A New Soft Robot Control Method

Using Model Predictive Control for a Pneumatically Actuated Humanoid

By Charles M. Best, Morgan T. Gillespie, Phillip Hyatt,
Levi Rupert, Vallan Sherrod, and Marc D. Killpack

Traditional rigid robots, such as those used in manufacturing, have been effective at precise, accurate, rapid motions in well-structured environments for many decades now. However, they operate largely behind cages due to the danger of injury when moving in close proximity to people. A significant and recent shift in robotics involves trading rigid links and rigid actuators for soft, deformable links and compliant actuators. These soft robots generally have lower inertia and avoid many of the problems caused by the high effective inertia resulting from the high gear ratios necessary for rigid robots.

This shift to soft robots will allow robots to safely operate in close proximity to humans. The design of such soft robots, which has been a major emphasis in research, does not completely enable soft robots to effectively perform tasks. The focus of this article is the development of dynamic models and control methods to allow multiple-degrees-of-freedom

(DoF) soft robots to perform useful tasks. We present results on fully inflatable, pneumatically actuated soft robot platforms that include a 14-DoF humanoid robot.

The Benefits of Soft Robots

There has been significant interest in making robots more effective at interacting with humans and operating in human environments. Robots currently have limited uses in homes, hospitals, schools, or other areas where safe interaction with people or the environment may be necessary. One reason why robots are not common in these places is because traditional robots can be dangerous to people or property when there is incidental contact.

Benefits from soft robot platforms such as the one for control development that we describe in this article include the fact that the compliance comes from passive elements (air) instead of requiring active sensors and controllers for added safety. In addition, we control the robot at fairly low pressures, including 6.9–13.8 kPa (1–2 lbf/in^2) gauge in the main body chamber and 172.4 kPa (25 lbf/in^2) gauge in the actuation bladders. In comparison, these pressures are less than a bike tire, where pressures range from 206.8 to 896.3 kPa (30–130 lbf/in^2). This means there is lower risk of injury when there is failure such as bursting or leaking of air. In addition to compliance, lightweight robots, such as the platform for which

we present results, have less inertia and are less likely to cause bodily harm, because of lower contact forces and lower overall momentum when moving at varying speeds. Finally, this specific platform can be contained in a very small packing volume when deflated. This combination of low weight and small packing volume is very beneficial for applications where the cost of larger and heavier robots becomes prohibitive, such as in space or even in search and rescue operations. Our research is motivated by a desire to take advantage of the positive characteristics of soft and inflatable robotic systems while maintaining a level of control that will allow them to be useful. Applications for a robot of this type include search and rescue, health care, assistance with activities of daily living, and space exploration.

> **This combination of low weight and small packing volume is very beneficial for applications where the cost of larger and heavier robots becomes prohibitive.**

In this article, we present dynamic models and control methods for soft robots. Our results show the feasibility of an approach that is considerably different than most current soft robotics research for manipulation, since the complete structure and actuation of our robot comes entirely from air instead of any rigid links or cable-driven actuators. Although control of our fabric-based, pneumatically actuated soft robots is a particularly difficult problem, we present encouraging and repeatable results.

Related Research on Soft Robots and Control

In [1], lightweight structures are listed as a desirable design characteristic for a soft robot. The soft robots we use in this article are at least an order of magnitude lighter than most multi-DoF robots, even those previously referenced as lightweight in [1], since our entire humanoid robot (not including the pneumatic control valves) has a mass of approximately 13.6 kg (30 lb). Past research similar to that presented in this article can be divided into two main areas: first, control methods for other soft robots, and second, past applications of a specific type of optimal control that we are using called *model predictive control* (*MPC*).

Controlling Soft Robots

Our soft robot platform is a pneumatically actuated, inflatable robot that is lightweight and has a high strength-to-weight ratio. Related to this fabric-based test bed, there has been a significant amount of work done in developing materials, sensors, structures, and actuators that are lightweight and compliant. These materials are often inspired by biological systems, and many are discussed in the literature [2]. Past research involving inflatable robots has mostly looked at the design and performance of an actuator or a series of actuators. In our research, we show that an entire system can be inflatable and that control methods can be developed for the system to effectively complete tasks normally done by a robot with a rigid structure. The lack of literature on the control of inflatable structures where there is a wide range of applications suggests a novel and important area of robotics research.

In previous work on controlling soft robots, researchers were able to limit contact forces using inflatable links with cable tendon and dc servo motor actuators [3], [4]. While cable-driven actuators are an effective means of actuation for inflatable structures, our work has focused on using antagonistic pneumatic bladders, which are more consistent with the design intent of completely inflatable structures.

In [5]–[7], using an actuation method similar to that described in this article, it was found that motion planning was possible for fluid-driven elastomer actuators using dynamic models and constant-curvature kinematics. However, although they use multiple DoF, the manipulator motion is restricted to two dimensions in a plane. Additionally, to reach specific locations, they require learning a new control policy. Also similar to our work is research that uses rotary elastic chamber actuators, such as in [8] and [9], where two antagonistic bellows impart torque on an armature rotating about a rigid rotary joint. However, these compliant joints and their benefits are limited by the fact that they are still connected by rigid, higher-inertia links.

A major improvement in the results we present in this article is that previously we modeled torque on an inflatable joint with a linear impedance model [10]. This model included a mapping between a desired joint angle and corresponding equilibrium pressures. This mapping overly simplified the model from two individual actuation chamber pressures for each joint to representing actuation pressures as a single input. In preliminary work with a single DoF, we showed that including actuation pressures as state variables significantly improved performance and allowed us to control position and stiffness simultaneously [11]. In this article, we show how we identified new models that related pressure in the antagonistic actuation chambers to torque applied at the joint. We show significant improvement in performance as compared to our results in [10] and quantify repeatability for a multi-DoF soft robot manipulator unlike that in [11]. We expect that relating torque to pressure on an individual link-by-link basis will allow us to more accurately model joint coupling in future work.

