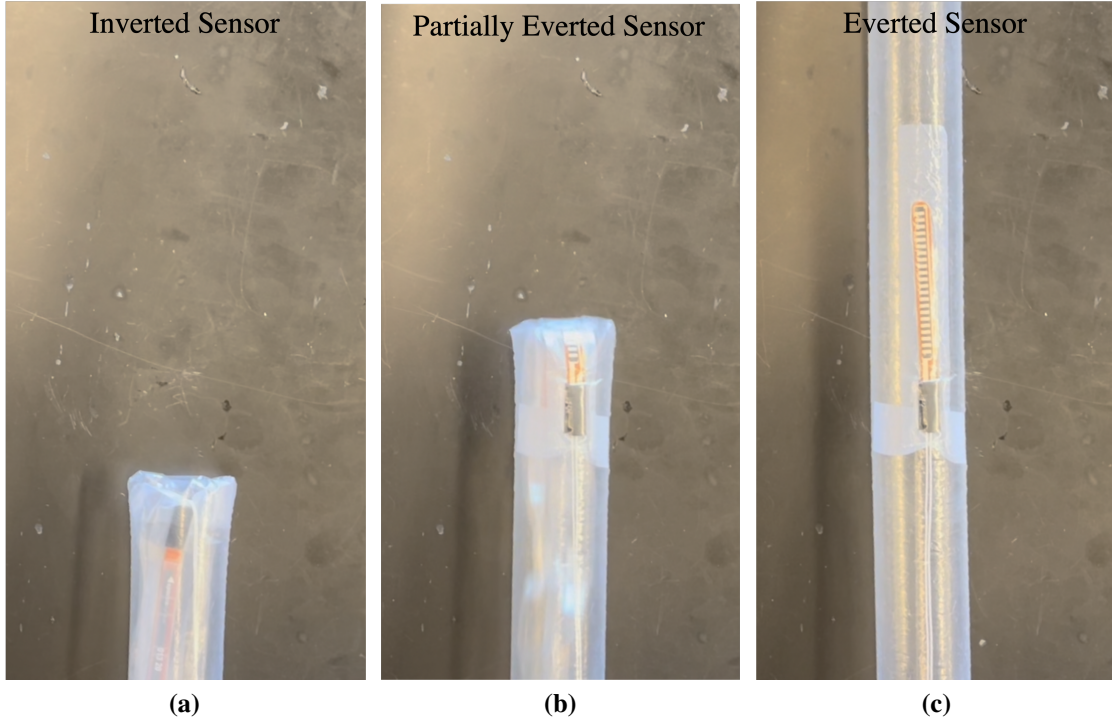
Figure 2.5. A flex sensor has been attached on to the inside of a clear plastic everting robot wall and is pictured while (a) inverted, (b) undergoing eversion, and (c) fully everted.

angle to the curvature, the following relationship is achieved:

$$k_i = \frac{d\theta_i}{dx_i},$$ (2.2)

where k_i and x_i refer to the instantaneous curvature and x location along a specific sensor respectively. Solving for the average curvature along the sensor length yields:

$$k_{avg,i} = \frac{\Delta\theta_i}{l_i},$$ (2.3)

where $k_{avg,i}$ and l_i refer to the average curvature along a sensor length and total sensor length respectively. Rearranging 2.3 and substituting into 2.1 allows for a relationship between the average curvature along a sensor and the resistance reading of the sensor. Combining constants yields the following relationship which is used to relate resistance to curvature:

$$k_{avg,i} = C_i * (R_i - R_{o,i}), \tag{2.4}$$

where C_i now represents a new proportionality constant relating resistance of a single sensor to the average curvature of that sensor.

These resistances are read in using a voltage divider circuit created using a single 22 kOhm resistor for each sensor and an analog output sent to the Arduino Mega 2560. Using multiple of these sensors, we can determine local curvature measurements at discrete locations along the growing robot backbone.

2.4   Sensor Integration

Simply adhering the sensors directly to the growing robot is problematic because the robot behaves mechanically as an inflatable beam [24]. Therefore, if stiffer locations exist along the robot body — caused by something such as a flex sensor adhered directly to the robot fabric, for example — the robot will buckle around these locations when the robot encounters a contact force, rather than forming a smooth curve [32] [11]. Because buckling causes a discrete change in the curvature between sensors, it is not measured, making the contact force unobservable. Another situation is that the wrinkling, which is the mechanism for bending of an everting robot, may occur uniformly everywhere besides at the sensor locations due to the stiffness mismatch between the sensor and the robot body [29]. The wrinkling in this case would occur at a much lower frequency and magnitude at the sensor locations or even not at all causing lower or no signals from the sensors.

To solve this issue, we introduce a pouch device (Fig. 2.6) to house and fasten the curvature sensors to the growing robot. The pouch is made out of the same material as the robot body, sewn closed on three sides and sized such that the sensor can slide inside, while being constrained from unwanted movement (10 cm long, 0.75 cm wide). The pouches are then glued onto the inside body of the robot, using the same Sil-poxy Silicone adhesive, and the sensor is
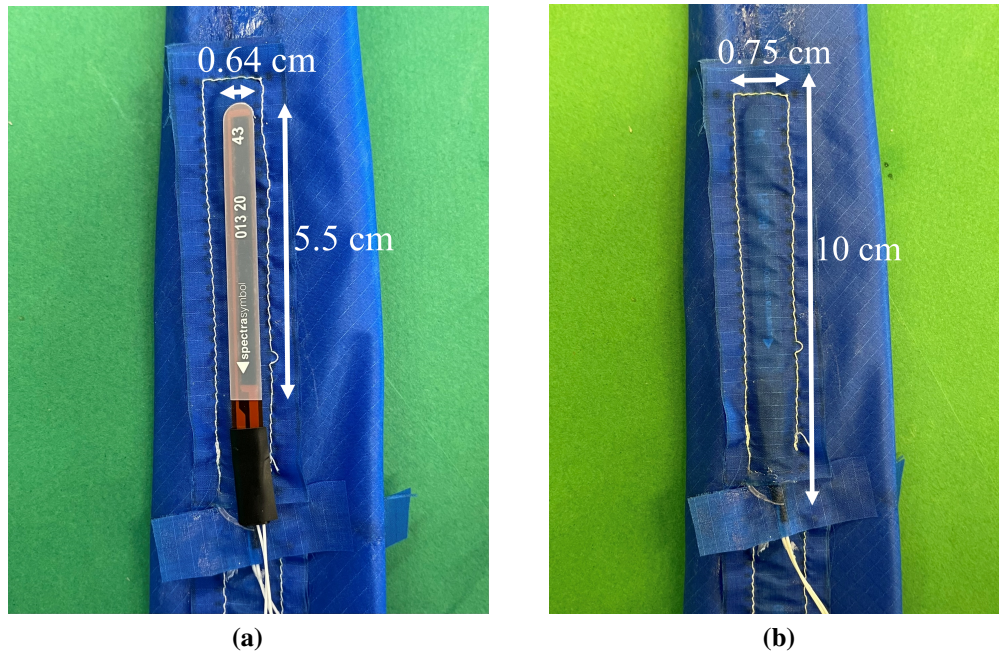
Figure 2.6. (a) Example off-the-shelf, flexible sensor used to measure local robot curvature, along with its dimensions. (b) To integrate the sensors with the growing robot, they are sealed inside individual sensor pouches with dimensions as shown.

locked in place by gluing another small piece of fabric over the open entrance.

This pouch allows the robot to bend, while avoiding the non-uniform wrinkling and buckling issues that can be caused by stiffness mismatches. Specifically, for a growing robot made from a relatively inextensible material to bend in a given direction, the length of the inner side of the curve must shorten. However, as this material is relatively inextensible the main mechanism that the fabric uses to shorten on the inside curve is by wrinkling. The pouch enables the material to act in a way that we would expect from a robot body with no sensors attached. As the sensor is no longer directly adhered but is simply constrained, the robot material can act in a predictable way that can follow analytical models. This allows for the sensors to be used to model contact and actuator interactions. This design makes the combined stiffness of the sensorized robot more uniform and more effectively transmits applied forces to the sensors so that changes in curvature can be measured. For our robot, we place five, 10 cm long pouches into a single column along each side of the robot.

