Robotic Skins That Learn to Control Passive Structures

Jennifer C. Case [ID], Michelle C. Yuen [ID], Jane Jacobs, and Rebecca Kramer-Bottiglio [ID]

Abstract—Robotic skins incorporate sensors and actuators into stretchable and flexible planar substrates. Wrapping a robotic skin around a passive, deformable structure imparts controllable motion onto that structure, rendering it an active robotic system. Robotic skins can be applied to the surface of a structure, then removed and re-applied to the surface of another structure. This reconfigurability enables use of the same robotic skin to achieve multiple motions, which depend on the interaction between the skin and its host structure. Considering the broad range of use cases for robotic skins in resource-limited environments, it may not be possible to pre-characterize this skin–structure interaction for all potential systems. Therefore, it is advantageous to have systems that can learn their models *in situ*, which saves considerable time in realizing a functional system. Previously, we have shown that robotic skins can be used to estimate state and stiffness of the underlying passive structure they are attached to. In this letter, we demonstrate how this ability to measure state and stiffness can be used to learn model parameters *in situ* for feedforward control, and show how feedback control can be implemented simultaneously with the same system. We further show how this learning is expandable to multi-segment systems and will compensate for gravitational effects by adjusting model parameters.

Index Terms—Modeling, control, and learning for soft robots, sensor-based control, robotic skins.

I. INTRODUCTION

W HEN working in remote locations with limited access to resources, it is advantageous for a robotic system to be versatile so that it can accomplish a variety of tasks. For

Fig. 1. (a) A two-segment continuum robot made from two robotic skins wrapped around soft cylindrical segments. The sensors and actuators in the robotic skins are used to derive properties of the underlying segment and to control the system. (b) A visual description of a bending segment. (c) A cross-section of a segment with sensor labels.

this, we look to soft robotic systems which are lightweight and mechanically robust in unknown environments [1]–[3]. While some soft robotic systems have shown multifunctionality [4]–[6] and reconfigurability [7], [8], most are still generally designed around completing a single task, such as locomotion [1], [9], [10], grasping [11]–[13], or manipulation [14], [15]. In contrast, robotic skins, which incorporate both sensing and actuation in a flexible planar substrate, are versatile: the same robotic system can be used across a wide variety of applications, including locomotion, grasping, manipulation, and wearables [16]. Robotic skins operate by attaching to deformable host structures and imparting motion onto those structures. They are removable and transferable, meaning they can be attached to one host structure to attain one behavior, then removed and attached to a different host structure to attain a different behavior. This ability to

transfer from system-to-system makes robotic skins useful for tasks in unstructured environments that may require the system to adapt to changing task requirements.

In this letter, we present robotic skins that learn about their underlying host structures to facilitate feedforward control of the system. We demonstrate this approach on cylindrical continuum segments since they are widely applicable in soft robotics and can be easily stacked to create multi-segment arms, as shown in Figure 1(a). In this work, we define "learning" as the ability to use sensory information to obtain certain values of the host structure, specifically length, radius, and bending stiffness of the segments, and apply those values to populate control models. Previously, we demonstrated how robotic skins can be used to estimate both state and stiffness of cylindrical continuum segments [17]. Here, we show how this ability to estimate state and stiffness of structures can be extended to learn model parameters of continuum segments made from different materials *in situ*, which enables feedforward control. We show the necessity of having the skins learn by considering both a generalized feedforward model, which is provided with a bending stiffness, and compare it to our learned bending stiffness estimation. Additionally, we use the sensors in the robotic skin to apply feedback control both on its own and in combination with our learned feedforward model control. Finally, we demonstrate how this ability to learn about the underlying structures is expandable to multi-segment systems and is able to adjust parameters (i.e., the bending stiffness) to account for gravity, which is not included in the model of the cylindrical segment.

While there are many control strategies in soft robotics [18]–[21], we believe this is the first demonstration of robotic skins learning about structures for control purposes. By learning about the system, it is possible to change the system on-the-fly without the need to intervene and perform time-intensive system re-calibrations manually.

II. MODEL

The model used in this work was developed from existing models in continuum robotics [22]–[24], which describe the deformation of cylindrical segments given loads that run parallel to or spiral around the segment [25]. While dynamic models of these continuum robots exist [26]–[28], this work focuses on a static model to demonstrate that simple models can be used with robotic skins to control soft continua.

A full model description is presented in [17], and is summarized briefly here. The state of a cylindrical segment can be described by segment length (l_{seg}) and curvature in the x- and y-axes (respectively, κ_x and κ_y). A visual representation of curvature is shown in Figure 1(b). Sensors that are aligned parallel to the neutral axis of the segment can be used to identify the state of the segment:

$$\kappa_x = \frac{l_{s,2} - l_{seg}}{l_{seg}r} = \frac{l_{seg} - l_{s,4}}{l_{seg}r} \tag{1a}$$

$$\kappa_y = \frac{l_{s,1} - l_{seg}}{l_{seg}r} = \frac{l_{seg} - l_{s,3}}{l_{seg}r} \tag{1b}$$

where $l_{s,i}$ for $i = 1, 2, 3, 4$ is the length of the ith sensor, as shown in Figure 1(c), and r is the radius of the segment. The

bending stiffness of the combined segment and skin (K_b) can be estimated by applying a single actuator force (F) and relating this to the system curvature:

$$K_b = \frac{rF}{\kappa} \tag{2}$$

where $\kappa = \sqrt{\kappa_x^2 + \kappa_y^2}$.

III. FABRICATION

The robotic skins, shown in Figure 2(a), were designed to fit around cylindrical segments ($r = 17.5$ mm and $l_{seg} = 100$ mm). Though the segment dimensions are carried over from our prior work [17], we improved upon the robotic skin design to have it learn all necessary segment properties (r, l_{seg}, and K_b) on its own. Thus, the approach we present here can be readily applied to passive cylindrical segments of arbitrary sizes and materials (within reason). This concept can be expanded to arbitrary shapes in future work. In the following, we discuss the fabrication of the actuators, sensors, robotic skins, segments, and the experimental setup.

