Fabrication, Modeling, and Control of Plush Robots

James M. Bern[1], Grace Kumagai[2], and Stelian Coros[1]

***Abstract*— We present a class of tendon-actuated soft robots, which promise to be low-cost and accessible to non-experts. The fabrication techniques we introduce are largely based on traditional techniques for fabricating plush toys, and so we term the robots created using our approach "plush robots." A plush robot moves by driving internal winches that pull in (or let out) tendons routed through its skin. We provide a forward simulation model for predicting a plush robot's deformation behavior given some contractions of its internal winches. We also leverage this forward model for use in an interactive control scheme, in which the user provides a target pose for the robot, and optimal contractions of the robot's winches are automatically computed in real-time. We fabricate two examples to demonstrate the use of our system, and also discuss the design challenges inherent to plush robots.**

I. Introduction

Soft robotics has been shown to have great potential for producing robots which are versatile to variable tasks [1] and are inherently safe [2], both very important properties for creating robots that physically interact with humans [3].

Typical approaches to soft robotics are often inaccessible to the general public, due to their reliance on exotic fabrication techniques like silicone casting and expensive or expensive actuation methods like pneumatics [4] and shape memory alloys [5]. A benefit of using fabrics is that they promise to increase the accessibility of soft robotics. Our work relies only on standard machine sewing, DC motors, and a kit of low-tolerance 3D-printable plastic parts, and so seeks to democratize the process of soft robotic creation.

Textiles have been investigated as a material to make robots more suitable for human robot interaction scenarios including pediatric medicine [6] or eldery care [7]. However, the robots in this prior work typically consist of a traditional rigid robot covered by a fabric skin, and so do not benefit from the aforementioned advantages of soft robots. We propose a fabrication and actuation methodology which produces robots with a fabric skin and a largely soft interior throughout. For the robots in this paper, we iteratively designed fabric cutting patterns and actuator layouts, and did not make use of computational tools. However, we note that research has been done to address the problem of assisting users with the design of cutting patterns for static plush toys [8] and actuator layouts for animated plushies [9].

Our fabrication methodology promises to be low-cost and accessible to non-experts. We were inspired by existing work which proposed a soft robotic teddy bear driven by internal winches [10]. This work employs winches that are affixed to an internal rigid box, and each appear to handle a single tendon. Our mechanical design is different because our winches float within the body of the plush robot, and can each handle a large number of tendons. Because of this, the plush robots we fabricate can be squished throughout their bodies, and use far fewer actuators than if they had been designed with the previous approach. Furthermore, we propose a general approach to fabrication, modeling, and control, rather than a one-off prototype.

Our control methodology is a general approach to controlling tendon-driven soft robots, but is particularly well-suited to plush robots as it is based on an intuitive posing interface and prioritizes computational efficiency over high-precision modeling. We were inspired by previous work that uses finite element modeling to solve the inverse problem of matching tendon contractions to a target configuration of a soft robot [11]. We propose a very different treatment of cable-driven control of soft robots which explicitly computes the relationship between changes in cable contraction and the resulting changes in deformed shape of the overall robot. Furthermore, we investigate new fabrication techniques that leverage well-developed practices from the textile and plush toy industries.

We make the following contributions:

- Fabrication methodology for *plush robots*, tendon-actuated soft robots, fabricated using textiles and driven by internal winches.
- A forward model for mapping winch contractions to deformations of the robot's body.
- A control method that inverts the forward model to determine optimal winch contractions given a target deformed pose.
- Two fabricated examples of plush robots: a hexapod and a gripper.
- Discussion of several design challenges inherent to plush robots.

We present the fabrication methodology in Section II, the approach to modelling in Section III, and the control methodology in Section IV. In Section V we present two specific fabricated examples, and in Section VI we discuss the design challenges we encountered.

II. Fabrication

We present in this section a general set of fabrication techniques and mechanical devices which can be adapted for the creation of a wide variety of plush robots. An overview of the fabrication process is shown in Figure 1. The philosophy governing our fabrication methodology is that the tactile

[1]The Robotics Institute, Carnegie Mellon University, USA and Department of Computer Science, ETH Zurich, Switzerland. jamesmbern@gmail.com, scoros@gmail.com

[2]Division of Engineering Science, University of Toronto, Canada. gkumagai1@gmail.com

Fig. 1. Overview of fabrication procedure showing how a laser cut pattern (a) is sewn, inverted, and stuffed (b), and finally rigged with tendons (c). In (b) we show a breakdown of the components inside of the hexapod, with the stuffing from one leg removed to visualize its volume. We also include the two tools required for assembling the hexapod, which are a standard screw driver and a blunt eye needle. We note that power supply and control boards are external to the robot, and so not included in this figure. In (c) we show a close-up of the final leg, emphasize the routing path in bold yellow lines in (c.i), and show the actual tendons in (c.ii).

experience of a plush robots should match that of a plush toy. We break this into two main design goals, which are that plush robots should be 1) soft to the touch, and 2) deform easily when squeezed. To accomplish the first goal we fabricate the skin of our robots out of fabric commonly used in plush toys. To accomplish the second goal we bury rigid mechanical components deep within a plush filled body, and transmit force to the skin via tendons.