MPC for Robotics

The fact that for our hardware platform we have two control inputs for each joint (i.e., the dynamics between the states and inputs are coupled) makes traditional proportional-integral-derivative (PID) control or other single-input, single-output (SISO) control methods less applicable. The SISO designation simply means that a dynamic system has one control input and one state variable or output of interest. Instead, we use MPC, a model-based control method. MPC is a form of optimal control that has long been used in the chemical processing industry. The main idea is that we can minimize a cost

function over a finite time horizon subject to the dynamics of our system expressed as an equality constraint. This is similar in many ways to a linear-quadratic regulator (LQR) with a finite horizon. However, there are two major differences. The first is that we often include other constraints on the states and inputs that are useful in either describing real limitations or in forcing certain states to be within user-defined limits. The second difference between MPC and LQR is that we solve this optimization for every time step that the controller is running and apply only the first resultant control input. This allows us to update the system model, constraints, and disturbances to get some of the benefits of both closed-loop feedback control and optimal control.

Recent advances in computing power and dynamic optimization techniques such as those presented in [12] have made MPC a viable control method in applications that require a high control rate. MPC has been demonstrated in robotics applications such as in the control of unmanned aerial and surface vehicles [13]–[15] and more recently in robot manipulation with rigid links [16]–[19].

Soft Robot Test Beds

The platforms used for this research include a 14-DoF humanoid robot called King Louie (Figure 1) and a single-DoF joint called a *grub* (Figure 2). Both were developed and built by Pneubotics, an affiliate of Otherlab. The robot platforms are based on the designs for rotary, fabric-based, pneumatically actuated joints that were shown as the PneuArm in [20] and [21]. Besides internal electronics such as inertial measurement units (IMUs) and pressure sensors, these platforms are made entirely of ballistic nylon fabric with internal bladders to prevent air leakage. The structure of these robots comes from an inflatable main bladder that is pressurized to 6.9–13.8 kPa (1–2 lbf/in^2) gauge.

At each joint, there are two antagonistic actuators that can be filled to pressures of 0–172.4 kPa (0–25 lbf/in^2) gauge. For this research, we use pressures of 0–117.2 kPa (0–17 lbf/in^2) gauge because of pinching effects in the main chamber at higher pressures. The source pressure is provided by an air compressor regulated to 172.4 kPa (25 lbf/in^2) gauge.

Enfield LS-v25s five-port spool proportional flow valves are used to control the variable flow of air from the pressure source to the actuation bladders or from the actuation bladders to the atmosphere. This platform utilizes only one output port of the Enfield valves, effectively using the valves as three-port spool valves. As seen in Figure 3, each actuation bladder has an individual valve for control of air flow, while both bladders share the same pressure source.

We use the Robot Operation System (ROS) to access pressure sensor data and motion-capture data as well as to send valve and pressure commands. Our controller code is operating in nonrealtime on an Ubuntu workstation. Data for the pressure sensors can be read at approximately 1,000 Hz, while data from the motion-capture system, which is used to estimate joint angles, is limited to approximately 300 Hz. Pressure for each bladder is controlled by an

Figure 1. A 14-DoF soft robot named King Louie with no rigid internal structure.
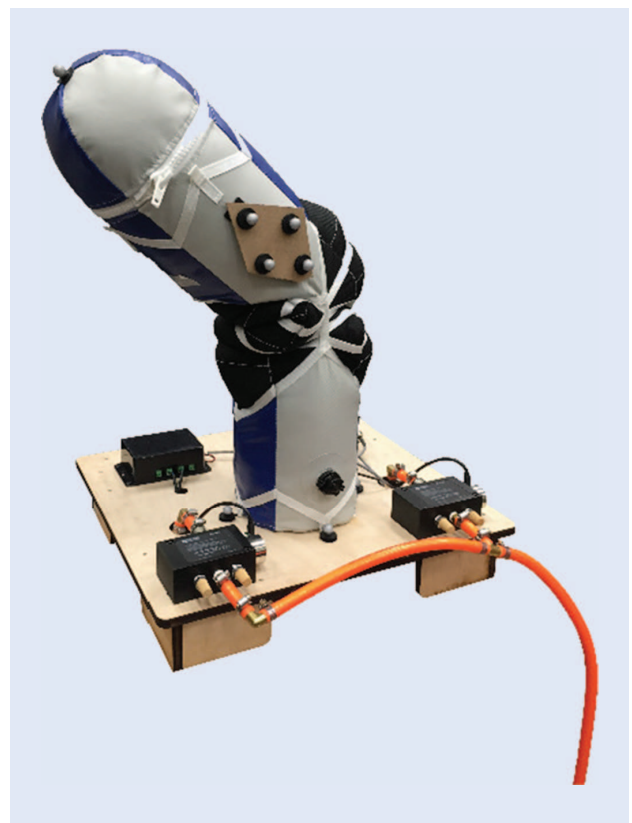

Figure 2. A single-DoF soft robot platform called a grub.

underlying PID controller also operating at approximately 1,000 Hz. Commanded pressures are published over the ROS, and the valves are actuated to achieve commanded pressure values.

Initial work for model development and control of the inflatable robotic systems was done with the grub. After we

performed initial analysis and testing on the one-DoF system (the grub) we applied the same methods to the more complex six-DoF arm on the 14-DoF humanoid system, King Louie. King Louie's arm configuration, orientation, and motion can be approximated with Denavit–Hartenberg (DH) parameters using an assumption of rigid links and compliant joints. We use DH parameters to estimate joint angles. However, these parameters are currently measured by hand. To calculate more accurate forward and inverse kinematics, we will need to perform a more accurate kinematic calibration for this soft robot platform. The focus of this article is instead on our results for direct joint control to known locations.
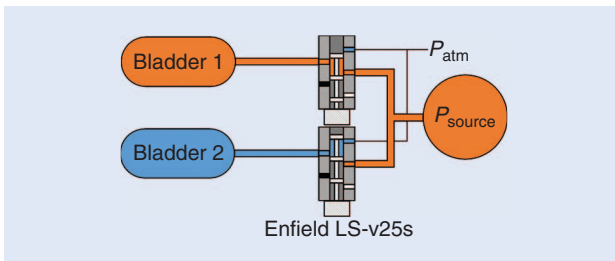
> **MPC has been demonstrated in robotics applications such as in the control of unmanned aerial and surface vehicles and more recently in robot manipulation with rigid links.**