A. Actuators

The model assumes actuators that either contract or elongate and, thus, we chose to use McKibben pneumatic actuators which contract as they are pressurized. Each actuator was made with a latex balloon bladder surrounded by mesh braid (1/4" dia., McMaster-Carr). The balloon was tied on one end with the braid ziptied onto the balloon just inside of the knot, as shown in Figure 2(b). On the other end, we inserted Tygon tubing (6.4 mm outer diameter, 1.6 mm inner diameter) into the balloon and secured two zipties over the braiding around the tubing to prevent leaking. The McKibben actuator was designed to be 120 mm long to allow a sufficient range of motion, but still contract enough (to ≈85 mm) to deform the segment when pressurized.

B. Sensors

We used capacitive strain sensors to measure the lengths and radii of segments. The sensors were fabricated as a parallel-plate capacitor using an exfoliated graphite and silicone composite for the conductive layers and pure silicone elastomer for the dielectric layers. These sensors have five layers to help isolate the sensors from external disturbances. Descriptions of how to make the exfoliated graphite composite and the five layer sensor material are available in [29] and [16], [17], respectively. The sensors were designed with an active (gauge) length of 80 mm (overall length 132.1 mm). The vertical sensors were pre-strained by 20 mm when installed on the segment; the horizontal sensors were designed with a 16.5 mm pre-strain. The sensors were redesigned from [16], [17] to be more compact and now have the sensor board sitting on top of the sensor material rather than attached externally to the sensor footprint. Figure 2(c) shows how the final sensor components were assembled. The polystyrene (0.65 mm thick) and PET (0.1 mm thick) ends were cut with a laser (Universal Laser Systems VLS 2.30). The ends were sewn through the available holes to hold the sensor together. Finally, the end with the exposed copper was wrapped in electrical tape
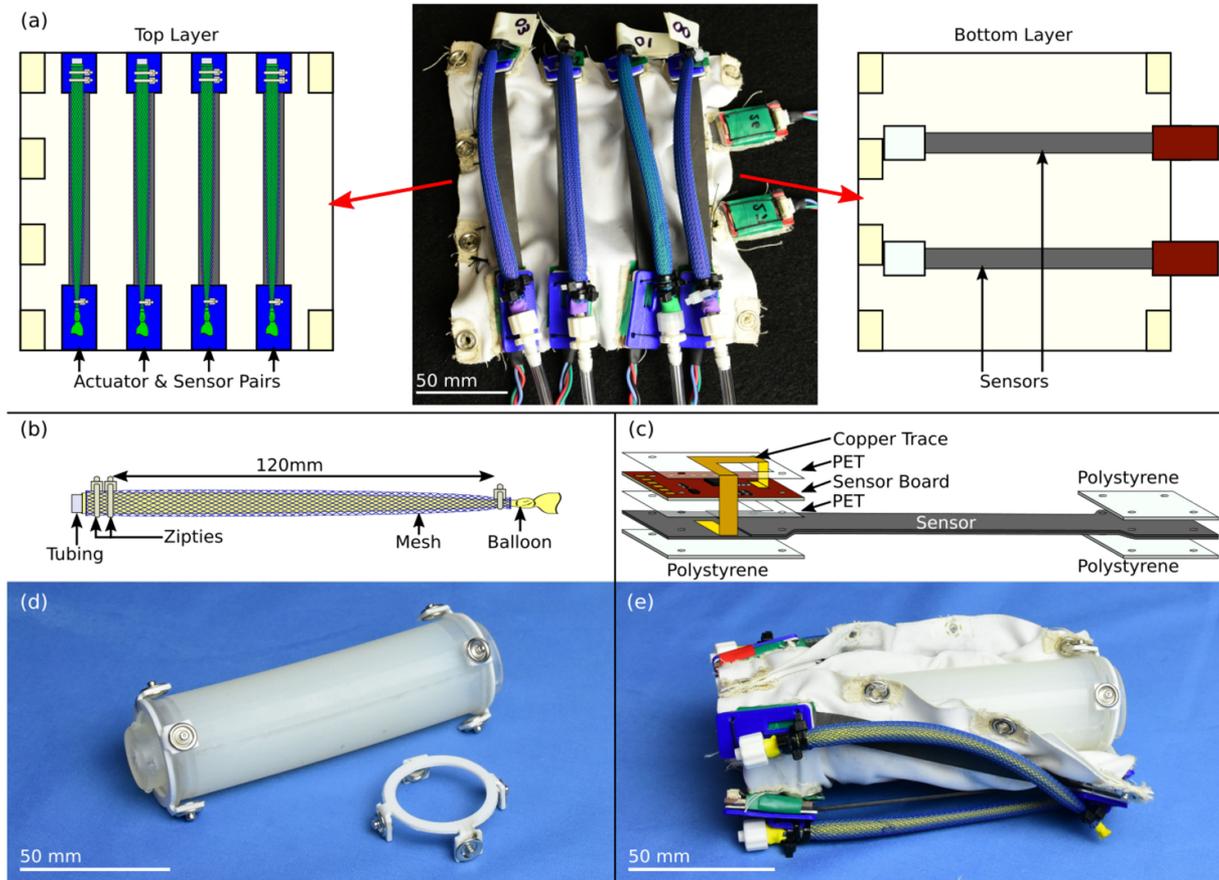
Fig. 2. (a) The robotic skin is composed of two layers of spandex instrumented with sensors and actuators. Diagrams show the layout of the skin. (b) Diagram and dimensions of the actuators. (c) Components and construction of the sensors. (d-e) A Dragon Skin 10 Slow elastomer segment with polystyrene skin attachments both (d) without and (e) with the robotic skin.

to prevent spurious signals arising from contact with objects. To reduce noise in the sensor response, we applied an exponential filter to the sensor output, $s_{f,t} = \alpha s_{r,t} + (1 - \alpha)s_{f,t-1}$, where $s_{f,t}$ is the filtered response at time t, $s_{r,t}$ is the raw sensor response at time t, and α is the decay rate, which we set $\alpha = 0.8$.