A secondary goal of our work was to develop a fabrication methodology which was easily accessible to non-experts. For this reason, the fabrication methodology was chosen to be low-cost, and relies on easily-accessible cut-and-sew techniques. The mechanical devices we use consist of off-the-shelf components and 3D-printed parts, and so should also be accessible to the novice user.

A. Body

We laser cut the skin of our robots out of acrylic craft felt. In the same pass, we also cut lines of small holes where we will route tendons. Standard zippers are sewn onto the skin for later use as access ports for installing motors, filling with stuffing, and routing electrical cables to off-board power and controller boards. The body is then sewn together with all-purpose thread, and inverted to hide seams and achieve the appearance of a traditional plush toy. The robot is then stuffed with standard polyester fiber filling.

B. Tendons

We use braided fishing line for tendons. Monofilament was also tested, but was observed to become plastically deformed after being secured and then adjusted. The tendons will be routed through small holes in the fabric skin using a large-eye blunt needle.

C. Internal Winch

A winch consists of a geared DC motor, a spool attached to the shaft of the motor, and a housing attached to the body of the motor. Tendons are tied to the main cylinder of the spool and passed through channels in the housing. The entire winch assembly floats within the body of the plush robot.

In most cases we make use of a general purpose housing with many channels through which we can route tendon (Figure 2). This general purpose housing can accommodate a single motor. In certain cases we can reduce mechanical

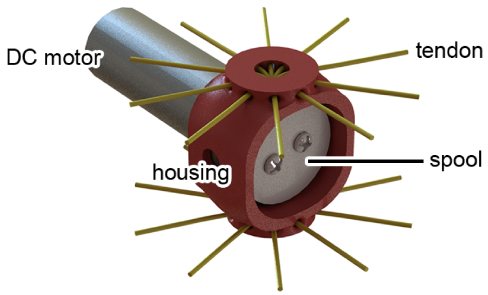

Fig. 2. Winch assembled with general purpose housing. In practice the winch can be assembled with an appropriate subset of the possible tendons in yellow.

complexity by designing a single *compound housing* that can accomodate multiple spools. We do this for our hexapod example (Figure 3).

D. Stoppers

Some means is required to prevent a tendon from being pulled through the final waypoint in its routing path, while at the same time fixing the total length of the tendon.

Traditional buttons were considered, however tying the tendon to them is cumbersome, and leads to a system that cannot be easily adjusted. Adjustable tendon lengths are key for tuning the system to match the simulated model. In the case of our hexapod example this can be the difference between the hexapod standing and falling over.

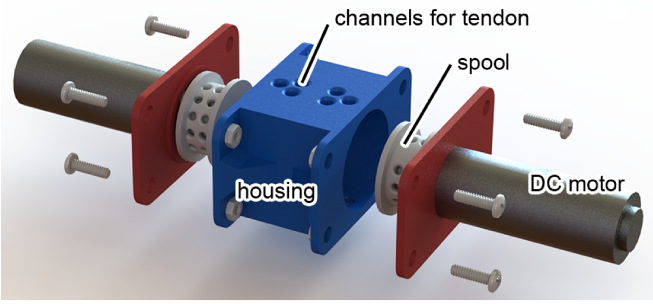

Fig. 3. Exploded view of overall internal winch for the hexapod.

We suggest the use of an adjustable clamping *stopper*, and provide here a model which can be 3D-printed and uses only a single screw (Figure 4). For ease of tuning we assemble our

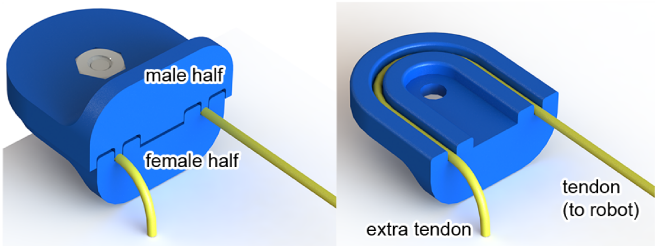

Fig. 4. Rendering of general-purpose stopper (left), and just the female half for reference (right).

robots with the stoppers external to the robot skin. However, if desired it is entirely possible to have all stoppers contained within the skin of the plush robot.

E. Tendon sheaths

For certain designs, it is desirable to avoid compressing certain regions of the plush robot. For example, we would prefer to avoid compressing the region of our hexapod in between the central internal winch and the legs. We can accomplish this by sheathing the tendon in these regions with PTFE tubing (Figure 5). The use of tendon sheaths also prevent cables from rubbing against each other as they exit the housing. We pay a price for using tendon sheaths in the form of tension losses, as detailed in [12], though in practice are still able to acheive the desired deformations for our hexapod example (Figure 8).