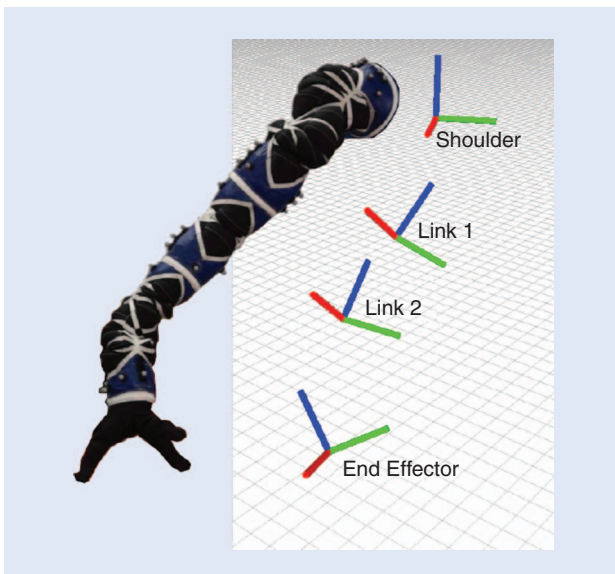**Figure 3.** A representative figure of a valve and actuator bladder configuration.

Figure 4. King Louie's arm, with motion-capture frames shown relative to the corresponding links.

Sensing and Joint-Angle Estimation

We use two different angle estimation methods for the grub and for King Louie. The joint angle for the grub is measured with an IMU located on the distal link. With the grub oriented at rest parallel to the gravity vector, the joint angle can be estimated from the measurements of two perpendicular accelerometers measuring the direction of gravity at any angle. We use a Kalman filter during actuation to produce a smoothed state estimate for both the joint angle and velocity.

King Louie has six controllable DoF for each arm. One of those six is the gripper. However, to reduce position error at the end effector, we modeled King Louie as having six joint angles (five controlled and one passive at the shoulder). These joint angles are estimated using data from a motion-capture system from Motion Analysis. The standard DH convention was used to place coordinate frames at each axis of rotation. We approximate all of our joints as rotary pin joints. The reason for choosing pin joint models over piecewise constant curvature models is that in initial tests we found both models to be comparable in terms of kinematic accuracy, and rotary models are obviously simpler. The motion-capture frames corresponding to the shoulder and links are seen in Figure 4. Infrared markers were placed at the shoulder and at the first, second, and third links, and nominal rotations were established between these marker frames and their corresponding DH frames.

The motion-capture system gives us the rotation from a global frame to the DH frame on each link, from which we can find the rotation between individual DH frames on links. We intentionally do not use the position information from the motion-capture system for joint configuration estimation, because we want to be able to use the same joint estimation approach with other orientation sensors that do not require multiple external cameras. The reflective markers for motion capture were located on the same links where King Louie has embedded IMUs. For this work, we are currently not using the IMU sensors with King Louie. However, the measurements we use from the motion-capture system for control are the link orientations represented as quaternions. This is the same data output that we expect to use with IMUs in future work.

Dynamic Soft Robot Models

To effectively control the soft robot platforms, we use model-based control methods that include the effect of multiple inputs on our desired output of each joint angle. This requires that we have reasonably accurate models of the system dynamics, where reasonably accurate is simply defined as being accurate enough that we achieve satisfactory control performance using that model.

For this research, we have made rigid-body assumptions to simplify the model for the soft robot system. These assumptions include there being no lateral or torsional deflection along the links, except in the passive joint that we model at King Louie's shoulder for joint-angle estimation. We also treat each joint and link as being a decoupled dynamic

system. Deviations from this approximate model will degrade the overall system performance, but we currently treat them as disturbances in the system. Although approximate, the results of MPC for the models presented here show that this rigid-body, decoupled model is sufficient for basic control of the system.

Link Dynamics

The differential equation that we use to describe the motion of a single link is that of an inverted pendulum:

$$I\ddot{q} + K_d\dot{q} + mg\frac{L}{2}sin(q) = \tau_a, \tag{1}$$

where q, $\dot{q}$, and $\ddot{q}$ are the joint angle, velocity, and acceleration, respectively; I is the moment of inertia of the link about its joint; K_d is a damping coefficient; m is the mass of the link; g is the gravity constant; L is the length of the link; and τ_a is the actuation torque applied. Initial system identification for the grub found that gravity effects were three orders of magnitude less than pressure effects, which include damping and stiffness. This led to making the assumption that gravity terms are negligible due to the low mass of the linkages, which reduced (1) to

$$I\ddot{q} + K_d\dot{q} = \tau_a. \tag{2}$$

Adding a more significant link-side load to a single joint or controlling a multijoint robot more accurately will require reexamining this approximation. An important improvement that we present in this article over our past work is a model relating the actuation chamber pressure and the resultant torque applied on the link.

The Link Torque Model

Our new torque model is developed with the idea that each bladder produces an independent torque on a joint that is a function of the pressure in the bladder and the current angle of the arm [Figure 5(a)]. The difference of these torques produces the total torque τ_a that is seen by the links. To test this model and find the form of the torque functions, we first measured the torque produced by a single bladder. For the test, we kept the base link of the grub fixed while we filled a given bladder to different pressures and measured the resultant force. Figure 5(a) shows a picture of the grub in the test rig and the force sensor, a setup where τ_1 was being characterized.