C. Robotic Skins

The robotic skins, shown in Figure 2(a), were organized such that they were able to learn the segment dimension. Each robotic skin consisted of two layers of spandex, six capacitive strain sensors and four actuators. Two horizontal sensors in the bottom layer measured the circumference of the underlying cylinder to allow the skin to determine the radius of the system while the other four vertical sensors in the top layer measured the segment length and are aligned as shown in Figure 1(b). The sensors were sewn directly onto the spandex fabric. An actuator was ziptied in place over each of the four vertical sensors using a 3D printed actuator attachment (Printrbot Metal Plus).

In order to facilitate testing, snap buttons were incorporated into the skins to allow for quick connection ($\approx$1.5 min) and disconnection ($\approx$0.7 min) from segments (Figure 2(d-e), Supplemental Video). The skins themselves were 140.0 mm $\times$ 160.1 mm and had muslin reinforcements where the sensor and actuators were attached and where snap buttons were mounted.

The button snaps were mounted through both layers of spandex to hold the two layers together and were located underneath all sensor-actuator attachment points and in all four corners of the skin.

D. Continuum Segments

For the continuum segments, we used two elastomers with different stiffness properties and fabricated them such that they could easily connect and disconnect to each other as well as the robotic skins (Figure 2(d)). The segments were molded from Dragon Skin 10 Slow (Smooth On, Inc.) and Smooth-Sil 935 (Smooth-On, Inc.), which have elastic moduli of 265 kPa and 536 kPa, respectively [17]. Male and female end caps were mechanically locked onto either end of the segment by casting the segments with the caps in place so that the elastomer infiltrated the porous body of the caps. The end caps and the molds for the segments were 3D printed with a Form 2 SLA printer (FormLabs). To attach the robotic skin to the elastomer, we made polystyrene (1.6 mm thick) fabric attachments (Figure 2(d)). These attachments were cut out using a laser cutter (Universal Laser Systems VLS 2.30) and have snap buttons placed on the tabs, which were bent in place by heating the polystyrene and bending the tab over. When assembled with the robotic skin (Figure 2(e)), the tension in the skin holds the fabric attachments securely on the ends of elastomer segment. The ability to

rearrange the segments and the skins using the snap buttons made it easy to assemble and disassemble the system for testing.

E. Experimental Setup

The robotic skin-elastomer system was attached to a base which held it vertically for experiments (Figure 1(a)). The 3D printed base had a female interlocking mechanism (FormLabs, Form 2) to lock the segment in place. Mortar board was placed over the 3D printed part and clamped in place to stabilize the system. The overall experimental setup was designed to allow the system to be quickly changed for various segments.

A central microcontroller (Arduino Uno) was used to communicate with the skins during operation. This microcontroller used I2C communication to read from and write to the sensor boards and pressure regulators that drive the actuators to a desired pressure [30]. The Arduino Uno relayed all the results to a computer through serial communication to record the system response.

IV. CHARACTERIZATION

A. Sensors

The sensors were characterized in a materials testing machine (Instron 3345) after they were integrated into the skin to determine a linear fit for each sensor. The skin was stretched such that the gauge length of the sensor stretched from 95 mm to 120 mm in steps of 2.5 mm; three sensor readings for each sensor were collected at each step. This process was repeated 10 times for a total of 30 data points at each step. The results for each sensor were averaged together and a linear fit, $IC_i = a_{0,i} l_{s,i} + a_{1,i}$, was found for each sensor's response. IC_i refers to the number of instruction cycles corresponding to the discharge time of the ith capacitive sensor within a measurement query and thus, serves as a proxy for the sensor's capacitance. $a_{j,i}$ for $j = 0, 1$ are the parameters of the equation for the ith sensor. A representative plot of the response and fit of a single sensor is provided in Figure 3(a). Since each sensor is unique, this characterization was performed on each sensor to attain an accurate relationship between length and sensor reading.

B. Actuators

The actuators were characterized prior to integration to relate the actuator's force response to its length and internal pressure. Each actuator was secured into the materials testing machine at an initial length of 110 mm, inflated to a constant air pressure, and then contracted to 70 mm at a rate of 40 mm/min. This test was performed at several pressures (10, 15, 20, and 25 psi) which were held constant using pressure regulators [30]. We then related the force to both length and pressure to obtain a theoretical surface fit:

$$F_i = b_{0,i} + b_{1,i} p_i + b_{2,i} l_{a,i} + b_{3,i} p_i l_{a,i} + b_{4,i} l_{a,i}^2 \quad (3)$$