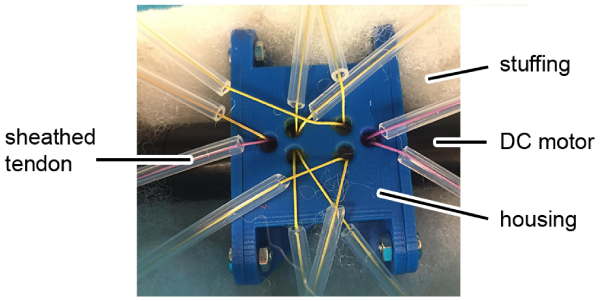

Fig. 5. Bowden cables used between the internal winch and the legs for the hexapod.

III. Modeling

In this section we present a forward simulation model to capture the deformation behavior of plush robots. This is useful for testing out designs without the need to actually fabricate them. We will make use of this same forward model in Section IV, where leverage it for our control purposes.

Our simulation is built around the assumptions of quasi-static motions. We seek configurations of the plush robot where tension forces running through the tendons are balanced by the internal deformations forces of the robot's soft body. Such a configuration minimizes the overall deformation energy of the robot. Given some contractions $\boldsymbol{\alpha^c}$ of the robot's winches (a vector encoding the length of tendon reeled in or let out by each winch), our goal is to find the resulting statically stable configuration $\boldsymbol{x}^*$ of the robot. We accomplish this by minimizing the overall energy E of the system, as written below.

$$\boldsymbol{x}^* = \arg\min_{\boldsymbol{x}} E(\boldsymbol{x}; \boldsymbol{\alpha^c}) \tag{1}$$

In the remainder of this section we will build the framework required to compute this energy. We will model the body of our robot as a finite element mesh, the elements of which contribute E_{plush} to the total energy of the system. To eliminate the rigid body modes of the robot's motion we will add virtual pins to our simulation, which contribute E_{pin} to the total energy of the system. We will model the tendons as unilateral springs, which contribute E_{tendons} to the total energy. The total energy of the system E is then written below.

$$E = E_{\text{plush}} + E_{\text{pins}} + E_{\text{tendons}} \tag{2}$$

A. Body

We discretize the continuous body of the robot into triangular elements for a 2D simulation, though the overall approach we present in this paper is general and would support a 3D simulation model as well. We take the configuration of the robot at assembly (with no tendons active) to be the *rest pose* $\boldsymbol{X}$ of the underlying finite element mesh. We will denote any other statically pose of the robot as $\boldsymbol{x} = \boldsymbol{x}(\boldsymbol{\alpha^c})$. We note that $\boldsymbol{X}$ and $\boldsymbol{x}$ are vector quantities that encode the positions of the nodes of the simulation mesh. We will often refer to $\boldsymbol{x}$ as the *pose* of the robot.

We model the elastic behavior of plush robots using linear finite elements with a non-linear material model. For each element e of the simulation mesh, we first compute the deformation gradient as $\mathcal{F} = \partial \boldsymbol{x}^e / \partial \boldsymbol{X}^e = \boldsymbol{d}\boldsymbol{D}^{-1}$, where $\boldsymbol{d}$ is a matrix whose columns store an element's edge vectors: $\boldsymbol{d}_i = \boldsymbol{x}_i^e - \boldsymbol{x}_0^e$ with $\boldsymbol{x}_j^e$ denoting the world coordinates of the j-th node of element e. $\boldsymbol{D}$ is similarly defined using rest configuration quantities. The deformation energy density of the element is then defined using a compressible Neo-Hookean material model:

$$\Psi(\boldsymbol{x}, \boldsymbol{X}) = \frac{\mu}{2}\text{tr}(\mathcal{F}^T\mathcal{F} - I) - \mu \ln J + \frac{\kappa}{2}(\ln J)^2, \tag{3}$$

where μ and κ are material parameters, I is the identity matrix and $J = \det(\mathcal{F})$. The elastic energy stored by the

element is obtained by integrating Equation (3) over its domain, which is trivial given the assumption that deformation gradients are constant across each element. The energy term E_{plush} is obtained by summing up the individual contributions of all elements in the simulation mesh.

To remove the rigid body modes of the robot's motion, we attach *pins* to nodes, which we model as zero-length springs with high spring constant. They contribute E_{pin} to the total energy of the system.

B. Tendons and Winches

We consider *tendons* as piecewise linear curves connecting nodes in the mesh, which we refer to as *via points*. We call the endpoints of the linear curves the tendon's *endpoints*. We refer to the overall sequence of via points as the tendon's *routing path*. In our model via points are assumed to be frictionless, so all following discussions of tendon energy depend only on the tendon's total length (from winch to stopper).

We model each tendon as a unilateral spring with high spring constant. Rather than e.g. using hard constraints to enforce a current length for a tendon, we will let the deformation energy of the tendon serve as a soft constraint on its length. The *current rest length* α of a tendon (in the underlying spring model) will therefore correspond to the length of that tendon in the physical assembly, measured from winch to stopper. We can then model the relevant winch reeling in (or letting out) of the tendon as a change of it rest length α.