After taking the initial data, we realized that there was another torque term that we had not taken into consideration. This was the torque due to the stiffness of the fabric joint and internal body bladder. We measured this stiffness using the same setup as in Figure 5(a), but we left the actuation bladders empty and measured the passive torque output of the link at different angles. The results suggested that the torque from the joint stiffness ($\tau_{\text{stiffness}}$) can be approximated as a linear function of the joint angle, with its x and y intercept at the origin.
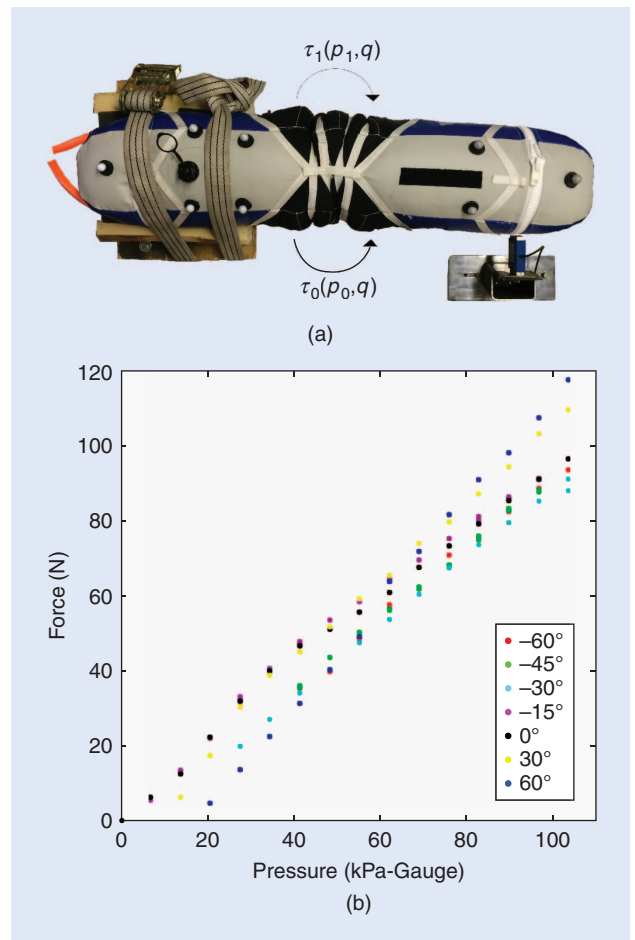

Figure 5. The torque model verification testing: (a) The testing setup with a force sensor mounted at 0°. (b) The actuator force at different pressures and angles.

We then reevaluated the data for the actuator while subtracting the force due to the joint stiffness and produced the results shown in Figure 5(b). Once again, the results suggested a linear relationship, but this time between force and pressure while being essentially independent of angle. It is important to note that while force was the value we measured, the relationship between the force output measured and the actual torque output of the system is just a scalar multiplier, which is the moment arm. The final description for the new torque models is shown in the following equations, where (3) is the torque due to the actuators, (4) is the torque due to the joint stiffness, and (5) is the full torque seen by the links.

$$\tau_0 - \tau_1 = \gamma_0 P_0 - \gamma_1 P_1, \tag{3}$$
$$\tau_{\text{stiffness}} = K_s q, \tag{4}$$
$$\tau_a = \tau_0 - \tau_1 + \tau_{\text{stiffness}}. \tag{5}$$

We initially found values for γ_i in (3) by using the data in Figure 5(b). In this case, γ_i was the slope of the line in this figure multiplied by a scalar. The scalar was the moment arm between the joint and the force sensor shown in Figure 5(a), and it converts the units on the slope from force (newtons) to torque (newton meters). K_s in (4) was found a similar way
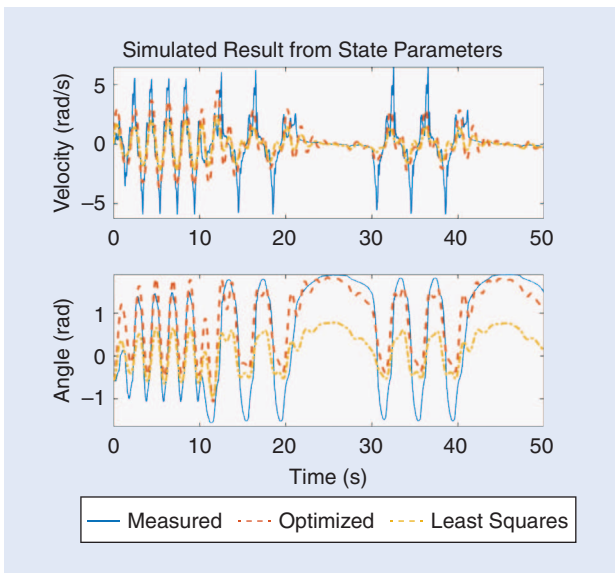
Figure 6. A forward simulation from initial conditions using estimated parameters for the state-space model.

from the data, where no pressures were applied and the output force was still measured at different angles. We then optimized and tuned these parameters as described later in the article. However, the form of the pressure-to-torque relationship that we found was essential to being able to improve our overall model and control.

Pressure Dynamics

Accurate models of the gas dynamics in the actuation chambers can be quite complicated. However, examining the pressure response to a step in commanded pressure for our PID pressure controller shows that the response is very close to a first-order, linear system. This means that we can represent the pressures dynamics as follows:

$$\dot{P} = -\alpha P + \beta P_d, \tag{6}$$

where P is the pressure in the chamber and P_d is the commanded pressure that is sent to the high-bandwidth PID controller. We can write equations like this for each individual actuation chamber on each joint and label them with subscripts 0 and 1 to differentiate between the chambers for a given joint.

Parameter Identification

The damping and torque coefficients were estimated by a gradient-free optimization minimizing the absolute error between the simulated angle and the measured angle. The initial parameters fed into the optimization were found using a least-squares estimation. Figure 6 shows a forward simulation using the least-squares and optimized estimated coefficients in the state-space model compared to measured data from the grub. The optimized parameters show better predictive accuracy for both the joint angle and velocity.

The coefficients for the pressure dynamics were found by comparing the first-order equation output to data collected from the grub with different step commands. These coefficients were then also tuned manually in line with the controller to improve the final performance. We expect that automating this procedure as a next step would significantly improve the performance and the time efficiency of the tuning.

Despite the optimized model performance in simulation, when we applied the model to the actual single-DoF system, the controller initially failed to reach the commanded angles with large magnitudes. Stiffness in the joint was initially set as a static, linear relation from the test platform, but the controller performance indicated that this was not the case. As such, we defined the stiffness to be a function of the joint angle, increasing the modeled stiffness when larger angles were commanded.