where F_i is the ith actuator force, p_i is the internal pressure of the ith actuator, $l_{a,i}$ is the ith actuator length (we assume $l_{a,i} \equiv l_{s,i}$ in our model), and $b_{j,i}$ for $j = 0, 1, 2, 3, 4$ are parameters for the ith actuator. A representative plot showing the responses and fits of a single actuator is provided in Figure 3(b). Since
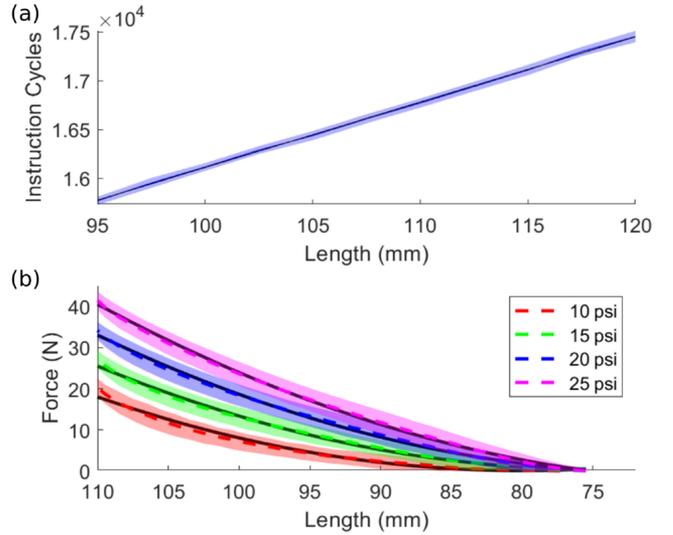
Fig. 3. Representative plots of (a) sensor and (b) actuator characterization. (a) Sensor characterization relates the sensor length to the sensor response given in instruction cycles . (b) 2D representation of the actuator characterization. The shaded regions represent the 95% confidence interval of the experimental data and the black lines represent the theoretical fits.

each actuator is unique, this characterization was performed on each actuator to attain an accurate relationship between length, pressure, and force.

V. RESULTS

A. State Estimation

State estimation was enabled by the model presented in Section II and the actuator characterization. The model was implemented in a MATLAB simulation that predicted the system state (i.e., κ_x and κ_y) given the segment radius, bending stiffness, and actuator pressures. The segment radius was measured from our horizontal sensors using the following equation:

$$r = (l_{s,h} + c + \Delta_{skin})/2\pi \quad (4)$$

where $l_{s,h}$ is the average length of the two horizontal sensors, c is the inextensible length of the polystyrene tabs (27.5 mm), and Δ_{skin} is the length the spandex fabric stretched, independent of the sensors. Because the sensors did not wrap all the way around the circumference of the segment, portions of the skin stretched without being measured by the sensors. For this work, we manually measured that unobserved stretch when needed. Complementing the simulation, the bending stiffness can be found via Equation 2 given the actuator characterization results (Equation (3)), segment radius, and curvature (i.e., system state). To demonstrate the robustness of our approach, we tested (1) different segment diameters while holding input pressures constant and (2) different pressure inputs while holding the segment diameter constant.

To demonstrate the skin's adaptability to segments with varying diameters, we compared the skin's theoretical and actual curvature on multiple Dragon Skin 10 Slow segments (35, 40, and 45 mm diameters). We set a single actuator to 22 psi, recorded the horizontal sensor lengths, and captured photos of segment. For the 35, 40, and 45 mm diameter segments, we found that
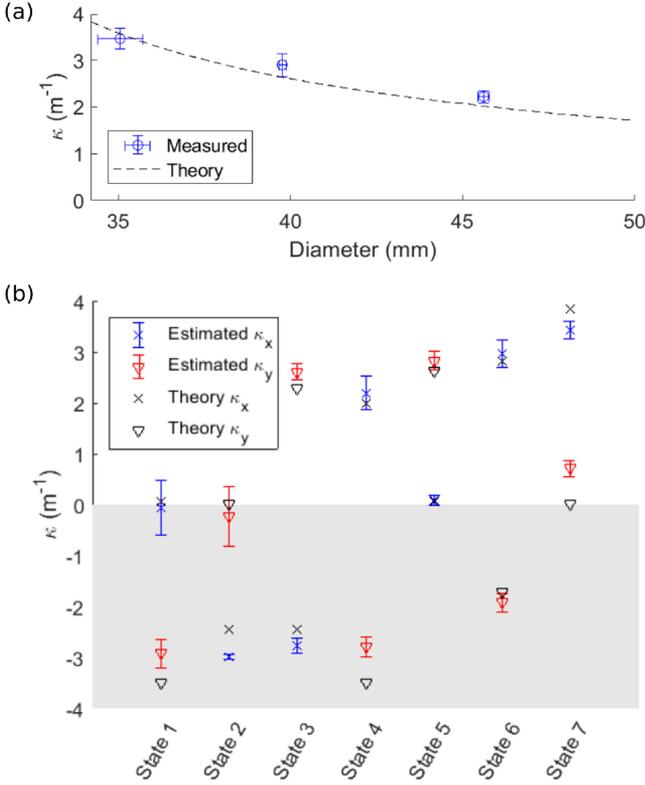
Fig. 4. (a) Comparison of measured and theoretical segment curvatures across different radii. Additionally, the plot shows the accuracy of the sensor-measured segment diameter. (b) State estimation results for each of the states presented in Table I. The error bars represent the 95% confidence interval.

TABLE I
STATE ESTIMATION CONDITIONS

State	p_1 (psi)	p_2 (psi)	p_3 (psi)	p_4 (psi)
1	20	0	0	0
2	0	15	0	0
3	0	15	15	0
4	20	0	0	10
5	0	0	17	0
6	10	0	0	15
7	0	0	0	22

$\Delta_{skin} = 0, 8, 16$ mm, respectively, which indicates that the stretching skin could be treated as a spring. We used ImageJ to measure the curvature from the images. Figure 4(a) shows how the measured diameters and curvatures compare to the model-predicted curvatures at various diameters. From the measured curvature, we calculated the bending stiffness of the different segments as 56, 82, and 135 kN mm^2 for the 35, 40, and 45 mm diameter segments. A linear regression was fit to these values to generate stiffnesses at different segment diameters for the theoretical model. We found that there is good agreement between the measured and theoretical values of curvature.