For any pose of the mesh $\boldsymbol{x}$ we can compute a tendon's *measured length* $\ell = \ell(\boldsymbol{x})$, *deformation* $\Gamma = \ell - \alpha$, and unilateral *strain energy* $U = U(\Gamma)$ defined as a smooth piecewise polynomial

$$U(\Gamma) = \begin{cases} 0 & \Gamma \leq -\varepsilon \\ \frac{K}{6\varepsilon}\Gamma^3 + \frac{K}{2}\Gamma^2 + \frac{K\varepsilon}{2}\Gamma + \frac{K\varepsilon^2}{6} & -\varepsilon \leq \Gamma < \varepsilon \\ K\Gamma^2 + \frac{K\varepsilon^2}{3} & \text{otherwise} \end{cases} \quad (4)$$

where K is a large spring constant, and ε is a small constant determining the domain of the cubic component, which smoothly interpolates between 0 (on the left, corresponding to no force when slack), and a quadratic (on the right, modeling Hooke's law). A tendon's *tension* is equal to the derivative of its strain energy $\frac{\partial U}{\partial \Gamma}$. We denote the net contribution of all tendon strain energies to the total energy of overall system as E_{tendons}.

Formally, a *winch* is just a set of tendons with a common endpoint. A winch acts by contracting all of its tendons by the same absolute amount, which we will refer to as the winch's *contraction* α^c. A positive change in contraction corresponds to reeling in the tendons, and a negative change in contraction corresponds to letting out slack.

We find it convenient to write a tendon's current rest length α as the difference of the tendon's *assembly length* α^0 and the contraction of the tendon's winch α^c.

$$\alpha = \alpha^0 - \alpha^c \quad (5)$$

A tendon's assembly length never changes, and is defined as the measured length of the tendon in the robot's rest pose. At assembly, all tendons are assumed to store zero energy and have zero slack. The robot sets the contractions of its winches, which in turn propagate their contractions into the rest lengths of their tendons. The new rest lengths of the tendons causes the robot to deform into a new statically stable pose. We efficiently compute this statically stable pose by minimizing total energy E using Newton's method, given that the first and second derivatives $\frac{\partial E}{\partial \boldsymbol{x}}, \frac{\partial^2 E}{\partial \boldsymbol{x}^2}$ of the individual terms can be readily computed.

IV. Control

We now describe a control method that builds on the simulation model presented in the previous section. Our key insight is that the quasi-static assumption we make about the robot's motions can be used to efficiently establish a relationship between changes in the winch contractions and the change they induce in the robot's deformed shape. Building on this technical component, our aim is to provide an intuitive approach to generating purposeful motions for plush robots, as illustrated in Figure 6. Through a graphical user interface, a user specifies a desired pose for the robot by dragging simulation nodes to different target positions. Based on this input, our system computes optimal winch contractions $\boldsymbol{\alpha^{c*}}$ that bring the statically stable pose $\boldsymbol{x}$ of the simulated robot as close as possible to the target pose $\boldsymbol{x'}$. This input modality is similar to inverse-kinematics schemes used to control traditional robots, and is therefore particularly intuitive to provide. We solve the underlying control problem by minimizing the following objective as a function of winch contractions:

$$\boldsymbol{\alpha^{c*}} = \arg\min_{\boldsymbol{\alpha^c}} \mathcal{O}(\boldsymbol{\alpha^c}; \boldsymbol{x'}) \quad (6)$$

The winch contractions computed by solving this optimization problem are passed to the real-world robot in real-time through a serial connection. An interesting observation

Fig. 6. A user poses the plush robot through an intuitive interactive control system.

of our control approach is that we do not need to wait until an optimal solution is found: the intermediate solutions obtained with iterative numerical optimization methods make consistent progress towards solving the control problem, and therefore provide reasonable trajectories from the initial configuration of the robot to the user-specified target.

In the remainder of this section we formalize the interaction modes available to the user, further develop the

optimization objective $\mathcal{O}$, and describe an efficient approach to deriving its gradient.

A. Interaction Modes

A user is presented with a simulation of the plush robot, always at some statically-stable position $\boldsymbol{x}$. Our control framework presents three main modes of interaction. First, the user can explicitly specify the contraction of any winch. In this case, the forward simulation model is used to predict the deformation of the plush robot, which can then be compared against the behavior of the physical prototype. Second, the user can specify a target pose $\boldsymbol{x}'$ for the plush robot in an intuitive fashion, by dragging simulation nodes to target positions using the mouse cursor. Last, the user can freeze (or unfreeze) the activation of any given winch. This is useful in situations where the deformation of a portion of the robot should be held constant while the rest of the robot moves (e.g. independently controlling picking up or letting go actions from the overall movement of a robot arm, as shown in Figure 7).