19

2.5 Touch Localization Algorithm

We consider a planar robot, which we assume to behave as a cantilever beam and can be described using an EB beam model [24] as

$$EI\frac{d^4y}{dx^4} = q(x), \tag{2.5}$$

where E is the Young's modulus of the robot, I is it's area moment of inertia, and y is the deflection of the robot due to applied load q at length x. For N_c point contacts, the function describing the applied load is

$$q(x) = \sum_{i=1}^{N_c} F_i \delta(a_i - x), \tag{2.6}$$

where $\delta(\cdot)$ is the Dirac delta function, F_i is the magnitude of the ith point contact force, and a_i is the contact location along the robot body. The position and orientation of the robot is fixed at its base, which imposes boundary conditions

$$y(x)\bigg|_{x=0} = 0, \qquad \frac{dy}{dx}\bigg|_{x=0} = 0. \tag{2.7}$$

As is common for growing robots with pneumatic artificial muscles, we further assume that actuation causes constant curvature deformation of the robot shape [2], leading to another boundary condition

$$\frac{d^2y}{dx^2}\bigg|_{x=L} = g(p), \tag{2.8}$$

which states that the curvature of the robot at the free end is a function of the actuator pressure, $g(p)$. If the robot is unactuated, then $g(0) = 0$. In the case where the robot is actuated, we can empirically fit a function that relates the constant curvature of the robot to the actuator pressure, as described in Section 2.2. Integrating Eq. (2.5) twice with these boundary conditions gives a

closed-form expression for the robot curvature,

$$\frac{d^2y}{dx^2} = \hat{k}(x) = g(p) + \sum_{i=1}^{N_c} \frac{F_i}{EI}(a_i - x)H(a_i - x), \qquad (2.9)$$

where $H(\cdot)$ is the Heaviside step function.

In order to determine the location of the point contact forces applied to the robot, we can pose an optimization problem to minimize the difference between the curvature sensor measurements and the curvature predicted by model (as in Eq. (2.9)) at the location of each sensor,

$$\min_{\mathbf{a},\mathbf{f}} \sum_{i=1}^{N_s} ||k_{sensor,i} - \hat{k}(x_i)||_1, \qquad (2.10)$$

where $\mathbf{a} = [a_1, ... a_{N_c}]$ is a vector of point contact locations, $\mathbf{f} = \frac{1}{EI}[F_1, ... F_{N_c}]$ is a vector of the magnitudes of each contact force normalized by the robot bending stiffness, x_i is the position of the ith sensor on the robot body, and $||\cdot||_1$ is the 1-norm. We choose to minimize the 1-norm to mitigate the impact of outliers due to sensor noise [33]. We solve this problem using an interior point method implemented by MATLAB's *fmincon* routine and impose the constraint that $a_i \in [0, L]$. Additionally, we find that solving for both $\mathbf{a}$ and $\mathbf{f}$ simultaneously, typically finds poor locally optimal solutions for the point contact locations, $\mathbf{a}$, due to the magnitude of the gradients with respect to each variable. The force term has a gradient that is on a scale of 10^9 larger than the gradient from the location term which causes the force term to be fit relatively well while very little change occurs in the location term.

Therefore, we iteratively optimize for either $\mathbf{a}$ or $\mathbf{f}$, one at a time, while holding the other constant until the difference between successive solutions for all of the point contact locations is less than a tolerance (0.5 cm). This is similar to performing a nonlinear conjugate gradient descent but with interior-point as the optimization algorithm rather than gradient descent, as we are forcing each successive step to be perpendicular to the next since the force and location term are perpendicular to each other in their contribution to the residuals. For initialization

21

.
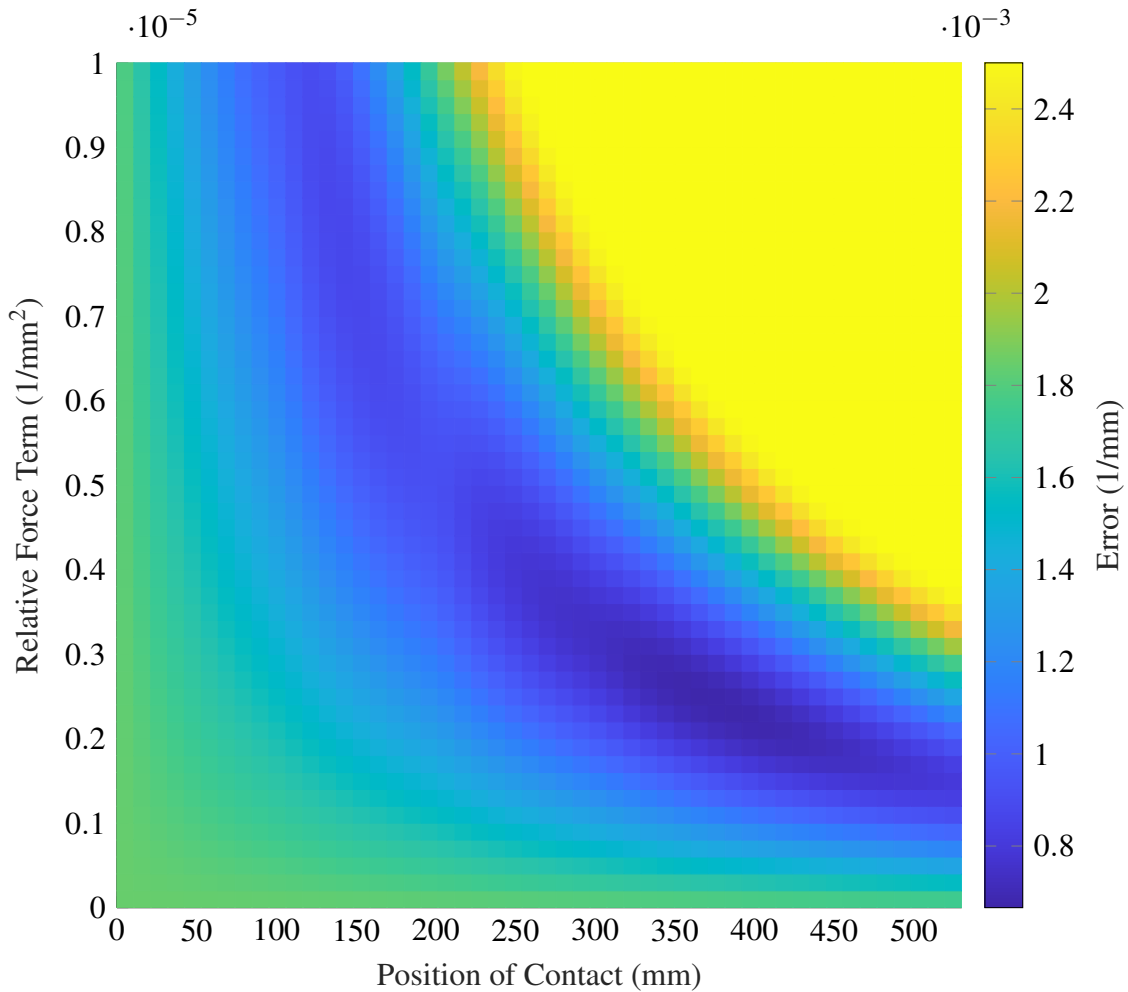
Figure 2.7. The surface plot of the residuals is shown for the case where a high force is applied to a fully deployed robot at $\frac{3}{4}$ of the length. The value of the residuals is developed by comparing the curvature sensor readings to what the curvature would be generated by applying a relative force (y-axis) at a specified distance from the base (x-axis). As can be seen there is a ravine generated in the residuals with two visible local minima. One around 130 cm and one at 350 cm. The true contact for this trial occurred at 380 cm

of each term, we initialize the force term at 0 and the position term to be at the tip. The

force initialization can begin at any point, but initializing at zero ensures we do not bias the

optimization towards assuming a right or left contact. The position initialization makes sense

for two reasons. The first is that when growing and actuating into an environment we would expect a contact to occur closer to the tip so initializing at the tip makes logical sense for the use case of this optimization. The second is that the the residuals form a ravine when optimizing over the terms as seen in Figure 2.7. This is because of the piece-wise nature of the residuals function. A lower contact position initialization allows for overfitting to the sensors nearest to the base while ignoring the sensors that may have relatively low signals that can be discounted thereby finding a local minima that is not accurate. This can be seen in Fig. 2.7 where two local minima can be seen. One minima finds a good fit to only the two sensors closest to the base while ignoring all other sensor information. The other minima finds the actual contact location. As we expect the contact locations to occur near the tip as the robot grows through an unknown environment we desire to find the minima in this ravine that is nearest to the tip. In the case of multiple contacts we split the length into uniform sections for initialization (e.g. for 2 point contacts we initialize the contacts at $\frac{L}{2}$ and L).

2.6   Algorithm Implementation

As described above, the sensors used in our experiments have a resistance that changes proportionally with their curvature. In order to determine the proportionality constant for each sensor integrated into the robot, we collect resistance measurements from all sensors as the tip of the unactuated robot is deflected to 10 different locations, and we record the tip positions using an overhead camera. Each of these tip position measurements, y_{tip}, is used as a boundary condition,

$$y(x)\Big|_{x=L} = y_{tip}, \tag{2.11}$$

together with the conditions in Eq. (2.7) and Eq. (2.8) to integrate Eq. (2.5) and solve for the force of the contact.

$$\frac{F}{EI} = \frac{3y_{tip}}{x_{tip}^3} \qquad (2.12)$$

The force and location of the contact can then be plugged into Eq. (2.5) to find the instantaneous curvature at the center of each sensor.