To verify that the skin's vertical sensors can also be used for state estimation, we compared the state generated by the sensors to the theoretical model. This verification was performed on a Dragon Skin 10 Slow segment across seven actuation states (Table I) each tested 10 times. We measured the segment length with the vertical sensors and the radius with

TABLE II
SETTLING TIME OF VARIOUS CONTROL STRATEGIES

	Feedforward	Feedback	Combined
Dragonskin 10 Slow	0.4 ± 0.5 s	1.9 ± 0.6 s	1.0 ± 1.5 s
Smooth-Sil 935	0.4 ± 0.6 s	2.1 ± 1.1 s	1.0 ± 1.8 s

the horizontal sensors ($r = 18.6 \pm 0.6$ mm.). Equation (1) requires the segment length to determine the curvatures of the system. Slight variations in the initial segment length can have a large impact on the resulting κ_i calculations, which can result in errors for our state estimation. Thus, we used unique initial sensor lengths ($l_{s,1} = 100.4 \pm 0.3$ mm, $l_{s,2} = 103.3 \pm 0.8$ mm, $l_{s,3} = 101.7 \pm 0.3$ mm, $l_{s,4} = 98.4 \pm 0.4$ mm, with an overall average of $l_{seg} = 100.9 \pm 3.1$ mm) for these curvature calculations depending on which sensors are used for the κ_i calculations. Similar to our previous work [17], the state estimation predicted by the sensors required a corrective offset (1.5 m^{-1} for this work) to match the state of the system.

The estimated states, given as κ_x and κ_y, are shown in Figure 4(b) for each state alongside the model-predicted state. The error in curvature did not exceed 0.7 m^{-1}, which we believe is an acceptable level of error for a 2D robotic skin being wrapped around 3D structures. This error can be reduced in the future by improving upon the state estimation model, sensor accuracy, and adaptability of the skin design.

B. Control

With reasonable state estimation, we implemented several control approaches for the one-segment system to evaluate the efficacy of each. The first approach was feedforward control which uses a model to predict what the inputs (pressures) of the system should be. For this, we considered two feedforward models: a generalized model, which had the bending stiffness supplied, and a learned model, which used a learning sequence in combination with state estimation to determine the bending stiffness of the segment. Additionally, we explored feedback control both on its own and in combination with learned feedforward models. Note that all of these single segment tests were performed with a single robotic skin to demonstrate that a single skin is not characterized for a particular material. The controllers operated at 10 Hz. The performance of each controller is represented by the average error per second (error/s) in Figure 5 and the settling times shown in Table II. By analyzing all of these control methods, we can select which control strategy is appropriate for future applications.

1) Feedforward Control: Because the robotic skins are able to detect material properties from the underlying structures and the constituent components have been thoroughly characterized, we were able to develop an approximate model of the system for feedforward control which predicts what actuator pressures are necessary for a desired state. For a given desired configuration ($\kappa_{x,d}$ and $\kappa_{y,d}$), the actuator pressures should be set as follows from Equation (3):

$$p_i = \frac{F_i - b_{0,i} - b_{2,i}l_{a,i} - b_{4,i}l_{a,i}^2}{b_{1,i} + b_{3,i}l_{a,i}} \tag{5}$$
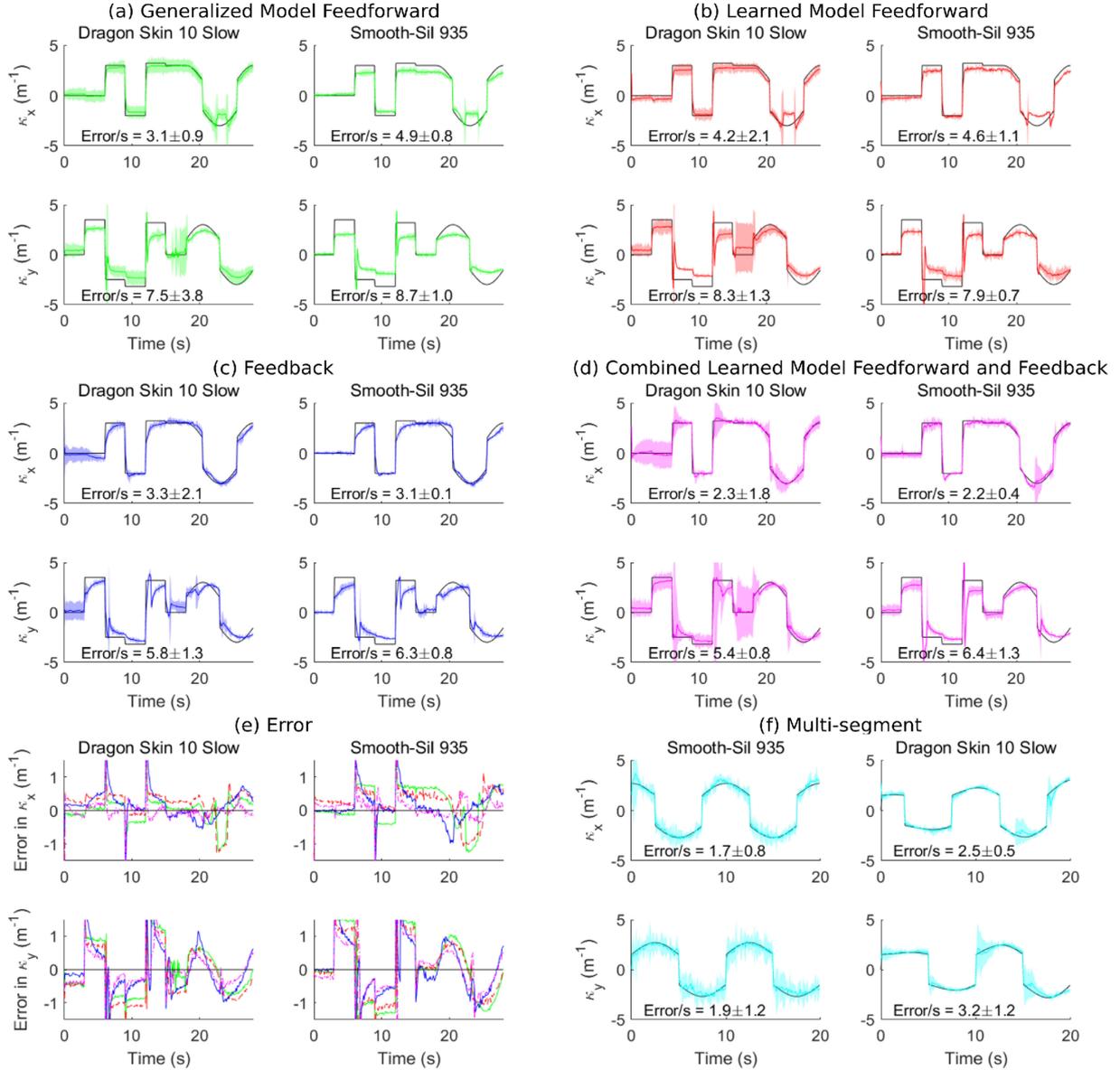
Fig. 5. Control response using (a) generalized model feedforward control, (b) learned model feedforward control, (c) feedback control, and (d) learned model with feedback control. The average error in control response is shown in (e). The colors in (e) correspond to the controllers in (a)-(d). Control response of two segments is shown in (f). The black line represents $\kappa_{x,des}$ and $\kappa_{y,des}$ while the colored line represents the average system response across three trials. The shaded region represents the 95% confidence interval.