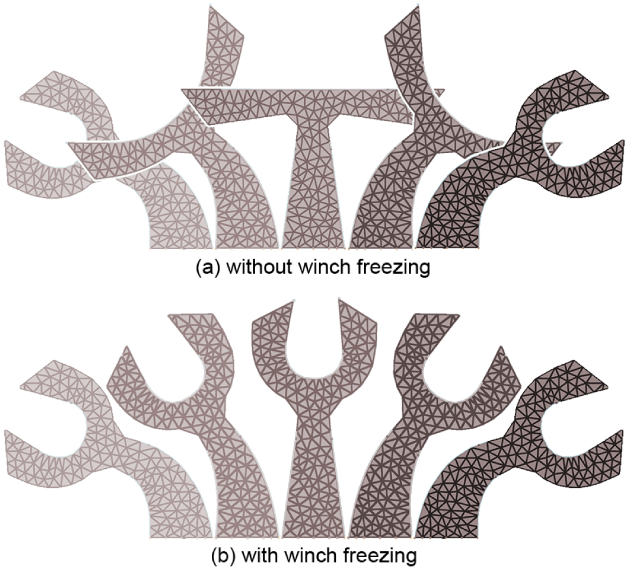

Fig. 7. Comparison of user-generated trajectories without freezing of the top winch (top) and with freezing of the top winch (bottom). Without freezing the gripper would lose its grasp, violating the intent of the user.

B. Optimization Objective

The optimization objective was chosen to penalize deviation of the statically stable pose $\boldsymbol{x}$ from the user-specified target pose $\boldsymbol{x}'$. Nodes without explicitly specified positions are masked out using a diagonal matrix $\boldsymbol{Q}$. Our system automatically computes winch activations in real time to minimize the objective

$$\mathcal{O}(\boldsymbol{x}) = \frac{1}{2}(\boldsymbol{x} - \boldsymbol{x}')^T \boldsymbol{Q}(\boldsymbol{x} - \boldsymbol{x}') \tag{7}$$

which brings the simulated mesh as close as possible to these target positions.

C. Gradient

The goal of our optimization-based controller is to solve for winch contractions $\boldsymbol{\alpha^c}^*$ that minimize the objective $\mathcal{O}$ defined in Equation (7). We begin by expanding the relevant gradient of the objective using the chain rule:

$$\frac{\partial \mathcal{O}}{\partial \boldsymbol{\alpha^c}} = \frac{\partial \mathcal{O}}{\partial \boldsymbol{x}} \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\alpha^c}} \tag{8}$$

The Jacobian $\frac{\partial \mathcal{O}}{\partial \boldsymbol{x}}$ is straight-forward to compute analytically.

We now seek an expression the Jacobian $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\alpha^c}}$, which maps changes in contractions $\boldsymbol{\alpha^c}$ to changes in positions $\boldsymbol{x}$. Unfortunately, the relationship between $\boldsymbol{\alpha^c}$ and $\boldsymbol{x}$ does not have a closed-form solution, as $\boldsymbol{x}$ is found by numerically solving an energy minimization problem (Eq. 1). We opt to perform sensitivity analysis on the nodal forces. The tensions $\boldsymbol{\tau}$ are distributed to the nodes of the finite element mesh via the relationship

$$\boldsymbol{F}_{\text{tendons}} = \boldsymbol{A}\boldsymbol{\tau} \tag{9}$$

where the matrix $\boldsymbol{A}$ encodes the tendon routings. We expand in terms of the tensions $\boldsymbol{\tau}$

$$\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\alpha^c}} = \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\tau}} \frac{\partial \boldsymbol{\tau}}{\partial \boldsymbol{\alpha^c}} \tag{10}$$

and solve for $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\tau}}$ using sensitivity analysis.

The implicit function theorem allows us to locally consider nodal positions as functions of tensions as $\boldsymbol{x}(\boldsymbol{\tau})$. Our assumption of quasi-statics means that a change in tensions $\boldsymbol{\tau}$ induces a corresponding change in position $\boldsymbol{x}$ such that force equilibrium is maintained.

$$\boldsymbol{F}(\boldsymbol{\tau}, \boldsymbol{x}) = 0 \tag{11}$$

This means that the total derivative of nodal forces with respect to tensions vanishes.

$$\frac{\mathrm{d}\boldsymbol{F}}{\mathrm{d}\boldsymbol{\tau}} = \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{\tau}} + \frac{\partial \boldsymbol{F}}{\partial \boldsymbol{x}} \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\tau}} = 0 \tag{12}$$

Note that $\frac{\partial \boldsymbol{F}}{\partial \boldsymbol{x}} = -\frac{\partial^2 E}{\partial \boldsymbol{x}^2}$ is the negative Hessian of the total energy of the system, and $\frac{\partial \boldsymbol{F}}{\partial \boldsymbol{\tau}}$ is equal to the matrix $\boldsymbol{A}$ since $\boldsymbol{F}_{\text{tendons}}$ is the only force term that depends on tensions. Substituting these relationships into Equation (12) and rearranging yields the following linear system

$$\boldsymbol{A} = \frac{\partial^2 E}{\partial \boldsymbol{x}^2} \frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\tau}} \tag{13}$$

that we solve numerically for the Jacobian of interest $\frac{\partial \boldsymbol{x}}{\partial \boldsymbol{\tau}}$. The Hessian $\frac{\partial^2 E}{\partial \boldsymbol{x}^2}$ is the same quantity necessary to compute–using Newton's method–the statically stable pose $\boldsymbol{x}$, and can therefore be reused to efficiently compute the gradient we need for control. The remaining Jacobian $\frac{\partial \boldsymbol{\tau}}{\partial \boldsymbol{\alpha^c}}$ is straightforward to compute analytically.