Given the single-joint dynamic model (including the effect of pressures in terms of torque) and the simplified pressure dynamics for the actuation chambers, we can discretize our state-space equations that we use for prediction using the bilinear transform, such that we have the following resultant difference equations instead of continuous time differential equations:

$$\begin{bmatrix} \dot{q}[k+1] \\ q[k+1] \\ P_0[k+1] \\ P_1[k+1] \end{bmatrix} = \boldsymbol{A}_d \begin{bmatrix} \dot{q}[k] \\ q[k] \\ P_0[k] \\ P_1[k] \end{bmatrix} + \boldsymbol{B}_d \begin{bmatrix} P_{0,d}[k] \\ P_{1,d}[k] \end{bmatrix}, \tag{7}$$

where k is a discrete time index. In terms of using this model for control, it assumes that the stiffness of the joint is constant over the time horizon for which we are predicting. This equation can then be used to predict the effect of commanded pressure inputs on the joint angle.

In this work, due to the low inertia of each link, we make the assumption that we can treat each link as an independent system, with no cross-coupling in the joints. We do this for two major reasons. The first is that it allows us to apply the models described in (7) to each joint individually after we have identified acceptable model parameters. This means that any actual cross-coupling and gravity-based torques are currently being treated as disturbances. We fully expect that doing system identification to identify these terms and at least including them as a known disturbance for each joint model would improve performance. The second reason we do this is that it allows us to solve an optimization that we describe next at the rate of 300 Hz. These higher control rates tend to improve the overall performance, which is not surprising.

Comparison with Past Control Performance

Because control for completely soft robots is a new area, we needed a straightforward way to compare the improvements in modeling and control. Specifically, the work

presented in [22], although most similar to ours, is still a very different type of actuator and does not provide dynamic performance for joint position tracking but instead uses a learning algorithm to attain desired positions in a plane. Therefore, we compare our recent advances against our own past work described in [10]. To differentiate, we will refer to the older controller as a *two-state MPC controller* and to the new controller presented in this article as a *four-state torque-based MPC controller*. By including pressures in the actuation chambers as states, we expected the performance of our soft robot control to improve when compared with our past work. We now describe at a high level the formulation of the new model predictive controller and then present the results of this comparison.

The Controller Formulation

The model predictive controller solves an optimization by predicting states over a time horizon that is T steps long while varying the pressure inputs to produce a trajectory resulting in the smallest cost subject to constraints. The discretized matrices A_d and B_d; the current states $\dot{q}[k]$, $q[k]$, $P_0[k]$, and $P_1[k]$; the previous inputs $P_{0,d}[k-1]$ and $P_{1,d}[k-1]$; the final goal joint angle q_{goal}; and the model constraints and weights are fed into an MPC solver at every time step. The cost function minimized across the horizon T is

$$\text{minimize} \sum_{k=0}^{T} \| q_{\text{goal}} - q[k] \|_Q^2 + \| \dot{q}[k] \|_R^2$$
$$+ \| P_0[k] - P_T \|_S^2 + \| P_1[k] - P_T \|_S^2, \qquad (8)$$

subject to the system model (7) as a constraint, as well as the following additional constraints:

$$q \geq q_{\min}, \qquad (9)$$
$$q \leq q_{\max}, \qquad (10)$$
$$P_{\min} \leq P_d \leq P_{\max}, \qquad (11)$$
$$|\Delta P_d| \leq \Delta P_{\max}, \qquad (12)$$

where Q, R, and S are manually tuned scalar weights, P_T is a target pressure, $q_{\min}$ and $q_{\max}$ are joint limits, $P_{\min}$ and $P_{\max}$ are bladder pressure limits, ΔP_d is the change in desired pressure from the previous time step, and $\Delta P_{\max}$ is the maximum change allowed in the desired pressure per time step. While simplified pressure dynamics are included in our model, the slew rate constraint on commanded pressure in (12) serves to prevent valve chatter.

We generated a solver for the MPC problem using CVX-GEN [12], a web-based tool for developing convex optimization solvers. The optimization solver written in C and subsequent Python code that calls the solver can be run at up to 300 Hz, predicting a trajectory horizon of $T = 20$ time steps (or 0.067 s) into the future. Once solved, the first time step from the optimized trajectory of desired pressures is made available over ROS. These desired pressures are received by the underlying pressure PID controller, and valve position commands are then sent to the individual valves. As shown in Figure 7, the current pressure states are measured and fed back into the pressure controller, while both the current estimated joint angle states and pressure states are fed into the MPC controller.

The Results for a Single DoF

In our preliminary modeling and control design, we focused on results for the single-DoF platform or grub. To compare performance, we commanded a series of 30° step-angle commands ranging from −60° to 60°, changing the command at time increments of 10 s. The resultant q of both controllers and commanded q_{goal} are plotted over time in Figure 8.

Compared to the previous two-state controller, utilizing pressure states in the four-state torque-based MPC controller significantly improved overall performance. As described in Table 1,

The soft robot platform is capable of repeatedly picking up an unknown object from the same location and placing it at a desired location.
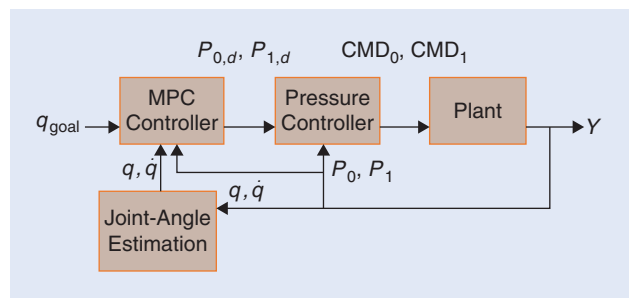

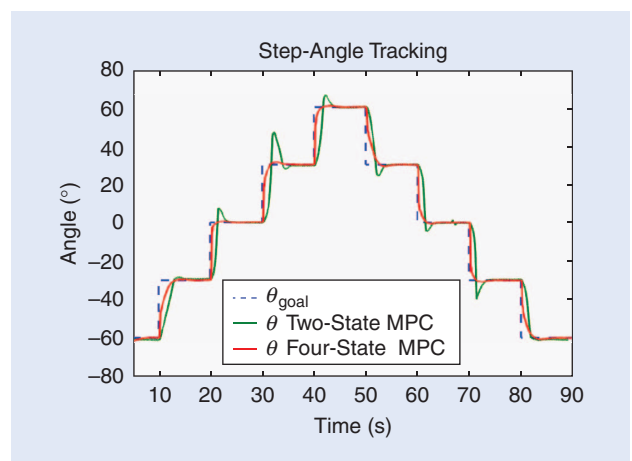**Figure 7.** An MPC control diagram for joint control.