where the actuator lengths can be defined as

$$l_{a,1} = l_{seg}(1 + \kappa_{y,d}r) \tag{6a}$$

$$l_{a,2} = l_{seg}(1 + \kappa_{x,d}r) \tag{6b}$$

$$l_{a,3} = l_{seg}(1 - \kappa_{y,d}r) \tag{6c}$$

$$l_{a,4} = l_{seg}(1 - \kappa_{x,d}r) \tag{6d}$$

and the forces of the actuators are defined as

$$\text{If } \kappa_{y,d} > 0 : F_{a,1} = 0, \ F_{a,3} = \frac{\kappa_{y,d}K_b}{r} \tag{7a}$$

$$\text{If } \kappa_{x,d} > 0 : F_{a,2} = 0, \ F_{a,4} = \frac{\kappa_{x,d}K_b}{r} \tag{7b}$$

$$\text{If } \kappa_{y,d} < 0 : F_{a,1} = -\frac{\kappa_{y,d}K_b}{r}, \ F_{a,3} = 0 \tag{7c}$$

$$\text{If } \kappa_{x,d} < 0 : F_{a,2} = -\frac{\kappa_{x,d}K_b}{r}, \ F_{a,4} = 0 \tag{7d}$$

To demonstrate the need for these robotic skins to learn about their underlying material, we tested feedforward control with a generalized model where the bending stiffness was set at $K_b = 60$ kN mm^2 and compared it to a learned model strategy where the stiffness is estimated by the skin ahead of time. Both feedforward models use the learned l_{seg} and r values, which are found immediately when the system is powered, since those are necessary for state estimation. For the learned feedforward model, the stiffness estimation was performed prior to running

the controller (shown in Supplemental Video). Each actuator was sequentially pressurized to 15.0, 18.5, and 22.0 psi. Using the sensors to measure the curvature, a stiffness value was calculated at each condition for a total of 12 stiffness measurements. These measurements were then averaged together to obtain the learned bending stiffness of the system. By allowing the system to learn about the system's behavior *in situ*, the control signal is better tuned to the system, which results in a reduced error.

Figure 5(a) shows the results of the generalized model feedforward control for both a Dragon Skin 10 Slow and Smooth-Sil 935 elastomer segment. Here, we can see the feedforward control for the Dragon Skin 10 Slow segment performed well since the generic bending stiffness value was around the bending stiffness for this specific skin-elastomer system. When the feedforward controller with the same generic bending stiffness was applied to a different material (Smooth-Sil 935), the response of the system suffered, as evidenced by the increased error. In contrast, Figure 5(b) shows the system response when the robotic skin learned about its own bending stiffness. The bending stiffnesses were found to be $K_b = 65 \pm 4$ kN mm^2 and $K_b = 105 \pm 7$ kN mm^2 for Dragon Skin 10 Slow and Smooth-Sil 935, respectively. The average error of each response is plotted in Figure 5(e) and this shows the slight improvement in error between the generalized model feedforward (shown in green) and learned model feedforward (shown in red) controllers for the Smooth-Sil 935 segment. For the Dragon Skin 10 Slow segment, the learned feedforward model did not perform as well as the generalized feedforward model due to slight error in the learned stiffness estimation. Overall, we observed that the performance of the learned feedforward model is comparable to that of the generalized feedforward model.

2) Feedback Control: Although the feedforward model showed reasonable accuracy in approximating the system, inaccuracies in the model itself and the input values to the model (i.e., bending stiffness) resulted in non-negligible error between the commanded and achieved segment curvatures. Since the robotic skin already has sensors, we implemented feedback control to reduce curvature error by adjusting the actuator pressure. We used a proportional-derivative (PD) feedback controller that applied a change to the pressure based on the error. The controller algorithm is given as

$$\text{If } \kappa_{y,d} > 0 : p_1 = 0, p_3 += k_p(\kappa_{y,d} - \kappa_y)$$
$$+ k_d \frac{\kappa_{y,d} - \kappa_y}{\Delta t} \quad (8a)$$
$$\text{If } \kappa_{x,d} > 0 : p_2 = 0, p_4 += k_p(\kappa_{x,d} - \kappa_x)$$
$$+ k_d \frac{\kappa_{x,d} - \kappa_x}{\Delta t} \quad (8b)$$
$$\text{If } \kappa_{y,d} < 0 : p_1 -= k_p(\kappa_{y,d} - \kappa_y)$$
$$+ k_d \frac{\kappa_{y,d} - \kappa_y}{\Delta t}, p_3 = 0 \quad (8c)$$
$$\text{If } \kappa_{x,d} < 0 : p_2 -= k_p(\kappa_{x,d} - \kappa_x)$$
$$+ k_d \frac{\kappa_{x,d} - \kappa_x}{\Delta t}, p_4 = 0 \quad (8d)$$