D. Actuation Bounds

There is one more issue that must be addressed before we can run a gradient based method to minimize Equation (7). We need some means for putting bounds on the values of $\boldsymbol{\alpha^c}$ explored by our optimizer. This is because for any given winch, a sufficiently large value of α^c corresponds to

setting a negative tendon rest length, which is non-physical. We could either employ a constrained optimization method, but to keep out optimization method simple we employ a reparametrization approach. For each winch we introduce a new variable ξ and a function $f(\xi)$ such that $f(\xi) \leq \alpha^{\max}$, where $\alpha^{\max}$ is our desired upper bound on the contraction of the winch. We reparameterize $\alpha^c(\xi) = f(\xi)$. With this modification our optimization will now be in terms of the new variables $\boldsymbol{\xi}$, and our gradient is expanded below.

$$\frac{\partial \mathcal{O}}{\partial \boldsymbol{\xi}} = \frac{\partial \mathcal{O}}{\partial \boldsymbol{\alpha^c}} \frac{\partial \boldsymbol{\alpha^c}}{\partial \boldsymbol{\xi}} \tag{14}$$

We can then use this gradient as part of an unconstrained optimization on $\boldsymbol{\xi}$. The specific function f we use in our implementation is a modified version of the function defined in Equation (4), negated and translated by $\alpha^{\max}$.

Once the gradient of the control function is computed, we use a standard line search method to determine how much the values of the winch contractions should change. With a new set of winch activations available, we compute the associated statically stable robot configuration (Eq. 1), and the control step (Eq. 7) repeats iteratively.

V. Examples

We fabricate two examples to demonstrate the use and versatility of our system, a hexapod and a gripper. The hexapod demonstrates the ability of plush robots to achieve large, controlled deformations (in this case standing up on three and six legs). The gripper demonstrates how our interactive control scheme can be used to make a plush robot perform a classic task (in this case gripping, moving, and releasing and object).

A. Hexapod

We design and fabricate a hexapod, as shown in Figure 8.

Actuation System: The actuation system of the hexapod uses two motors (each with its own spool) in a single compound housing. Each motor is responsible for contracting three of the hexapod's legs, and two tendons run down each leg of the hexapod. A total of 12 tendons are used for the hexapod, with 6 tendons reeled in simultaneously by each of the two motors.

Leg Design: To help avoid entering a non-recoverable configuration (wherein the hexapod is sitting on its own legs) we use a routing path that contracts each leg into an "S" shape, rather than the "C" shape typically achieved with longitudinal tensile actuators [3]. This is illustrated in Figure 9. In its contracted pose, each leg manifests a "foot", the geometry of which is encoded by the tendon routing path.

B. Gripper

We design and fabricate a gripper, as shown in Figure 10. The actuation system of the hexapod uses three motors, one responsible for bending to the left, the other for bending to the right, and the third for actually performing the gripping behavior.

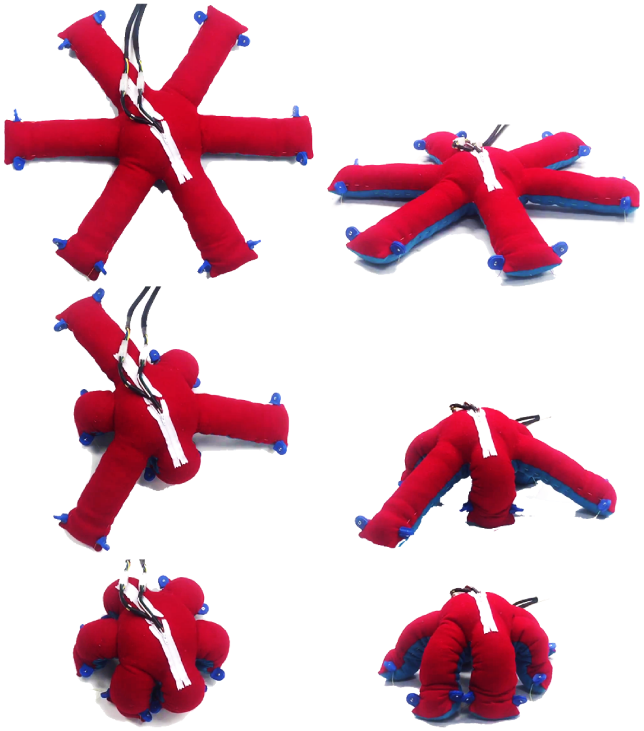

Fig. 8. Fabricated hexapod example with no legs contracted (top), one set of legs contracted (middle), and both sets of legs contraccted (bottom), shown from a top view (left) and a side view (right).

