Adaptive Control for Soft Robot Manipulators with Unknown Loads

Jonathan S. Terry Randal W. Beard Marc D. Killpack

***Abstract*—Soft compliant robots may greatly improve safety, especially when interacting with humans, over traditional robot structures. However, compliance makes accurate position control difficult, especially when working with unknown loads or as the system dynamics change over time. In this paper, we present a novel method of coupling Model Reference Adaptive Control (MRAC) with Model Predictive Control (MPC) for platforms with antagonistic actuators. We demonstrate its utility on a fully inflatable, pneumatically actuated soft robot manipulator, and compare control performance with and without MRAC when significant weight is attached to the end effector. Specifically, for tests on real soft robot hardware, our adaptive control showed a 50% average reduction in step input tracking error when compared to the performance of the nominal controller.**

I. Introduction

As the robotics field continues to expand into new areas of application, there is an increasing interest in developing robots that are safe around humans and capable in unmodeled environments. Soft, compliant robots provide a possible solution to this problem. A soft, lightweight, compliant robot is inherently more safe for both the robot and everything with which it may come in contact. The unique advantages of soft, inflatable robots over rigid robots for specific applications are what motivate this research. Applications include health care, living assistance, space exploration, search and rescue, orthotics, and prosthetics [1]–[3]. As shown in [1], a variety of soft grippers, actuators, and structures have been designed and built with the intent of moving robotics into environments that are not currently feasible. Generally, work in this area is pursued with the intent of seeking an alternative approach to a rigid structural design.

Studies have been done to evaluate and quantify the dangers of unintentional impacts from rigid robots. In [4] the Head Injury Criterion (HIC) is identified as a limit for serious head injury upon robot impact, where inertia of the robot is a major driving contributor. This indicates that with all operating parameters constant, a reduction in robot inertia directly reduces the HIC rating. Also in [5] and [6], it has been shown that contact forces from inflatable links can be controlled with cable driven actuators. The ability of soft robots to control or inherently limit forces can significantly reduce injuries or damage from unintentional impact.

In this paper, we focus on improving control for a specific type of soft robot platform called King Louie (Figure 1). King Louie is soft, fully inflatable, and has four Degrees of Freedom (DoF) in each arm. Each joint is pneumatically actuated with antagonistic pressure chambers. This platform is fabric-based, lightweight (each arm weighs about five pounds) and compliant, and therefore inherently safe around humans. It is also much less likely to damage itself or its environment upon accidental impact. This type of platform is one possible solution to the problems discussed above.

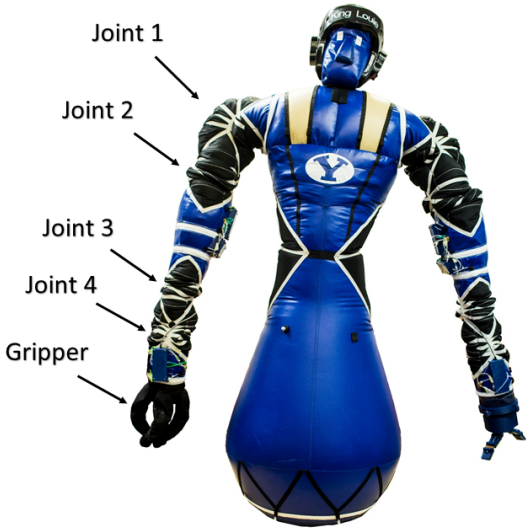

Fig. 1: King Louie is a fully inflatable, pneumatically actuated platform with four degrees of freedom in each arm.

However, the benefits of this platform also come with difficulties which are almost irrelevant in traditional robots. The compliance in the links and joints, coupled with the complex dynamics of pneumatic actuation, make the arms difficult to accurately control. Each arm has a maximum payload of about three pounds, which is a significant percentage of the total arm weight, so picking up a relatively heavy object will greatly change the dynamics of the arm. Additionally, when the robot is deflated and then re-inflated, the structural chambers have a tendency to settle in slightly different positions than before, which has an impact on the dynamics and kinematics of the arm.