Figure 8. The controller comparison results for a single joint.

Table 1. A performance comparison between the previous controller and the new controller.

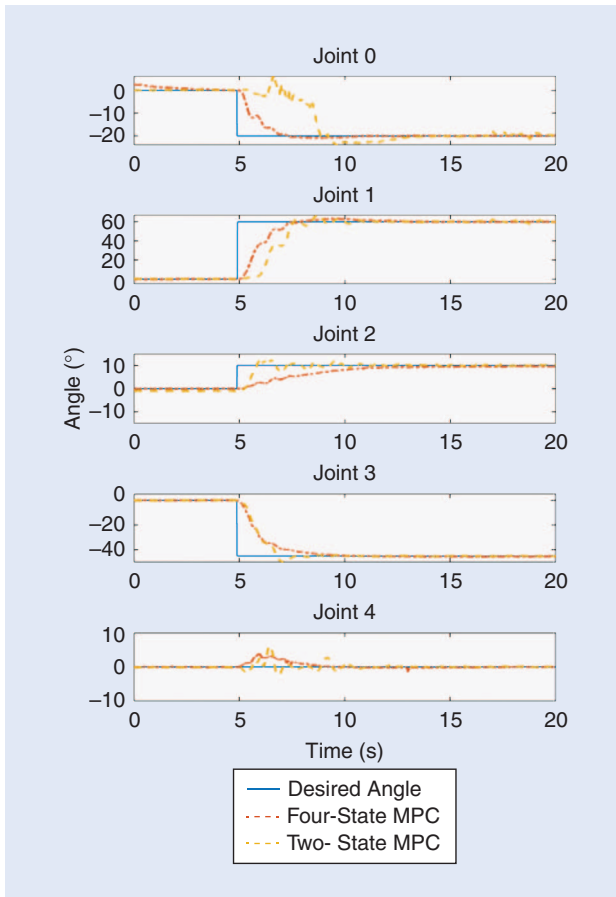|	Average Rise Time	Average Settling Time	Average % Overshoot
Two-state MPC	1.319 s	2.985 s	24.408%
Four-state torque-based MPC	1.360 s	2.592 s	2.587%
Improvement	−3.02%	15.13%	843.35%

Figure 9. A step response comparison between the current four-state MPC control response and the past two-state MPC control for King Louie's right arm.

Figure 10. King Louie moving to a set of specific joint angles: (a) King Louie, arm down and (b) King Louie, arm up.

joints was minimal, we tested our performance for controlling five DoF simultaneously. For comparison with our previous controller, the system response to a step input sent to each joint can be seen in Figure 9. In Figure 10(a) and (b), King Louie's arm can be seen at the commanded angles. This orientation was chosen to show that the robot can grab and manipulate objects in front of itself. Overall, we see that the new controller results in faster rise and settling times. This is especially true in joints 0 and 1, which are the two shoulder joints. These joints are especially difficult to control, given that we are neglecting the coupling and gravity effects.

Repeatability for Reaching in Free Space

Although the trials comparing the new controller to our previous controller for multi-DoF control are useful, they show nothing about the overall ability of the soft robot platform to repeatedly reach the same location. We do not expect soft robots to need the same precision as an industrial robot used for welding or other high-precision and high-accuracy tasks. However, a certain amount of precision may be necessary to perform useful tasks. We therefore tested the precision of our soft robot to reach to a commanded joint configuration.

We followed a process similar to the ISO 9283:1998 standard, but reached successively between only two locations for any given trial instead of three. In addition, although we allowed the robot to run and reach a steady-state temperature, the effect of this steady-state temperature for a pneumatic soft robot is much less clear, given that air is constantly being passed through the actuation bladders and that heating in the valves likely has a very different effect than the temperature of geared motors. These differences aside, we commanded the soft robot arm to go to 0° on all joints (Figure 11). After it settled, we commanded the arm to move to a given joint configuration and waited a set amount of time

the new controller produced a comparable 90% rise time, improved 15% in the settling time, and significantly decreased the percent overshoot. With angular step commands of only 30°, the two-state controller saw an average percent overshoot of 24.408% or 7.3°. The introduction of pressure dynamics reduced the average percent overshoot to just 2.587% or less than 0.78°.