For each deflection location, the sensor that is being analyzed is first set to 0 when there is no curvature present in the robot. This is done by subtracting the resistance in the sensor at zero curvature from the current curvature reading. This is equivalent to setting $R_{o,i}$ in Eq. (2.4) to 0. We then make the approximation that the average curvature along a sensor length is approximately equal to the instantaneous curvature found at the center of the sensor location. This allows us to use the curvature measurements determined from the camera data to find the proportionality constant for each sensor in the Eq. (2.4).

We compute a linear regression to fit each of the proportionality constants. Once these constants are computed, they can be used to determine the curvature of each sensor from its resistance measurement at any given time.

In addition to using the sensor curvatures to find the location of applied point contacts, we also use the sensor measurements to detect when the robot comes into contact with an unknown force. When the robot is not experiencing a contact force, it should have a constant curvature that depends on its actuator pressure, $g(p)$. If the average curvature measured by the sensors at the base of the robot deviates beyond an empirically determined threshold (0.005 cm^{-1}), then the robot is determined to be in contact with an obstacle and optimization can be used to determine the location of the contact. The base sensors are chosen for this as the moment exerted by a contact is maximized at the base. This causes the discrepancy between the predicted model and the measured value to be maximized at the base sensor.

Chapter 3

Experimental Results

We validate our proposed method for tactile sensing in growing robots through four different experiments. First, we characterize the localization accuracy of our method for a single point contact at different forces and locations on an unactuated robot body. Second, we determine conditions under which our method can discriminate between two contacts on an unactuated robot. Third, we assess contact localization accuracy for an actuated robot. Finally, we demonstrate an application of our method, wherein a growing robot uses our proposed tactile sensing to determine when it is in contact with obstacles as it explores an unknown environment.

3.1 Experimental Setup

A test environment was fabricated to assess our sensing methodology (Fig. 3.1). Metal dowels were placed at specified locations in the workspace and served as contacts with which the robot could interact. Their positions, relative to the robot base, were determined from 1920x1080 resolution images taken with an overhead camera (Nexigo N980P). The x and y axes of the coordinate frame of the growing robot are defined as in Fig. 2.1. In all experiments, localization error is defined as the absolute difference between the x position of the obstacle contact, as estimated by the localization algorithm, and the true obstacle position, as determined by the overhead camera.
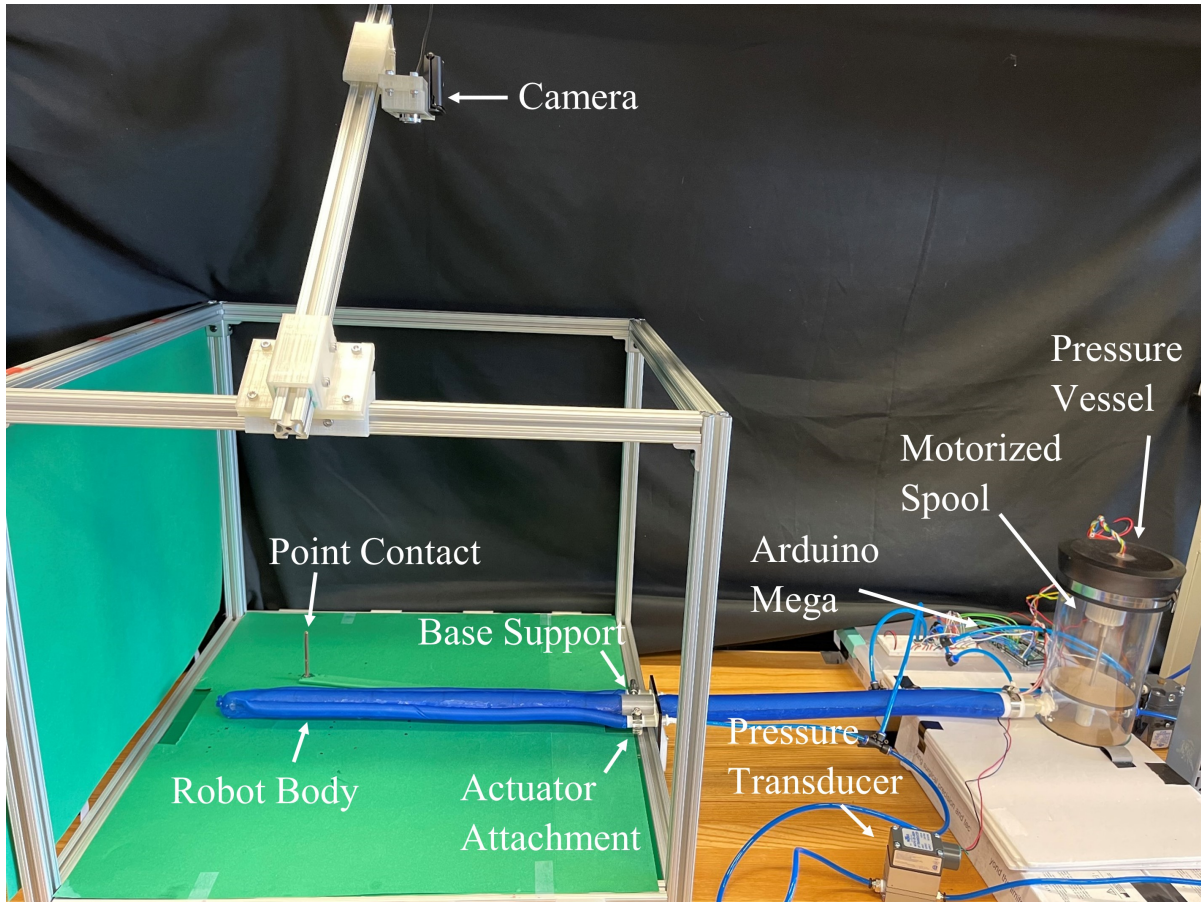
Figure 3.1. The experimental setup includes the growing robot and associated control hardware, along with the workspace into which it extends.

3.2 Single Contact

In this experiment, we look to assess how accurately we can localize an applied force of different magnitudes at various locations along the length of an unactuated robot. Specifically, we consider a low magnitude force and a high magnitude force, each applied to the following three locations along the robot body: at the tip, $\frac{3}{4}L$, and $\frac{1}{2}L$, where L is the total deployable length of the robot (see Fig. 3.2a). For this experiment, we apply 0.49 psi to the main body of the robot and collect data (curvature sensor measurements and an overhead camera image) from 5 trials for each of the six loading conditions to determine the repeatability of our results. For each loading condition, Fig. 3.2c shows the mean error between the position of the contact
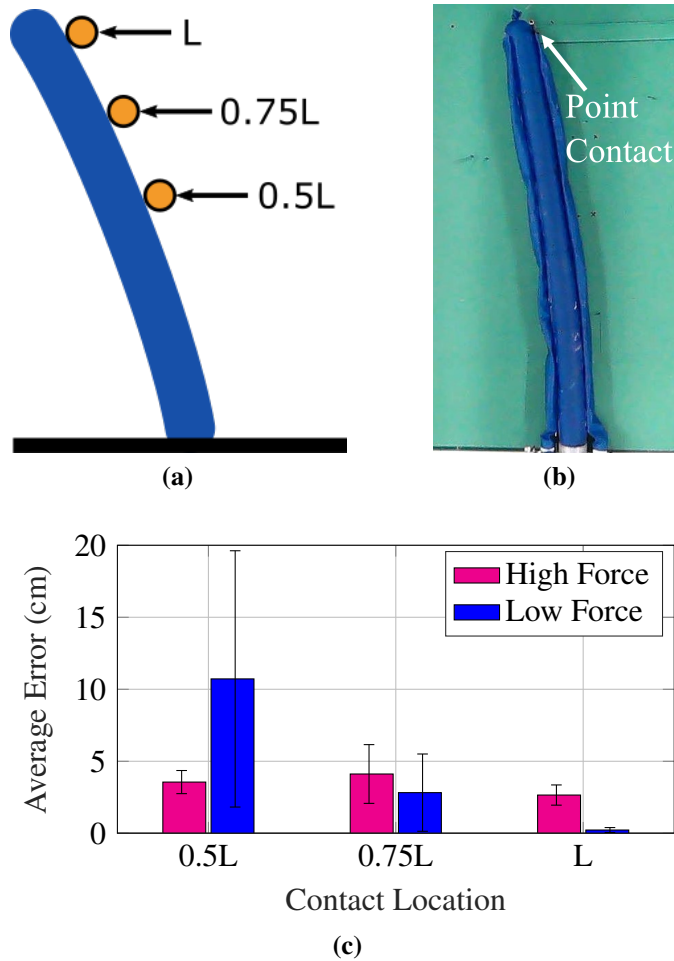
Figure 3.2. (a) Schematic illustrating the locations at which point contact forces are applied to the robot body. (b) An image of a low magnitude force applied to the growing robot at location $x = L$. (c) Contact position localization errors after optimization. The mean localization error is 3.44 ± 1.38 cm and 4.59 ± 6.78 cm for high and low magnitude forces, respectively.

force on the robot body as determined by our optimization and the position as determined by the overhead camera.