where k_p and k_d are the proportional and derivative gains and Δt is the change in time.
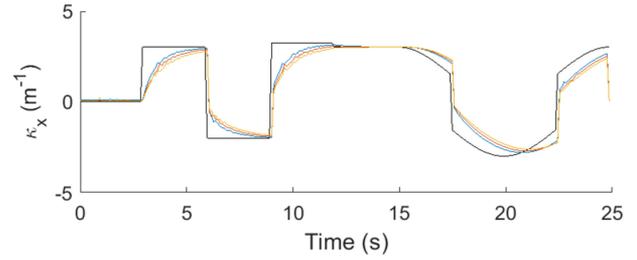
Fig. 6. Simulation showing how the feedback controller responds to changes in bending stiffness. The black line is the reference while the blue, red, and yellow lines show systems with bending stiffnesses of 60 KN mm^2, 80 kN mm^2, and 100 kN mm^2, respectively.

Here, we compared using only feedback control, shown in Figure 5(c), and using feedback control alongside the learned feedforward controller, shown in Figure 5(d). Since the system was entirely modeled, including the actuators, we built a simulation that predicted the system performance given a set of pressure inputs. This simulation was used to tune the feedback controller gains ($k_p = 0.75$ and $k_d = 0.005$), which were kept constant across all the tests regardless of materials, shown in Figure 6. For the learned model, the bending stiffnesses were found to be $K_b = 69 \pm 2$ kN mm^2 and $K_b = 100 \pm 5$ kN mm^2 for the Dragon Skin 10 Slow and Smooth-Sil 935 segments, respectively. The slight variations in the bending stiffness between these and the previous values are due to the fact that the system performs the learning sequence each time the controller is run rather than using a single system learning result across all tests. The average error responses are presented in Figure 5(e) for the feedback (blue) and feedforward and feedback (magenta) controllers. Using only feedback control, we can see that the system is controlled well and actually better controlled than the feedforward alone, as seen by the error measurements presented in Figure 5. Using a combination of learned model feedforward and feedback, we see an additional improvement in control of the system. The settling time of the system is improved due to the feedforward component of the controller, as shown in Table II, while the modeling error is reduced due to the feedback component of the controller.

While there is an advantage in using the combination of feedforward and feedback, this may not be necessary for every application and feedback alone may be sufficient. In scenarios where the system has a slower time response, it would be advantageous to use the combined feedback and feedforward controller, rather than pure feedback, since the feedforward model can be changed to account for the time component of the system.

C. Multi-Segment Control

To demonstrate that this combination of feedback and learned model feedforward control method is expandable to multi-segment continuum robots, we created a two segment system with a Dragon Skin 10 Slow segment stacked on top of a Smooth-Sil 935 segment. Each segment was controlled by independent feedforward controllers, and thus, each performed its own characterization to learn about its model consecutively (first the bottom segment, then the top) before beginning the control sequence. The bending stiffnesses were found to be

$K_b = 55 \pm 3$ kN mm^2 and $K_b = 140 \pm 10$ kN mm^2 for Dragon Skin 10 Slow and Smooth-Sil 935, respectively. The bottom segment was able to compensate for the additional weight of the top segment by increasing the bending stiffness estimation for the Smooth-Sil 935 segment. Based on the performance observed in the single-segments tests, we modified the control signal for the two-segment system to keep the desired curvature within the observed limits. The results of this two segment test are shown in Figure 5(f) and in the Supplemental Video. From these plots, we observe that by using this simple learning approach, controller performance is maintained as the system is expanded to multiple segments.

VI. Conclusion

In this letter, we demonstrated how robotic skins can be used to learn system parameters to create feedforward models, and to implement feedforward and feedback control of elastomeric continuum robots. We showed how different forms of feedforward and feedback control affect system performance. Given highly accurate models, feedforward control can be used alone to control systems. However, in the case of robots created on-the-fly, for example by wrapping robotic skins around arbitrary soft passive bodies, these models may not be available. As such, the efficacy of such on-the-fly robots relies upon the ability to self-characterize or learn system parameters *in situ* for adequate feedforward control. To refine learned feedforward control, robotic skins can simultaneously incorporate feedback to improve overall system performance. The robotic skin-enabled approach presented here allows for greater versatility and deployability of controllable soft robotic systems. Control of these systems can be further improved by applying adaptive control to the feedforward model, and tuning gains of the feedback controller. Additional work on the control of multi-segment continuum arms includes developing higher-level controllers that wrap around individual segment controllers to control aspects of the system such as end-point position. We also plan to develop a generalized model of the robotic skins to explain how they interact with underlying structures and to explore alternative actuators that can apply more force and increase the maximum curvature achieved by the system. In conclusion, robotic skins can learn system parameters *in situ* and use the generated feedforward model in tandem with feedback control to effectively transform arbitrary soft passive structures into robots on-the-fly.

References

[1] M. T. Tolley *et al.*, "An untethered jumping soft robot," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Sep. 2014, pp. 561–566.

[2] D. Trivedi, C. D. Rahn, W. M. Kier, and I. D. Walker, "Soft robotics: Biological inspiration, state of the art, and future research," *Appl. Bionics Biomechanics*, vol. 5, pp. 99–117, Dec. 2008.

[3] S. Kim, C. Laschi, and B. Trimmer, "Soft robotics: A bioinspired evolution in robotics," *Trends Biotechnology*, vol. 31, pp. 287–294, May 2013.

[4] A. A. Stokes, R. F. Shepherd, S. A. Morin, F. Ilievski, and G. M. Whitesides, "A hybrid combining hard and soft robots," *Soft Robot.*, vol. 1, pp. 70–74, Jul. 2013.