Work has been done previously on this platform, where models and control methods were developed, a PID controller for pressure in the actuation chambers was designed, and a Model Predictive Controller (MPC) for joint angle control was implemented [7], [8]. This paper addresses the problem of maintaining control performance when significant weight is grasped in the end effector and the approach is general enough to be applicable to other changes in structure and dynamics of the platform as well. Our main contribution is the development of a form of Model Reference Adaptive Control (MRAC) which couples with MPC for soft robots with antagonistic actuators and compliant links.

A. Related Work

One reason why robots with rigid joints and actuation have been used for so long is that low level system dynamics and sensing methods are well developed. With a soft robotic system, the dynamics are not straightforward and for each design new methods must be developed or learned through data to describe how the system behaves when actuated. An example of this is in [9], [10], and [11] where it was found that planning is viable for elastomer actuators using dynamic and constant curvature kinematic models. Designs for rotary, fabric-based, pneumatic actuated joints were proposed in [12] and the actuators used in this work are based on those designs. One purpose of our research is to develop models and general methods of control for robots like King Louie. The lack of literature on the control of inflatable structures despite a wide range of applications suggests a viable and interesting area for research.

Different forms of adaptive control methods have been developed for use with MPC both on a theoretical basis and also for specific platforms. Chowdary et al [13] used a form of MRAC with MPC for aerospace applications with fast dynamics. Fukushima et al [14] developed a general method for on-line system identification to work in conjunction with MPC. Adetola et al [15] and Chen et al [16] both developed Lipshitz-based adaptive MPC methods.

For platforms with robot manipulators, a typical adaptive control approach is on-line parameter identification using a regressor matrix (see [17]–[20]). MRAC has also been used for robot manipulators in [21]. Tahia et al [19] and Tonietti et al [20], cited above, both worked on manipulators with antagonistic actuators, similar to the manipulators used in this paper. Their manipulators, however, had rigid links with compliant actuation, whereas the platforms used in this paper have compliant links as well as compliant actuators. Medrano-Cerda et al [22] also did their work on manipulators with antagonistic actuators with rigid links, but the control in their work used an adaptive pole placement technique. Link compliance requires extra consideration in control development, especially when loads are added to the manipulator. This is a significant motivation for the research presented in this paper.

II. Methodology

A. Platform Description

The platforms used for this research are a single DoF joint called a Grub (Figure 2) and a humanoid robot called King Louie (Figure 1), with four DoF in each arm. Both were developed and built by Pneubotics, an affiliate of Otherlab. Besides internal electronics such as IMU and pressure sensors, these platforms are entirely made of ballistic nylon fabric with internal bladders to prevent air leakage. The structure of these robots comes from an inflatable bladder which is pressurized to between 2-3 PSI gauge. At each joint, there are antagonistic actuators which can be filled to pressures between 0-20 PSI gauge. These actuators are similar to the designs described in [12].

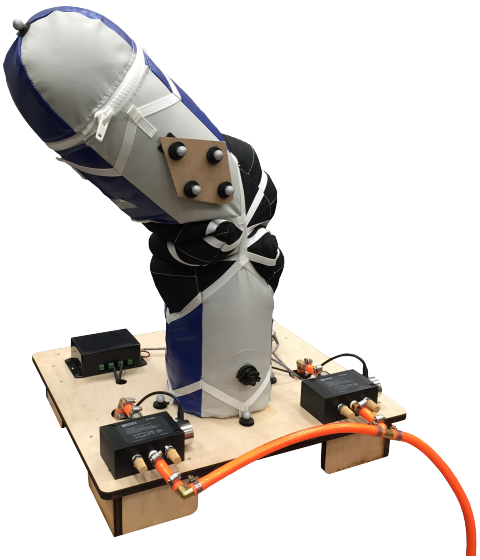

Fig. 2: The Grub is a single degree of freedom, fully inflatable, pneumatically actuated platform.