In general, high contact forces result in consistent localization error (3.44 ± 1.38 cm on average) regardless of the location at which they are applied to the robot (see Fig. 3.2c). Low magnitude contact forces, on the other hand, tend to result in increasing localization error (4.59 ± 6.78 cm on average) the further away from the tip the force is applied. This difference is likely because lower magnitude forces applied closer to the base of the robot do not induce

large changes in the curvature of the robot shape compared to those applied closer to the tip. Smaller changes in robot curvature correspond to smaller magnitude signals from each of the sensor measurements used for optimization, in turn making the optimization more sensitive to sensor noise. This causes poor optimization results where the localization overfits to noisy readings (Fig. 3.3b). High magnitude forces, however, induce consistently larger changes in robot curvature resulting in more consistent localization regardless of where the force is applied (Fig. 3.3a). We note that when the high magnitude forces are applied, the sensors at the base have significantly higher curvature measurements relative to the other sensors (see Fig. 3.3a), which can likely be attributed to the robot buckling at this location.

When the robot experiences a local moment that is greater than the maximum moment that the material can handle then the robot will buckle at the location that experiences this moment first. In the case where a point force is applied in an EB model, the moment is equivalent to the force of the contact multiplied by the distance from the base. This would then be maximized at the base of the robot which leads to the buckling occurring at this location. Any additional force after this critical moment is passed does not change the curvature along the length of the robot but instead only adds to the buckling angle at the base [2] as seen in Fig. 3.4. These higher curvature measurements from the sensors at the base do contribute some amount to the error, but the increased signal from a higher force overcomes this error contribution overall.

In order to verify this hypothesis that the buckling is a contribution to the error, we empirically fit a critical curvature that is caused by the critical moment in the robot by performing a linear regression on the sensors that do not buckle to predict the curvature that we would read at the base when applying a high force at the tip without buckling. We can employ this method of finding the critical curvature as the robot shape stays constant once the critical curvature is achieved at the base. Any moment exerted after the critical curvature is achieved at the base only changes the buckling angle. By averaging the critical curvature over 5 trials, we find a critical curvature value of $0.01108 \frac{1}{cm}$. We then perform a correction to the data read from the

(a)

(b)
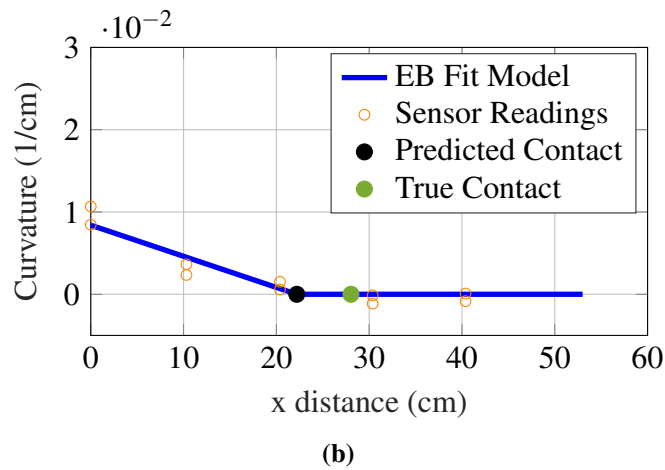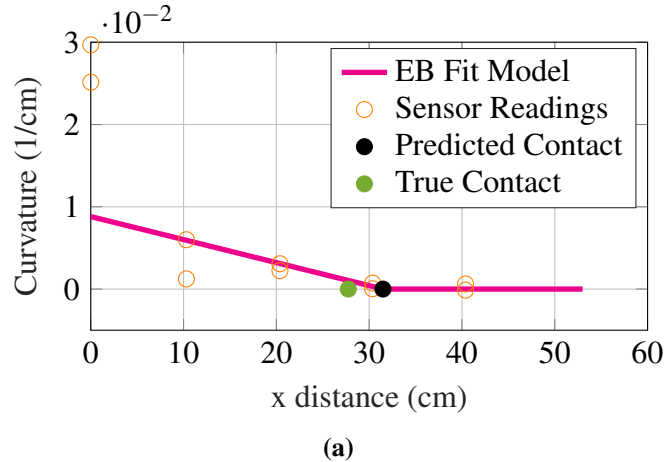
Figure 3.3. Representative example of the optimization results for (a) a high magnitude force and (b) a low magnitude force applied to the growing robot at $x = \frac{L}{2}$. Note the poor estimate of the point contact location in (b) due to the optimization over-fitting to noise in the sensors located on the more proximal half of the robot.

sensors at the base to limit their value from going over this critical curvature value.

After constraining the sensor curvature measurements from exceeding this empirically determined value, we find that the error in the high force case reduces to an overall average of 2.98 ± 1.38 cm which is a reduction of error by 13.45%. Recognizing that in a buckling conformation the robot has a critical curvature value, which is a constant, we can couple the force and position term when undergoing buckling. This allows us to perform a search in 1-dimension. To test this we used the Nelder-Mead Simplex method and received overall high force accuracy of 1.93 ± 1.04 cm.
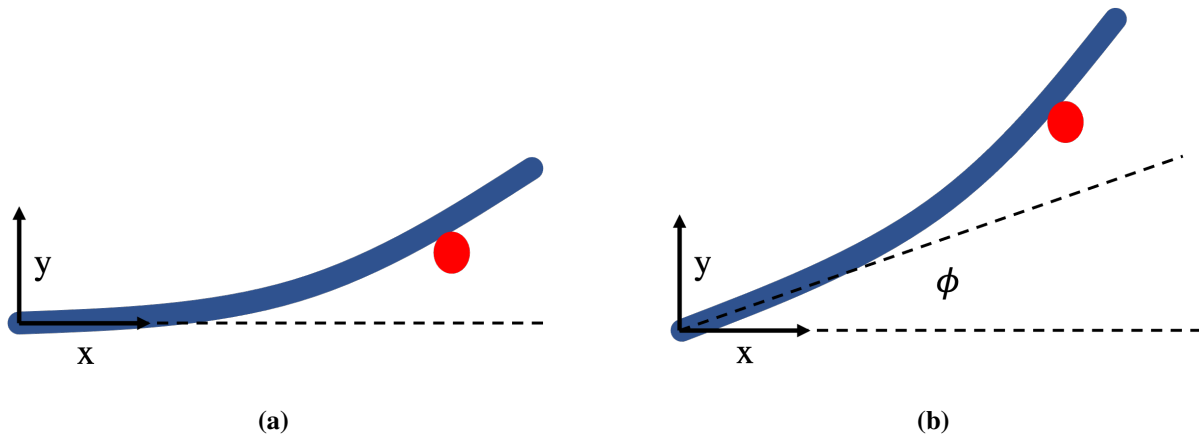
Figure 3.4. (a) A configuration of an everting robot when a critical moment is exerted to the robot body. The robot is at the maximum shape deformation for this current flexural rigidity. (b) A configuration of an everting robot when a moment greater than the critical moment is exerted to the robot body. The robot reaches the maximum shape deformation for this current flexural rigidity and begins to buckle at the base forming a bending angle Φ.

It is important to note as well that there are several reasons for the error beyond the issue of buckling. There are several assumptions that are made in the EB model that may not necessarily hold true such as the assumption that any planes perpendicular to the neutral axis remain perpendicular when shape deformation occurs. Furthermore the sensor mapping used the assumption that the deformed angles are relatively small in order to draw a relationship between the bending angle and the curvature. While we use this assumption for the sake of staying with the EB model it is noted that it may cause a higher error.