The Results for Simultaneous Control of Multiple DoF

Again using the assumption that each link can operate independently of the others and that coupling between the
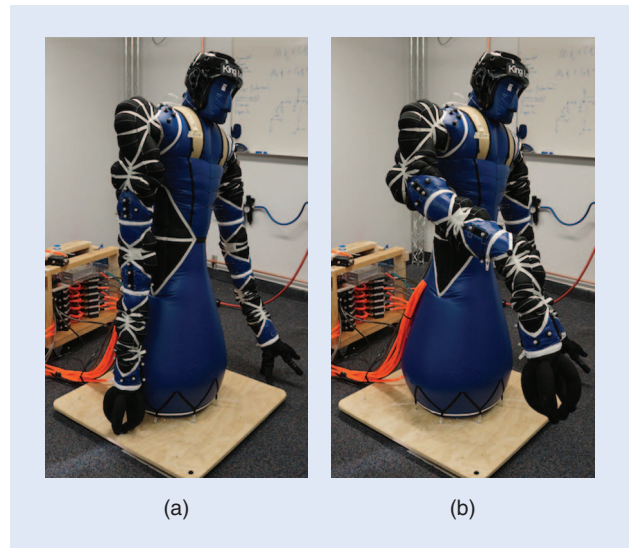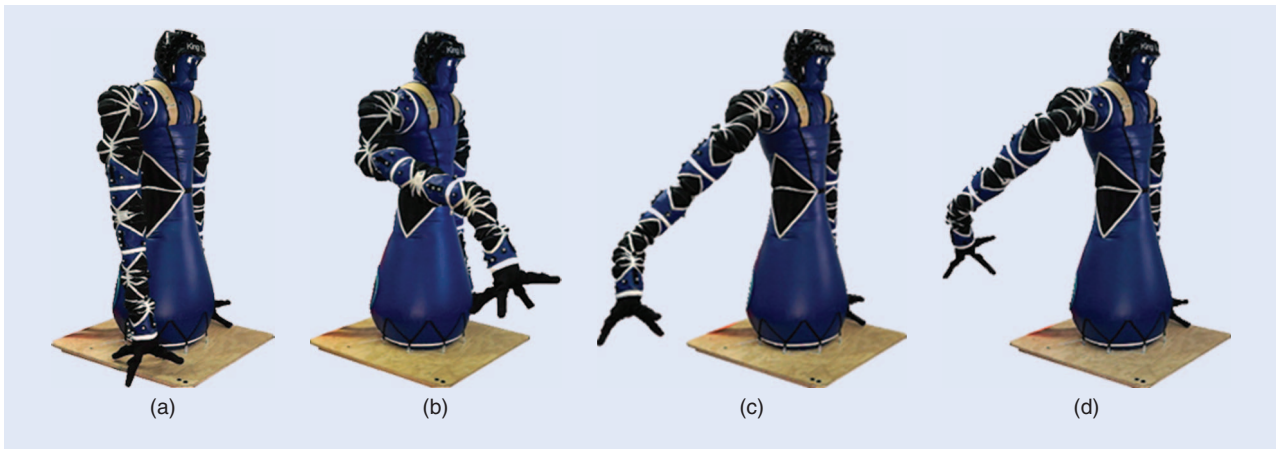
Figure 11. The initial pose and the three poses for the position controller repeatability tests: (a) King Louie, initial pose; (b) King Louie, pose 1; (c) King Louie, pose 2; and (d) King Louie, pose 3.

before recording the final Cartesian position. We chose three different joint configurations, as shown in Figure 11 and commanded the arm to go to each location ten times. For one trial out of the 30, we had an error in our controller and joint estimation code, which we discarded as an outlier. We then calculated repeatability in the standard way as the radial distance that would encompass 99.8% of end effector measurements, given exactly the same joint commands. To measure the end effector location, we used the motion-capture position measurement of the end effector link that we used to estimate repeatability but did not use for control. It is also important to note that although the hip joints were inflated, they were not controlled to a specific joint position. This means that we have good reason to believe that the repeatability values reported in Table 2 could be even better with minimal effort. A video of these reaching trials is available at https://youtu.be/4T-FN581RkA.

The Effect of Manipulating an Unknown Load

In addition to free space reaching and repeatability, we tested the repeatability of the controller and joint-angle estimation when manipulating a load of unknown mass. For a normal, rigid robot manipulator, moving an unknown mass would not be expected to change performance. However, for a soft robot like ours, an unknown and unmodeled mass may have a significant effect on control performance, which is what we wanted to test. A ball with an approximate diameter of 18 cm was placed at pose 2, as shown in Figure 11. King Louie picked up the ball from the location at pose 2, moved it to pose 1, which can be seen in Figure 12, and then released it. This was repeated ten times. The arm was successful at grabbing the object all ten times and successful at placing the ball in the bucket nine out of ten times. The only placement failure was when the ball hit the edge of the bucket. A video of these trials is available at https://youtu.be/p30jKn7_pV4. Although this is a simple trial, it shows that the soft robot platform is capable of repeatedly picking up an unknown object from the same location and placing it at a desired location.

Table 2. The repeatability measures for reaching to three different poses.

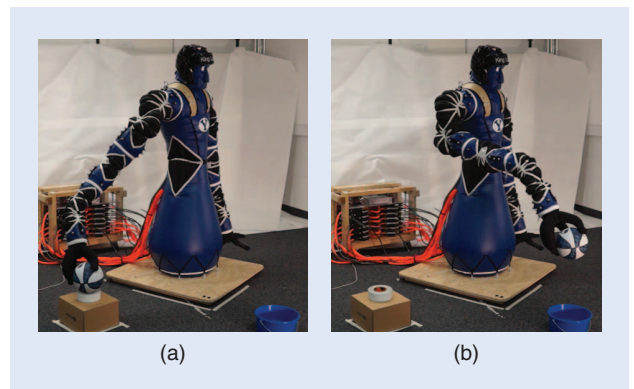| Pose	1	2	3
Repeatability (cm)	0.43	2.8	2.6

Figure 12. The initial pose and the final pose for King Louie grabbing a ball and placing it in a bucket: (a) the initial pose for grabbing the object and (b) the final pose for releasing the object.

Conclusions

In this article, we have presented an approach to manipulation with soft robots. In particular, we have shown the validity of a new method for joint-angle estimation for robots with compliant links and joints. We have also developed a torque model for inflatable joints with pneumatic actuation and have validated the model. We have been able to use this torque model with MPC of a single joint to significantly improve upon our past controller's performance. Applying this control to multiple DoF allowed us to demonstrate repeatability of approximately 2 cm for multiple locations in the robot's workspace and also allowed us to repeatedly perform a simple pick-and-place task.

The work we have presented is preliminary in the sense that we expect future progress to bring improvements for control performance with our soft robot. However, the fact

that a soft robot with no internal structure besides the structure provided by air can reach a location with a repeatability of approximately 2 cm is promising. Much of the current variability in end effector control comes from only controlling King Louie's hip joints to specific pressures instead of angles as well as limitations in our rigid-body kinematic models. We expect that specific improvements in modeling kinematics, link flexibility, and multi-DoF joint dynamics will provide immediate opportunities to improve the performance of our controllers. The fundamental ability presented in this article to control soft robots to a reasonable degree of precision will open avenues of research and applications for human–robot interaction and robot assistance that will change the way humans currently view and interact with robots.