[5] M. Calisti *et al.*, "Design and development of a soft robot with crawling and grasping capabilities," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2012, pp. 4950–4955.

[6] M. Calisti *et al.*, "An octopus-bioinspired solution to movement and manipulation for soft robots," *Bioinspiration Biomimetics*, vol. 6, Sep. 2011, Art. no. 036002.

[7] S. W. Kwok *et al.*, "Magnetic assembly of soft robots with hard components," *Adv. Functional Mater.*, vol. 24, pp. 2180–2187, Apr. 2014.

[8] S. A. Morin *et al.*, "Using click-e-bricks to MAKE 3D elastomeric structures," *Adv. Mater.*, vol. 26, pp. 5991–5999, Sep. 2014.

[9] D. Drotman, S. Jadhav, M. Karimi, P. deZonia, and M. T. Tolley, "3D printed soft actuators for a legged robot capable of navigating unstructured terrain," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2017, pp. 5532–5538.

[10] M. Luo *et al.*, "Slithering towards autonomy: A self-contained soft robotic snake platform with integrated curvature sensing," *Bioinspiration Biomimetics*, vol. 10, no. 5, 2015, Art. no. 055001.

[11] F. Ilievski, A. D. Mazzeo, R. F. Shepherd, X. Chen, and G. M. Whitesides, "Soft robotics for chemists," *Angewandte Chemie*, vol. 123, pp. 1930–1935, Feb. 2011.

[12] M. Manti, T. Hassan, G. Passetti, N. dElia, M. Cianchetti, and C. Laschi, "An under-actuated and adaptable soft robotic gripper," in *Biomimetic and Biohybrid Systems* (Lecture Notes in Computer Science). Cham, Switzerland: Springer, Jul. 2015, pp. 64–74.

[13] K. C. Galloway *et al.*, "Soft robotic grippers for biological sampling on deep reefs," *Soft Robot.*, vol. 3, pp. 23–33, Jan. 2016.

[14] A. D. Marchese, K. Komorowski, C. D. Onal, and D. Rus, "Design and control of a soft and continuously deformable 2D robotic manipulation system," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2014, pp. 2189–2196.

[15] A. D. Marchese and D. Rus, "Design, kinematics, and control of a soft spatial fluidic elastomer manipulator," *Int. J. Robot. Res.*, vol. 35, no. 7, pp. 840–869, 2016.

[16] J. W. Booth *et al.*, "OmniSkins: Robotic skins that turn inanimate objects into multifunctional robots," *Sci. Robot.*, vol. 3, Sep. 2018, Art. no. 1853.

[17] J. C. Case, J. Booth, D. S. Shah, M. C. Yuen, and R. Kramer-Bottiglio, "State and stiffness estimation using robotic fabrics," in *Proc. IEEE Int. Conf. Soft Robot.*, Apr. 2018, pp. 522–527.

[18] T. George Thuruthel, Y. Ansari, E. Falotico, and C. Laschi, "Control strategies for soft robotic manipulators: A survey," *Soft Robot.*, vol. 5, no. 2, pp. 149–163, Jan. 2018.

[19] J. C. Case, E. L. White, and R. K. Kramer, "Sensor enabled closed-loop bending control of soft beams," *Smart Mater. Struct.*, vol. 25, no. 4, 2016, Art. no. 045018.

[20] E. L. White, J. C. Case, and R. Kramer-Bottiglio, "A soft parallel kinematic mechanism," *Soft Robot.*, vol. 5, pp. 36–53, Dec. 2017.

[21] S. Ozel *et al.*, "A composite soft bending actuation module with integrated curvature sensing," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2016, pp. 4963–4968.

[22] R. J. Webster and B. A. Jones, "Design and kinematic modeling of constant curvature continuum robots: A review," *Int. J. Robot. Res.*, vol. 29, no. 13, pp. 1661–1683, Jun. 2010.

[23] J. Burgner-Kahrs, D. C. Rucker, and H. Choset, "Continuum robots for medical applications: A survey," *IEEE Trans. Robot.*, vol. 31, no. 6, pp. 1261–1280, Dec. 2015.

[24] M. W. Hannan and I. D. Walker, "Kinematics and the implementation of an elephant's trunk manipulator and other continuum style robots," *J. Robot. Syst.*, vol. 20, pp. 45–63, Feb. 2003.

[25] D. C. Rucker and R. J. Webster, "Mechanics of continuum robots with external loading and general tendon routing," in *Experimental Robotics* (Springer Tracts in Advanced Robotics). Berlin, Germany: Springer, 2014, pp. 645–654.

[26] D. C. Rucker and R. J. W., III, "Statics and dynamics of continuum robots with general Tendon routing and external loading," *IEEE Trans. Robot.*, vol. 27, no. 6, pp. 1033–1044, Dec. 2011.

[27] F. Renda, M. Giorelli, M. Calisti, M. Cianchetti, and C. Laschi, "Dynamic model of a multibending soft robot arm driven by cables," *IEEE Trans. Robot.*, vol. 30, no. 5, pp. 1109–1122, Oct. 2014.

[28] E. Tatlicioglu, I. Walker, and D. Dawson, "New dynamic models for planar extensible continuum robot manipulators," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Oct. 2007, pp. 1485–1490.

[29] E. L. White, M. C. Yuen, J. C. Case, and R. K. Kramer, "Low-cost, facile, and scalable manufacturing of capacitive sensors for soft systems," *Adv. Mater. Technol.*, vol. 2, no. 9, Art. no. 1700072.

[30] J. W. Booth, J. C. Case, E. L. White, D. S. Shah, and R. Kramer-Bottiglio, "An addressable pneumatic regulator for distributed control of soft robots," in *Proc. IEEE Int. Conf. Soft Robot.*, Apr. 2018, pp. 25–30.