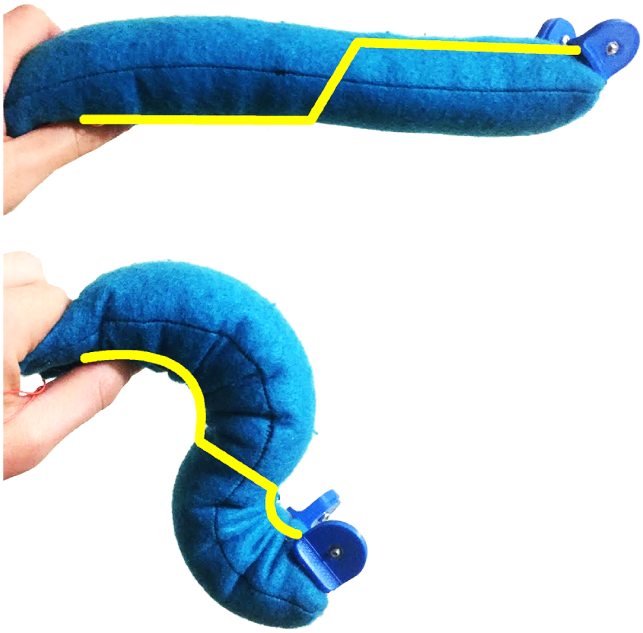

Fig. 9. Prototype of leg design for the hexapod, shown at rest (top) and contracted (bottom), with the tendon routing path indicated in bold yellow.

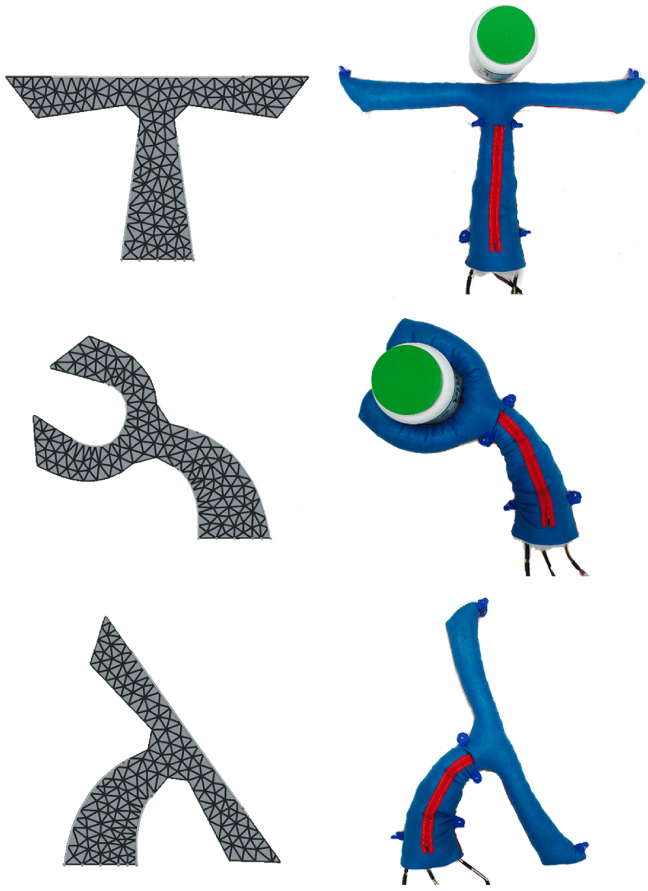

Fig. 10. Gripper example in multiple poses with corresponding pose in simulation on the left.

VI. Discussion

A. Hysteresis

One of the fundamental limitations of our fabrication methodology is that we must rely on the elasticity of the plush stuffing to restore the plush robot back to its original shape. We observe hysteresis when a robot with all cables uncontracted still does not return to its rest pose. We show an instance of this for our gripper example in Figure 11.

We identify two sources of this hysterises. First, the force of friction between the tendons and the skin may be too large for the elastic force to overcome Second, the stuffing itself may rearranged into a configuration that no longer matches our model (and has a different statically stable pose). This is because our stuffing is in the form of amorphous clumps of fibers, rather than e.g. a single chunk of elastomer. This is particularly prevalent when less stuffing is used.

Sometimes we can rely on gravity to serve as an additional restoring force, as is the case with the hexapod example. Another potential solution we investigated is to use antagonistic pair of cables. Following the contraction of one cable, another cable can be used to supply a restoring force. However, we note that friction can still build up in the system, as was observed in our gripper example.

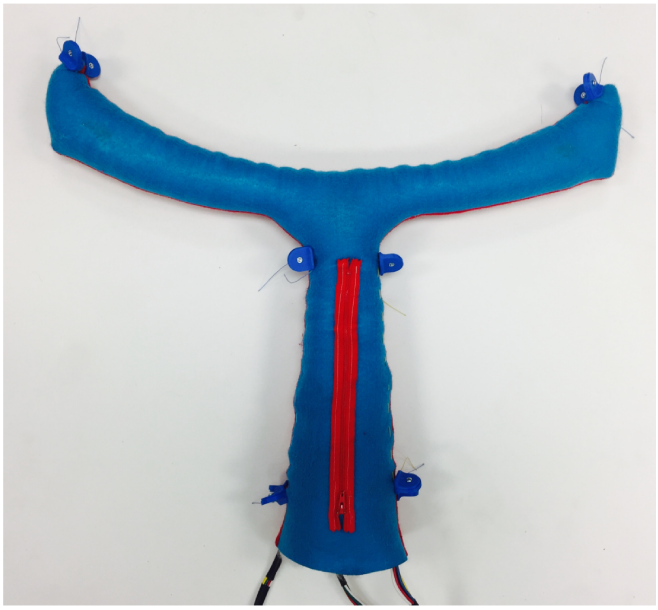

Fig. 11. Illustration of hysteresis in our gripper example. Note that even though all winches are slack, the top of the "T" is no longer a straight line, and the body of the "T" exhibits rippling (and will be physically shorter).