A single pressure source provides the air for each of the actuators, and each actuator's pressure is controlled using an Enfield LS-v25 spool valve. Pressure data for each actuator is read using Honeywell HSCDRNN100PGSA3 pressure sensors. An underlying PID controller running at 1 kHz controls pressure in each actuator.

Joint angles for the Grub are determined using IMU (inertial measurement unit) located on the link. Joint angles are measured on King Louie by using HTC Vive Trackers. The HTC Vive system uses two emitter/sensor Lighthouses to determine the orientation of each Tracker. Trackers were placed on links of King Louie's arms and used to back out each joint angle.

Communication between the sensors and controller is achieved using ROS (Robot Operating System) operating in non-real time on an Ubuntu workstation.

B. Control Methods

Model Predictive Control (MPC) is an optimal control method originating from the process industry in chemical production and oil refinement [23] and more recently in robotics [24]–[26] and UAV research [27]–[29]. Recent advances in computing power and optimization techniques such as those described in [30], [31] have made this possible. MPC allows for the minimization of a customizable cost function across a discrete finite time horizon given specific constraints. MPC not only allows for model-based control but also for the consideration of real-world constraints such as joint limits, pneumatic valve actuator limitations, and hardware pressure limits.

The most significant constraint for the MPC optimization used in this work is the state space model. Previously we presented an effective means of linearizing and decoupling the model, called the Coupling-Torque method [8]. This method was utilized and built upon for the development of the adaptive control scheme discussed in this paper.

Joint angle control for King Louie (Figure 1) is run individually on each joint, with each joint-level controller receiving information about torques generated from the motion of other links and treating them as disturbances, as described in [8]. This is referred to as the Coupling-Torque (CT) method, and is summarized here.

The state equations used for the Grub and King Louie are

$$\dot{\mathbf{x}} = \mathbf{A}(\mathbf{x})\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{M}^{-1}(\mathbf{q})\mathbf{G}(\mathbf{q}) \tag{1}$$

where

$$\mathbf{x} = \begin{bmatrix}\dot{\mathbf{q}} & \mathbf{q} & \mathbf{p}^+ & \mathbf{p}^-\end{bmatrix}^T \tag{2}$$

$$\mathbf{u} = \begin{bmatrix}\mathbf{p}^+_{des} & \mathbf{p}^-_{des}\end{bmatrix}^T \tag{3}$$

$$\mathbf{A} = \begin{bmatrix} -\mathbf{M}^{-1}(\mathbf{C}+\mathbf{K_d}) & -\mathbf{M}^{-1}\mathbf{K_p} & \mathbf{M}^{-1}\mathbf{K}^+_\gamma & -\mathbf{M}^{-1}\mathbf{K}^-_\gamma \\ \mathbf{I} & \mathbf{0} & \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{K}^+_\alpha & \mathbf{0} \\ \mathbf{0} & \mathbf{0} & \mathbf{0} & \mathbf{K}^-_\alpha \end{bmatrix} \tag{4}$$

and

$$\mathbf{B} = \begin{bmatrix} \mathbf{0} & \mathbf{0} \\ \mathbf{0} & \mathbf{0} \\ \mathbf{K}^+_\beta & \mathbf{0} \\ \mathbf{0} & \mathbf{K}^-_\beta \end{bmatrix} \tag{5}$$

where $\mathbf{q}$ and $\mathbf{p}$ are vectors of joint angles and pressures, respectively, $\mathbf{p}^+$ are the pressures that yield positive torques, $\mathbf{p}^-$ are the pressures that yield negative torques, $\mathbf{M}$ is the mass matrix, $\mathbf{C}$ is the Coriolis matrix, $\mathbf{G}$ is gravity, and $\mathbf{K_p}$ and $\mathbf{K_d}$ are diagonal matrices that model the joint stiffness and damping. All other $\mathbf{K}$ matrices are diagonal matrices that contain the different coefficients for the respective joint pressures and were found through empirical data and controller tuning. Note that for the Grub $\mathbf{x}$ is a vector of four elements and $\mathbf{u}$ is a vector of two elements, whereas for King Louie $\mathbf{x}$ is a vector of 16 elements and $\mathbf{u}$ is a 2×4 matrix.