3.3   Two Point Discrimination

In the case where the growing robot is subjected to two contact forces from opposite sides, we seek to determine the distance that the forces must be separated in order to discriminate between them. To determine this distance empirically, we apply one distal contact at the tip of the robot and a second proximal contact at 6 different locations (see Fig. 3.5a), moving incrementally closer to the distal contact (39 cm, 34 cm, 29 cm, 23.5 cm, 18.5 cm, and 13 cm apart). Again, we apply 0.49 psi to the main body of the robot and collect image and sensor

30
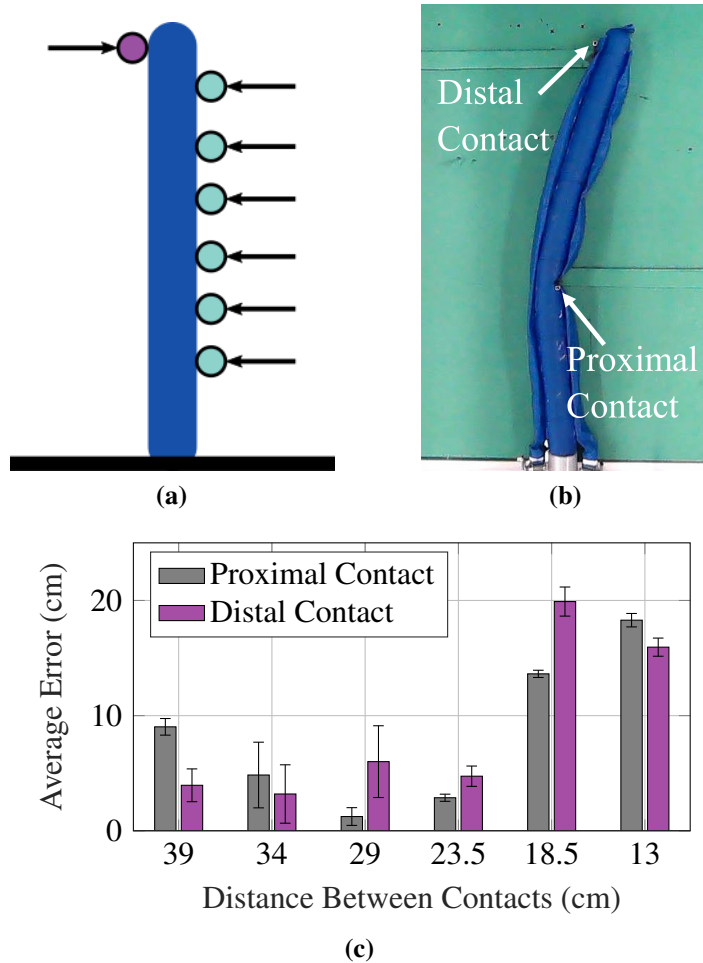
Figure 3.5. (a) Schematic illustrating the locations where point contacts are applied to the robot body. A distal contact (in purple) is held constant at the tip for all cases, while an opposing proximal contact is moved incrementally closer to the distal contact location. (b) An image of the robot when the proximal contact is 29 cm away from the distal contact. (c) Contact force localization errors after optimization.

data 5 times for each loading condition. An example image is shown in Fig. 3.5b, where the proximal contact is located 29 cm away from the distal contact.

Results in Fig. 3.5c indicate that when the point contacts are separated by a distance of 23.5 cm or greater, our proposed optimization approach can effectively determine the location of both points with an average error of 4.49 ± 2.79 cm. Below this threshold however, localization error is much higher, 16.94 ± 2.55 cm on average, suggesting that this is the minimum distance at which our method can effectively discriminate between applied point contacts.
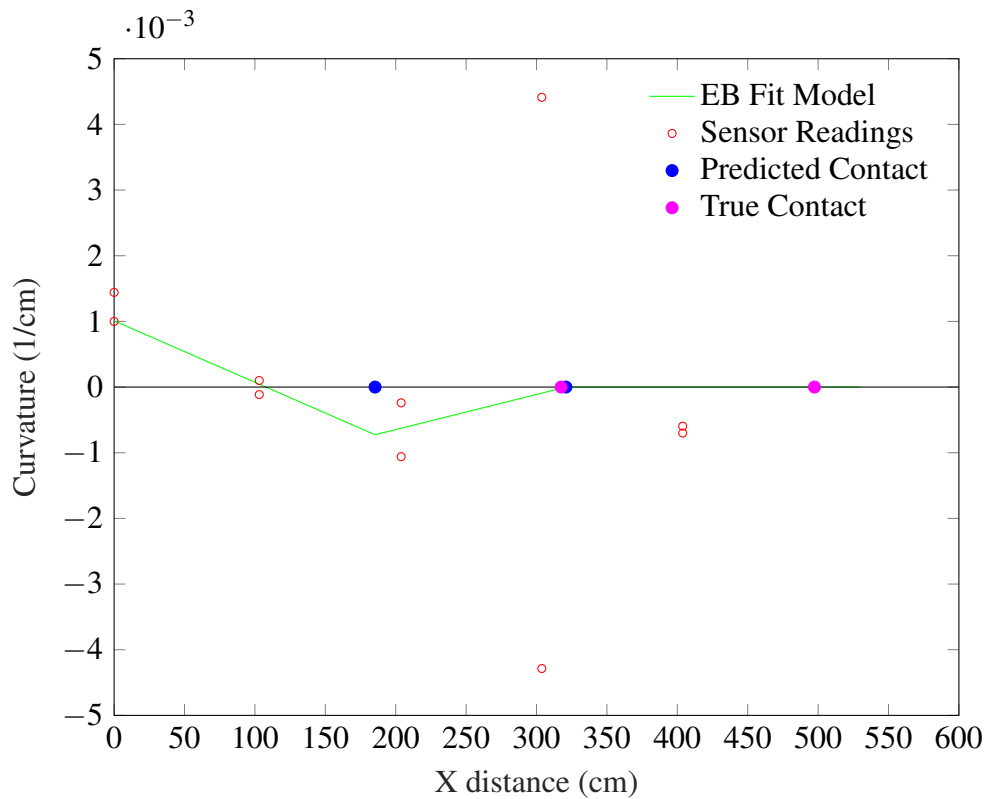
Figure 3.6. Example graph of an optimization result from the two point discrimination experiment when the two contacts are below the discrimination threshold (contacts 18.5 cm apart). The localization algorithm correctly localizes the proximal contact and finds another local minima closer to the base.

There are several reasons for the errors in our localization. As the distance between point forces, which are applied in opposite directions, decreases, the effect each force has on the overall robot shape begins to cancel out that of the other force. This phenomenon leads to smaller measured changes in curvature and increases the impact of sensor noise on the optimization result.

There are also a few important trends to point out. The error for the distal contact is minimized when the contacts are the furthest away. Furthermore, while the error appears to spike for the proximal contact prediction at 18.5 cm distance between the contacts, the optimization is actually finding the correct location for the proximal contact. The localization found two minima: one at the true proximal contact and one at a local minima found nearby the base (see Fig. 3.6). This means that the error for the proximal contact is actually minimized when the contacts are closest together. These two trends make sense when analyzing how the function is split. There are 3 essential regions of the piecewise function: the region from the base to the first contact, the region from the first contact to the second contact, and the region from the second contact to the tip of the robot. As we do not have any sensor information in the last region we can reduce the regions we are essentially looking at down to two.

When the contacts are the furthest away, the region that stretches from the base to the first contact is minimized which means we also have the fewest amount of sensors placed in that region. However the amount of sensors in the region between the contacts is maximized. This explains why we have the most accuracy for the distal contact but the lowest for the proximal contact. While our sensors are giving us good information about where the contacts are, more sensors helps to reign in the noise from the sensors allowing for a more accurate prediction. This trend continues when the contacts are the closest. The proximal contact has the lowest error while the distal contact not only has the highest error but is never found. It instead finds the local minima close to the base. These results are consistent with the reasoning for the error and also helps to explain why our method can only find contacts that are separated by 23.5 cm as this is about two sensor lengths. This essentially says that in conjunction with the forces cancelling the other out and lowering the signals of our sensors, we need 2 good sensor readings between the contacts before we can differentiate between two contacts.

3.4   Single Contact with Active Bending

Here we consider contact localization for a single contact, similar to Section 3.2, but this time for an actuated robot. When an fPAM is actuated, its length decreases, causing the robot body to wrinkle where it is attached to the actuator in order to accommodate this length change and leading to bending of the robot body. While this approach is an effective means of actuation, the wrinkling of the material of the robot body induces hard-to-model responses in the sensors attached to an activated actuator. We believe this is due to the sensor being sensitive to forces applied along its length. The changes in the material shape due to wrinkling that it is laying along causes a change in the forces applied along the length. This makes the sensor readings along the side of the actuated side hard to model without taking these changes into account. Therefore, for contact localization with actuated growing robots, we only use measurements from the curvature sensors attached to the currently inactive actuator, which is on the outer edge of the curved robot.