B. Stuffing Density

One obvious solution to hysteresis is to increase the stuffing density of the robot, and thereby increase the elastic restoring force. We observe this to generally be an effective strategy in practice, however we note two potential downsides of this technique. First, increasing the stuffing density will increase the tendon tensile force required to reach a target pose, and this increased tension will also increase the friction present in the system. I.e., in trying to fight against friction, we will also increase the friction we have to fight against. Second, increasing the stuffing density will make the plush robot stiffer, and in the limit, the tactile experience of the plush robot will cease to resemble that of a plush toy.

VII. Conclusion

We presented plush robots, a class of tendon-actuated soft robots fabricated largely from textiles. The fabrication techniques we introduced promise to be inexpensive and help democratize the process of creating soft robots. We provided a forward simulation model to predict the deformation behavior of a plush robot without needing to fabricate it. We then described how to invert this forward model for use in an interactive control scheme. We fabricated a hexapod and a gripper to demonstrate the use of our system. We discussed the sources of hysteresis in our prototypes, and explained how hysteresis can be mitigated.

Future work will focus on incorporating friction into our model, as well as contacts. We will also move the simulation into 3D, so we can begin to investigate spatial motions such as twisting. We will also continue to develop more involved examples, and investigate different types of robots that can walk, crawl, and roll.

ACKNOWLEDGMENT

We would like to thank Timothy Mueller-Sim, Merritt Jenkins, Frances Tso, and Kai-Hung Chang for their help.

REFERENCES

[1] S. Hirose and Y. Umetani, "The development of soft gripper for the versatile robot hand," *Mechanism and machine theory*, vol. 13, no. 3, pp. 351–359, 1978.

[2] S. Sanan, J. B. Moidel, and C. G. Atkeson, "Robots with inflatable links," in *Intelligent Robots and Systems, 2009. IROS 2009. IEEE/RSJ International Conference on*. IEEE, 2009, pp. 4331–4336.

[3] D. Rus and M. T. Tolley, "Design, fabrication and control of soft robots," *Nature*, vol. 521, no. 7553, pp. 467–475, 2015.

[4] M. T. Tolley, R. F. Shepherd, B. Mosadegh, K. C. Galloway, M. Wehner, M. Karpelson, R. J. Wood, and G. M. Whitesides, "A resilient, untethered soft robot," *Soft Robotics*, vol. 1, no. 3, pp. 213–223, 2014.

[5] S. Seok, C. D. Onal, K.-J. Cho, R. J. Wood, D. Rus, and S. Kim, "Meshworm: a peristaltic soft robot with antagonistic nickel titanium coil actuators," *IEEE/ASME Transactions on mechatronics*, vol. 18, no. 5, pp. 1485–1497, 2013.

[6] W. D. Stiehl, C. Breazeal, K.-H. Han, J. Lieberman, L. Lalla, A. Maymin, J. Salinas, D. Fuentes, R. Toscano, C. H. Tong, *et al.*, "The huggable: a therapeutic robotic companion for relational, affective touch," in *ACM SIGGRAPH 2006 emerging technologies*. ACM, 2006, p. 15.

[7] C. D. Kidd, W. Taggart, and S. Turkle, "A sociable robot to encourage social interaction among the elderly," in *Robotics and Automation, 2006. ICRA 2006. Proceedings 2006 IEEE International Conference on*. IEEE, 2006, pp. 3972–3976.

[8] Y. Mori and T. Igarashi, "Plushie: an interactive design system for plush toys," in *ACM Transactions on Graphics (TOG)*, vol. 26, no. 3. ACM, 2007, p. 45.

[9] J. M. Bern, K.-H. Chang, and S. Coros, "Interactive design of animated plushies," in *ACM Transactions on Graphics (TOG)*, vol. 36, no. 4. ACM, 2017.

[10] Y. Yamashita, T. Ishikawa, H. Mitake, Y. Takase, F. Kato, I. Susa, S. Hasegawa, and M. Sato, "Stuffed toys alive!: cuddly robots from fantasy world," in *ACM SIGGRAPH 2012 Emerging Technologies*. ACM, 2012, p. 20.

[11] C. Duriez, "Control of elastic soft robots based on real-time finite element method," in *Robotics and Automation (ICRA), 2013 IEEE International Conference on*. IEEE, 2013, pp. 3982–3987.

[12] L. S. Chiang, P. S. Jay, P. Valdastri, A. Menciassi, and P. Dario, "Tendon sheath analysis for estimation of distal end force and elongation," in *Advanced Intelligent Mechatronics, 2009. AIM 2009. IEEE/ASME International Conference on*. IEEE, 2009, pp. 332–337.