References
[1] A. Albu-Schaffer, O. Eiberger, M. Grebenstein, S. Haddadin, C. Ott, T. Wimbock, S. Wolf, and G. Hirzinger, "Soft robotics," *IEEE Robot. Automat. Mag.*, vol. 15, no. 3, pp. 20–30, Sept. 2008.

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

[3] S. Sanan, J. Moidel, and C. Atkeson, "Robots with inflatable links," in *Proc. 2009 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, pp. 4331–4336.

[4] S. Sanan, M. H. Ornstein, and C. G. Atkeson, "Physical human interaction for an inflatable manipulator," in *Proc. 2011 Annu. Int. Conf. IEEE Eng. Med. Biol. Soc.*, pp. 7401–7404.

[5] 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. 2014 IEEE Int. Conf. Robot. Automat.*, pp. 2189–2196.

[6] A. D. Marchese, R. K. Katzschmann, and D. Rus, "Whole arm planning for a soft and highly compliant 2D robotic manipulator," in *Proc. 2014 IEEE/RSJ Int. Conf. Intell. Robots Syst.*, pp. 554–560.

[7] 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, June 2016.

[8] O. Ivlev, "Soft fluidic actuators of rotary type for safe physical human-machine interaction," in *Proc. 2009 IEEE Int. Conf. Rehabil. Robot.*, vol. 28359, pp. 1–5.

[9] I. Gaiser, R. Wiegand, O. Ivlev, A. Andres, H. Breitwieser, S. Schulz, and G. Bretthauer, "Compliant robotics and automation with flexible fluidic actuators and inflatable structures," in *Smart Actuation and Sensing Systems: Recent Advances and Future Challenges*, G. Berselli, R. Vertechy, and G. Vassura, Eds. Rijeka, Croatia: InTech, pp. 567–608, 2014.

[10] C. M. Best, J. P. Wilson, and M. D. Killpack, "Control of a pneumatically actuated, fully inflatable, fabric-based humanoid robot," in *Proc. 2015 IEEE-RAS Int. Conf. Humanoid Robots (Humanoids)*, pp. 1133–1140.

[11] M. T. Gillespie, C. M. Best, and M. D. Killpack, "Simultaneous position and stiffness control for an inflatable soft robot," in *Proc. 2016 IEEE Int. Conf. Robot. Automat.*, pp. 1095–1101.

[12] J. Mattingley and S. Boyd. (2012 Mar.). CVXGEN: A code generator for embedded convex optimization. *Optimization and Engineering* [Online]. 13(1), pp. 1–27. Available: http://dx.doi.org/10.1007/s11081-011-9176-9

[13] D. H. Shim, H. J. Kim, and S. Sastry, "Decentralized nonlinear model predictive control of multiple flying robots," in *Proc. 42nd IEEE Conf. Decision Control*, vol. 4, 2003, pp. 3621–3626.

[14] P. Abbeel, A. Coates, M. Quigley, and A. Y. Ng, "An application of reinforcement learning to aerobatic helicopter flight," in *Proc. 19th Advances Neural Informat. Process. Syst. (NIPS)*, 2006, pp. 1–8.

[15] A. S. K. Annamalai, R. Sutton, C. Yang, P. Culverhouse, and S. Sharma, "Robust adaptive control of an uninhabited surface vehicle," *J. Intell. Robot. Syst.*, vol. 78, no. 2, pp. 1–20, May 2015.

[16] M. D. Killpack and C. C. Kemp, "Fast reaching in clutter while regulating forces using model predictive control," in *Proc. 2013 13th IEEE-RAS Int. Conf. Humanoid Robots (Humanoids)*, pp. 146–153.

[17] M. D. Killpack, A. Kapusta, and C. C. Kemp, "Model predictive control for fast reaching in clutter," *Auton. Robots*, vol. 40, no. 3, pp. 537–560, 2016.

[18] T. Erez, Y. Tassa, and E. Todorov, "Infinite-horizon model predictive control for periodic tasks with contacts," *Robotics: Sci. Syst. VII*, pp. 73–80, June 2012.

[19] L. Rupert, P. Hyatt, and M. D. Killpack, "Comparing model predictive control and input shaping for improved response of low-impedance robots," in *Proc. 2015 15th IEEE-RAS Int. Conf. Humanoid Robots (Humanoids)*, pp. 256–263.

[20] S. Sanan, P. S. Lynn, and S. T. Griffith, "Pneumatic torsional actuators for inflatable robots," *J. Mechanisms Robot.*, vol. 6, no. 3, pp. 031003–031003-7, 2014.

[21] S. Sanan, "Soft robots for safe physical human interaction," Ph.D. dissertation, Robotics Institute, Carnegie Mellon Univ., Pittsburgh, PA, 2013.

[22] A. D. Marchese, R. Tedrake, and D. Rus, "Dynamics and trajectory optimization for a soft spatial fluidic elastomer manipulator," in *2015 IEEE Int. Conf. Robot. Automat. (ICRA)*, New York, pp. 2528–2535.

Charles M. Best, Mechanical Engineering Department, Brigham Young University, Provo, Utah. E-mail: irbest111@gmail.com.

Morgan T. Gillespie, Mechanical Engineering Department, Brigham Young University, Provo, Utah. E-mail: morgan.t.gillespie@gmail.com.

Phillip Hyatt, Mechanical Engineering Department, Brigham Young University, Provo, Utah. E-mail: phyatt16@gmail.com.

Levi Rupert, Mechanical Engineering Department, Brigham Young University, Provo, Utah. E-mail: levi.rupert@gmail.com.

Vallan Sherrod, Mechanical Engineering Department, Brigham Young University, Provo, Utah. E-mail: vallan.sherrod@byu.edu.

Marc D. Killpack, Mechanical Engineering Department, Brigham Young University, Provo, Utah. E-mail: marc_killpack@byu.edu.