As in Section 3.2, we consider contacts occurring at 3 different locations along the robot body at 2 different force magnitudes. We now use the robot's own actuation to initiate contact with the obstacle, as would be the case when the robot is exploring a static, unknown environment, as shown in Fig. 3.7a. Obstacle locations are chosen such that they are barely touching the robot body at the $\frac{1}{2}L$, $\frac{3}{4}L$, and L lengths when the fPAM actuator pressure is set to 6 psi. For the low and high magnitude force loading cases, we set the actuator pressure to 10 psi and 14 psi respectively. Again, we apply 0.49 psi to the main body of the robot and collect data from 5 trials for each of the 6 loading conditions, the results of which are shown in Fig. 3.7c.

First, it is important to note that our optimization routine localizes the point contact with low error for all applied locations in the high magnitude force case. The average localization error for these trials is 4.62 ± 0.95 cm, which is only slightly higher than it was for the unactuated case. The slight increase in error is to be expected for the actuated case because the localization algorithm only uses measurements from sensors attached to the inactive actuator, making it
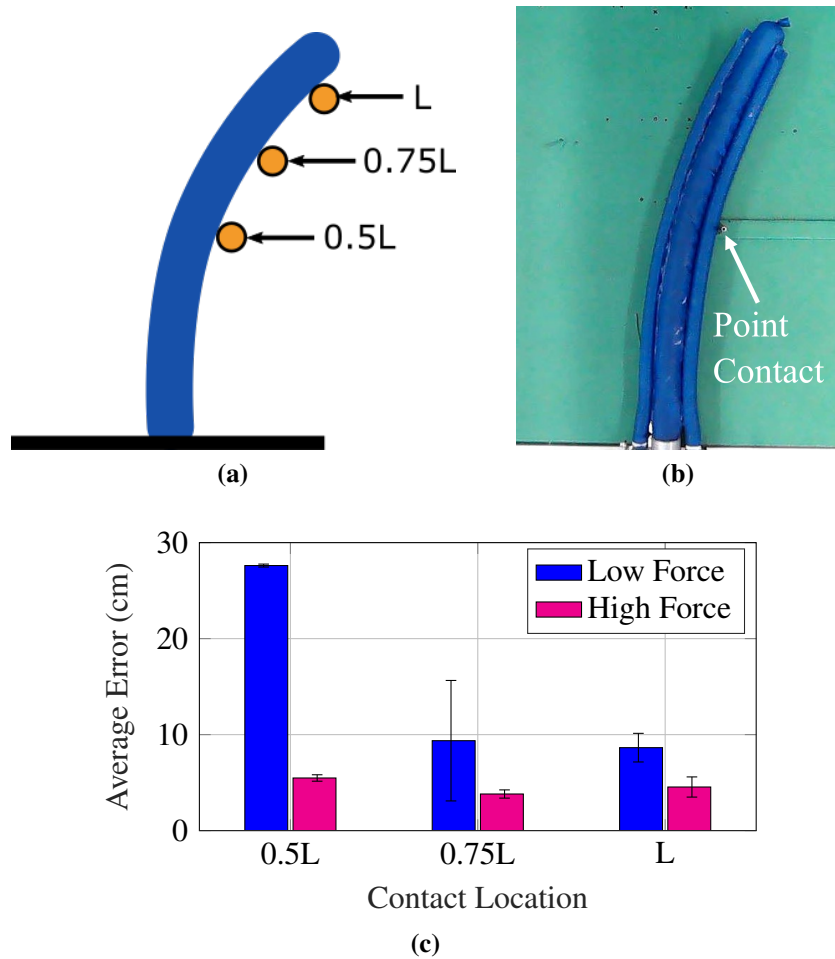
Figure 3.7. (a) Schematic illustrating the locations at which the robot collided with point contacts during active bending. (b) An image of a low force collision that occurs at location $x = \frac{L}{2}$. (c) Contact position localization errors after optimization. The mean localization error is 4.62 ± 0.95 cm and 15.21 ± 9.72 cm for the high and low magnitude forces, respectively.

more susceptible to noisy measurements from any one sensor. For the low magnitude force case, however, the average error across all trials, 15.21 ± 9.72 cm, is much higher than the error for high magnitude forces. This large error indicates that the optimization fails to effectively determine the location of the point contact at low forces, and we note that all errors for actuated contact at low forces, are higher than the maximum error from the unactuated case (Section 3.2). It is important to note as well that the amount of force applied in this experiment are lower than the forces applied in the unactuated case as it is now the robot itself pushing against the
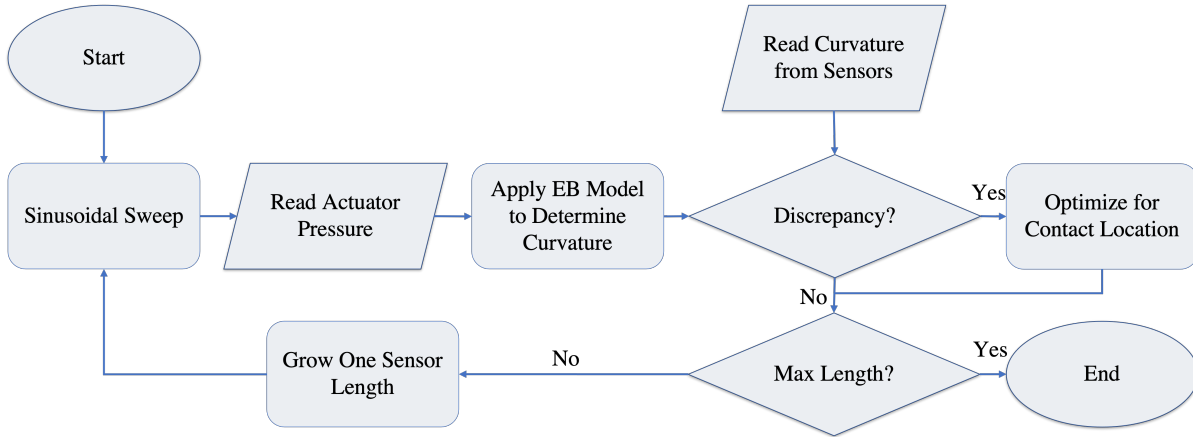
Figure 3.8. The process that the robot takes during the active and growing test for the contact localization is shown above in a flowchart. It will have the pressure in the actuator increase and decrease in a sinusoidal wave. During this process, the true pressure in the actuator will be read in and passed into the EB model to predict curvature of the robot. In parallel, the robot will read curvature from the flex sensors located on the robot body. The predicted values and sensor values will be compared for a discrepancy. If the discrepancy exceeds an empirical threshold then a contact has occurred and the algorithm will optimize over the force and location of the contact. In both cases where a contact is detected and where a contact is not detected the robot will determine if it is fully deployed. If not then the robot will grow one full sensor length (10 cm) and start over from the beginning. If the robot is fully deployed then the robot will cease operation.

object. This decreased force could also be a reason for the lower accuracy from the algorithm. Therefore, in order to determine the location of smaller forces on an actuated growing robot, more sensor measurements are necessary.

3.5   Single Contact with Active Bending and Growing

Finally, we consider a scenario where a growing robot is exploring an unknown environment. We make use of tactile perception to determine when and where an obstacle is in contact with the robot. Such information could be valuable for use in a planning algorithm (e.g. [2]) or to identify a support structure the robot could anchor itself to, as proposed in [34].

We place the unknown obstacle for the robot to encounter at the same location as that of the furthest obstacle used in Section 3.4. This location is chosen because we have found that

36

using fewer than 5 sensor measurements — which is the number of available measurements when our current robot is actuated and fully everted — for optimization, leads to less reliable localization. We discuss this current limitation of our method and robot hardware, along with future approaches that can address this challenge, in Section 4.

In order to explore the workspace, the robot first grows one sensor length and then executes a sweeping motion generated by sending a sine wave to the fPAM actuators with an amplitude equal to the maximum pressure the actuator can reliably hold (18 psi). If a contact is detected during the sweeping motion (as described in Section 2.5), we average 10 sensor measurements recorded at the peak of the sine wave to use to localize the contact. We choose to use the sensor readings collected when the actuator is at maximum pressure based on the results of Section 3.4, which show consistently lower point contact localization errors for higher forces applied to the obstacle (for a visualization of the process used in this demonstration see Fig. 3.8). To achieve robot growth, we use the minimum possible body pressure at all times, which keeps the bending stiffness of the robot low and allows it to achieve higher curvature configurations. For growth lengths from zero to $\frac{L}{2}$, growth is achieved with a pressure of 1.47 psi. For lengths $\frac{L}{2}$ to $\frac{2L}{3}$ we apply 1.01 psi, and from $\frac{2L}{3}$ to L we apply 0.49 psi. Pressure to curvature mappings are created for each body pressure following the same method detailed in Section 2.2. In order to minimize the noise from each sensor we zero the sensor readings at the end of each sweep and after each growth cycle. This is done simply by recording what the reading is at when the actuators are both inactive and subtracting that amount from the current reading.

Results of the experiment are shown in Fig. 3.9. Our method successfully identifies when the robot is in contact with an obstacle and localizes the contact with 6.82 cm error which is approximately $\frac{2}{3}$ of the length of one sensor. It is worthwhile to note that this error is higher than the average of the previous experiment in Section 3.4. This increased error is likely due to additional noise in the sensor readings caused by the rapid change in curvature of the eversion process that facilitates robot growth. When the sensors are in a straight configuration, as they
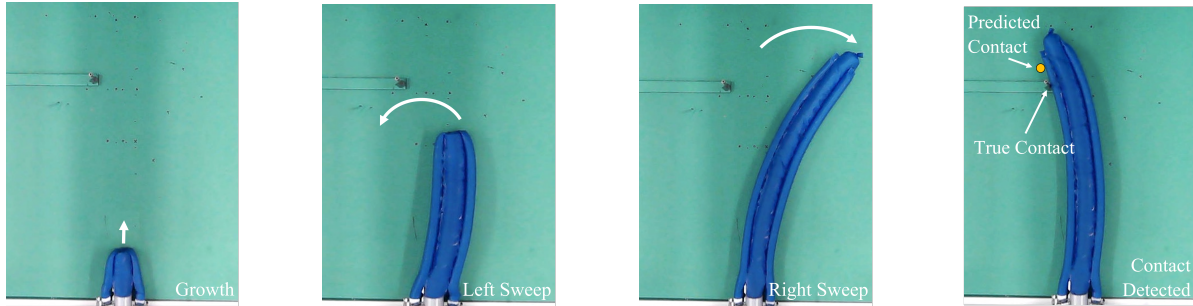
Figure 3.9. (a)-(d) Images of the robot growing and bending over time, exploring an environment containing an obstacle with unknown location. (e) The robot collides with the unknown obstacle, triggering the localization algorithm, which returns an estimate of the location of the contact force applied to the robot. The error between the predicted point contact and actual point contact is 6.82 cm.

are immediately after robot growth, they should measure zero curvature. In reality, however, we find that there is a delay in the sensor response as it goes from a highly bent configuration to a straighter configuration, which contributes to the increased localization error. This hysteresis in the sensor seems to follow an exponential decay curve that on average reaches stability within 30 seconds after eversion. To combat this we currently zero the sensors as stated after each sweep. We also do not use any readings from a sensor until a minimum of 30 seconds has passed after the sensor was everted. In the future, this delay could be modeled and compensated for so as to mitigate its impact on the optimization.

Chapter 4

Conclusions and Future Work

In this work, we have presented an approach to enable tactile sensing for growing robots. Our approach consists of a method to integrate flexible, off-the-shelf sensors into the body of a growing robot via a pouch design, along with an optimization routine to deduce the locations of point contacts on the robot body based on these sensor measurements. We experimentally characterized the error of this approach in localizing a single contact applied to both an actuated and unactuated robot and determined a threshold beyond which it was possible to distinguish between two separate contact forces applied to the unactuated robot. Finally, we showed that this method was able to detect and localize an obstacle that came into contact with a growing robot as it explored unknown surroundings.

Our system has significant benefits when it comes to localizing a contact location. It takes very little computational power to analyze, has the potential to be an online method, and can sense anywhere along the robot. This method can be implemented with an actuated or unactuated version of the robot and only needs a model for the curvature to generate a comparison between the sensor values and the model of the robot. We used an EB model, however this method can be used with any robot or model whose curvature can be solved for. It is important to note that the model chosen does contribute to the error. The assumptions of the EB model do not necessarily hold true in the curvature analysis of the robot.

As expected in the experimentation, the high force experiments resulted in higher

accuracy and lower variation in the results. This makes sense due to the increased signal received from the sensors in the high force case that can make overcoming sensor noise much easier. This trend holds true in both the actuated and unactuated case. For the two contact discrimination, our method cannot discriminate between the two contacts when they are less than 23.5 cm away, but it can still find the proximal contact. Interestingly, while it can find the proximal contact, it predicts it to be the distal contact and finds another local minima caused by sensor noise close to the base of the robot. This makes sense as the sensor that overlaps with the base experiences buckling in the two contact discrimination experiment as high forces are applied to each contact. This buckling likely causes a local minima to be present in this location, but the minima is believed to be relatively small as it is not found by the localization algorithm in any of the other cases.

The pouch employed is believed to fix most of the issues due to a stiffness mismatch by allowing the stiffness of the robot where there is a sensor attached to become similar to the stiffness where there is no sensor attached. However, this hypothesis has only been confirmed qualitatively and indirectly by evaluating the wrinkling visually and confirming that the analytical models can be used on the robot when the sensors are integrated. It is desirable to quantify the difference in stiffness for these areas where the sensors are integrated. This could be done by applying a known moment to the robot and comparing the curvature over the section with the sensor and the curvature over the section without the sensor.

There are several limitations of the method outlined in this paper. The first and most important is that our method requires 5 sensor readings before the localization can be conducted. After performing a sensitivity analysis that analyzes how sensitive our algorithm is to noise (modeled as a Gaussian distribution with a varying standard deviation) and to the number of sensors attached to the length of the robot, we determined that this actually does not arise from noise, but it instead comes from the cost function we used. Specifically, this constraint arises from using the 1-norm over the 2-norm. Our algorithm operates very similarly to a

conjugate gradient descent algorithm which is designed to minimize over a smooth, nonconvex function. When using a 2-norm cost function our sensitivity analysis shows that we should be able to achieve minimization results with as few as 2-3 sensors in the absence of noise. When performing that same sensitivity analysis over a 1-norm cost function, we are not able to perform minimization until we incorporate 5 sensors. This is because a 2-norm cost function creates a smooth, nonconvex function which is what allows our algorithm to perform as expected, but a 1-norm cost function changes the function to become a nonsmooth, nonconvex function. We believe, given these results, that the number of sensors is not a requirement to cancel out sensor noise, but rather to increase the geometric density of the sensors which essentially is smoothing our function the more sensors we place.

To solve this problem there are two methods that can be taken. The first is to use a sensor that potentially has lower noise thereby allowing the usage of a 2-norm cost function. The reasoning for the usage of a 1-norm cost function is because it makes our function much more robust to sensor noise and outliers. Using a sensor with less noise, we can use the 2-norm which will automatically smooth out our function. Instead of swapping out the sensor altogether, we could use a model or a correction term to allow for the noise to be reduced thereby also allowing the usage of a 2-norm function. The second method would be to increase the number of sensors altogether along the length of the robot. This method could be done by either stacking the sensors on one location to decrease the sensor noise by averaging over the sensor readings, or by increasing the geometric density of the sensors to allow the 1-norm function to essentially become smooth.

There is another limitation in the selection at which points we can conduct sweeps by activating the actuators. These sensors are flexible enough to evert and bend with the robot, but they are not flexible enough to be in a partially everted state when the actuator begins activating. If half of the sensor is inverted and the other half is everted when the robot begins bending with the actuator, then the sensor begins to plastically deform. Ensuring that the sensors are not

in a partially everted state is a relatively easy workaround for this issue. When more flexible sensors such as the fabric based stress sensor become more commercially available, they can be swapped out to make this limitation no longer relevant.

Furthermore, this method has clear areas where immediate improvement could be found. Using a sensor with a higher signal to noise ratio could allow for the usage of an analytical gradient to significantly increase the speed with which the algorithm performs localization. Additionally, the usage of machine learning algorithms could significantly speed up the process as well as decrease the amount of error involved. Currently the algorithm used does not take into account any historical data and any trials collected to perform the localization. With proper preprocessing of the data, fitting a machine learning model would improve this accuracy. A decision tree based regression to solve for contact location has the potential to do particularly well, especially an ensemble method. This is because the function currently used is piecewise both due to the mechanical model and the 1-norm used to perform the optimization. A decision tree by its nature is a piecewise process and potentially the usage of an ensemble of trees could allow the trees to capture the effect that a varying force could have on the sensor readings. Combining this with a mechanical model has the potential to increase the accuracy even further. The sensors could also be used to measured more than a single curvature measurement. By soldering wires in between the segmented conductors, the resistance change between each conductor could be measured thereby giving several curvature measurements to be used rather than a single measurement.

This method of localization has many use cases. It can be used when growing inside of an unknown environment in order to map obstacles to be used in planning algorithms later or to find support structures to grow around. The technology could also be combined with other types of sensing modalities such as an electromagnetic sensor to combine measurements and achieve better estimates of contact locations. Furthermore allowing for more complex analyses of the obstacles could ease the constraint of a point contact. Using a method such as that used in [4]

where the contact is estimated as a series of sliding dirac deltas could allow for more complex shapes to be analyzed. In addition the estimation could be combined with a prior estimate of the shape given an understanding of the environment such as that of a CT-scan in order to give active feedback on current location compared to predicted location in the human body. It could also be used with a tip-mounted camera to perform feedback on camera estimations of contact locations and give feedback if an obstacle was missed during growth as the camera can only look forward. Any missed contacts can be picked up by this method and allow for a backup sensing method due to its attachment method and the low amount of processing power needed to compute it.

This thesis, in part, has been submitted to the IROS 2022 Conference, Micah Bryant; Connor Watson; Tania K. Morimoto. The thesis author was the primary investigator and author of this paper.

Bibliography

[1] E. W. Hawkes, L. H. Blumenschein, J. D. Greer, and A. M. Okamura, "A soft robot that navigates its environment through growth," *Science Robotics*, vol. 2, no. 8, 2017.

[2] M. Selvaggio, L. Ramirez, N. D. Naclerio, B. Siciliano, and E. W. Hawkes, "An obstacle-interaction planning method for navigation of actuated vine robots," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*, 2020, pp. 3227–3233.

[3] A. M. Gruebele, A. C. Zerbe, M. M. Coad, A. M. Okamura, and M. R. Cutkosky, "Distributed sensor networks deployed using soft growing robots," in *2021 IEEE 4th International Conference on Soft Robotics (RoboSoft)*, 2021, pp. 66–73.

[4] V. A. Aloi and D. C. Rucker, "Estimating loads along elastic rods," in *2019 International Conference on Robotics and Automation (ICRA)*, 2019, pp. 2867–2873.

[5] D. Rus and M. Tolley, "Design, fabrication and control of soft robots," *Nature*, vol. 521, pp. 467–475, 2015.

[6] H. Lipson, "Challenges and opportunities for design, simulation, and fabrication of soft robots," *Soft Robotics*, vol. 1, no. 1, pp. 21–27, 2013.

[7] M. Runciman, A. Darzi, and G. P. Mylonas, "Soft robotics in minimally invasive surgery," *Soft Robotics*, vol. 6, no. 4, pp. 423–443, 2019.

[8] P. A. der Maur, B. Djambazi, Y. Haberthür, P. Hörmann, A. Kübler, M. Lustenberger, S. Sigrist, O. Vigen, J. Förster, F. Achermann *et al.*, "Roboa: Construction and evaluation of a steerable vine robot for search and rescue applications," in *2021 IEEE 4th International Conference on Soft Robotics (RoboSoft)*, 2021, pp. 15–20.

[9] M. M. Coad, L. H. Blumenschein, S. Cutler, J. A. Reyna Zepeda, N. D. Naclerio, H. El-Hussieny, U. Mehmood, J.-H. Ryu, E. W. Hawkes, and A. M. Okamura, "Vine robots: Design, teleoperation, and deployment for navigation and exploration," *IEEE Robotics Automation Magazine*, vol. 27, no. 3, pp. 120–132, 2020.

[10] J. Luong, P. Glick, A. Ong, M. S. deVries, S. Sandin, E. W. Hawkes, and M. T. Tolley, "Eversion and retraction of a soft robot towards the exploration of coral reefs," in *IEEE Int. Conf. on Soft Robotics*, April 2019, pp. 801–807.

[11] J. D. Greer, L. H. Blumenschein, A. M. Okamura, and E. W. Hawkes, "Obstacle-aided navigation of a soft growing robot," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 4165–4172.

[12] A. Sadeghi, A. Mondini, and B. Mazzolai, "Toward self-growing soft robots inspired by plant roots and based on additive manufacturing technologies," *Soft Robotics*, vol. 4, no. 3, pp. 211–223, 2017.

[13] H. Dehghani, C. R. Welch, A. Pourghodrat, C. A. Nelson, D. Oleynikov, P. Dasgupta, and B. S. Terry, "Design and preliminary evaluation of a self-steering, pneumatically driven colonoscopy robot," *Journal of Medical Engineering & Technology*, vol. 41, no. 3, pp. 223–236, 2017.

[14] M. Li, R. Obregon, J. J. Heit, A. Norbash, E. W. Hawkes, and T. K. Morimoto, "Vine catheter for endovascular surgery," *IEEE Transactions on Medical Robotics and Bionics*, vol. 3, no. 2, pp. 384–391, 2021.

[15] L. H. Blumenschein, M. M. Coad, D. A. Haggerty, A. M. Okamura, and E. W. Hawkes, "Design, modeling, control, and application of everting vine robots," *Frontiers in Robotics and AI*, vol. 7, 2020.

[16] J. D. Greer, T. K. Morimoto, A. M. Okamura, and E. W. Hawkes, "Series pneumatic artificial muscles (spams) and application to a soft continuum robot," in *2017 IEEE International Conference on Robotics and Automation (ICRA)*, 2017, pp. 5503–5510.

[17] E. W. Hawkes, D. L. Christensen, and A. M. Okamura, "Design and implementation of a 300Conference on Robotics and Automation (ICRA), 2016, pp. 4022–4029.

[18] N. D. Naclerio and E. W. Hawkes, "Simple, low-hysteresis, foldable, fabric pneumatic artificial muscle," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 3406–3413, 2020.

[19] S. Wang, R. Zhang, D. A. Haggerty, N. D. Naclerio, and E. W. Hawkes, "A dexterous tip-extending robot with variable-length shape-locking," *CoRR*, vol. abs/2003.09113, 2020. [Online]. Available: https://arxiv.org/abs/2003.09113

[20] Y. Chen, L. Wang, K. Galloway, I. Godage, N. Simaan, and E. Barth, "Modal-based kinematics and contact detection of soft robots," *Soft Robotics*, vol. 8, no. 3, pp. 298–308, 2021.

[21] C. Watson and T. K. Morimoto, "Permanent magnet-based localization for growing robots in medical applications," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2666–2673, 2020.

[22] S.-G. Jeong, M. M. Coad, L. H. Blumenschein, M. Luo, U. Mehmood, J. H. Kim, A. M. Okamura, and J.-H. Ryu, "A tip mount for transporting sensors and tools using soft growing robots," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2020, pp. 8781–8788.

[23] J. Braam, "In touch: plant responses to mechanical stimuli," *New Phytologist*, vol. 165, no. 2, pp. 373–389, 2005.

[24] W. Fichter, *A theory for inflated thin-wall cylindrical beams.* National Aeronautics and Space Administration, 1966, vol. 3466.

[25] A. Hooshiar, A. Sayadi, M. Jolaei, and J. Dargahi, "Accurate estimation of tip force on tendon-driven catheters using inverse cosserat rod model," in *2020 International Conference on Biomedical Innovations and Applications (BIA)*, 2020, pp. 37–40.

[26] ——, "Analytical tip force estimation on tendon-driven catheters through inverse solution of cosserat rod model," in *2021 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2021, pp. 1829–1834.

[27] S. Sadati, A. Shiva, N. Herzig, C. D. Rucker, H. Hauser, I. D. Walker, C. Bergeles, K. Althoefer, and T. Nanayakkara, "Stiffness imaging with a continuum appendage: Real-time shape and tip force estimation from base load readings," *IEEE Robotics and Automation Letters*, vol. 5, no. 2, pp. 2824–2831, 2020.

[28] Q. Qiao, G. Borghesan, J. De Schutter, and E. Vander Poorten, "Force from shape—estimating the location and magnitude of the external force on flexible instruments," *IEEE Transactions on Robotics*, vol. 37, no. 5, pp. 1826–1833, 2021.

[29] Y. Liu, C. Wang, H. Tan, and M. Wadee, "The interactive bending and wrinkling behavior of inflated beams," *Proceedings of the Royal Society A*, vol. 472, no. 2193, 2016.

[30] H. Lee, M. Glasper, X. Li, J. Nychka, J. Batcheller, H.-J. Chung, and Y. Chen, "Preparation of fabric strain sensor based on graphene for human motion monitoring," *Journal of Materials Science*, vol. 53, no. 12, pp. 9026–9033, 2018.

[31] D. M. Vogt, Y.-L. Park, and R. J. Wood, "Design and characterization of a soft multi-axis force sensor using embedded microfluidic channels," *IEEE sensors Journal*, vol. 13, no. 10, pp. 4056–4064, 2013.

[32] D. A. Haggerty, N. D. Naclerio, and E. W. Hawkes, "Characterizing environmental interactions for soft growing robots," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2019, pp. 3335–3342.

[33] S. Bektas and Y. Sisman, "The comparison of l1 and l2-norm minimization methods," *International Journal of the Physical Sciences*, vol. 5, no. 11, pp. 1721–1727, 2010.

[34] M. Wooten, C. Frazelle, I. D. Walker, A. Kapadia, and J. H. Lee, "Exploration and inspection with vine-inspired continuum robots," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, 2018, pp. 5526–5533.