Dynamic coupling occurs because of the torques generated from the motion of other links in the arm, and is expressed mathematically via the Coriolis and inertial terms. The CT method treats these coupling torques as a constant known disturbance over the horizon duration.

The general equation that MPC uses for its forward prediction for joint i, $\forall i = 1 \ldots n$ where n is the number of joint, when using the CT method is

$$\mathbf{x}_{k+1,i} =_{mpc,i} \mathbf{x}_{k,i} + \mathbf{B}_{mpc,i}\mathbf{u}_{k,i} + \mathbf{d}_i \tag{6}$$

where $\mathbf{x}_{k,i}$ is the states directly pertaining to link i, and $\mathbf{A}_{mpc,i}$ and $\mathbf{B}_{mpc,i}$ are composed of the elements of $\mathbf{A}$ and $\mathbf{B}$ that are multiplied by the states and inputs directly pertaining to Link i, and $\mathbf{d}_i$ is the coupling torque generated from the motion of the other links. When implemented in the controller, $\mathbf{d}_i$ is calculated at the current states and then held constant over the MPC horizon. For a more in-depth discussion of how each of these terms are calculated, see [8].

III. Parameter Estimation

A common method for adaptive control is called Model Identification Adaptive Control (MIAC), where algorithms are designed to match experimental data with a model by manipulating model parameters. Generally, a more accurate model will result in improved model-based control performance. This strategy was employed to estimate the rotational inertia of the Grub (Figure 2), which is modeled as an inverted pendulum. MPC was used to control the joint angle, and a Recursive Least-Squares (RLS) algorithm was used to estimate the inertia. The Grub's construction makes it difficult to predict an actual inertia value, so a 3-pound weight was attached to the end, which should be the dominate contributor to the system's inertia (the moving soft link weighs about 1 pound). Treating the added weight as a point mass, we estimate that the system has a rotational inertia of roughly 0.1 $kg \cdot m^2$. The RLS algorithm settles at values to within $\pm 5\%$ of our estimated value within 0.4 seconds after the first step input, and the controller performance with the estimated rotational inertia value can be seen in Figure 3 (blue dashed line).

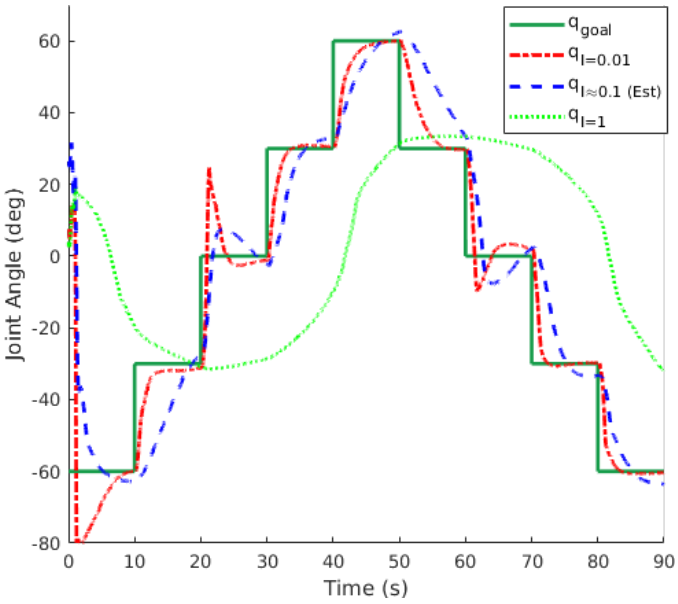

Fig. 3: Joint angle control on the Grub, varying the model's inertia value, with a three-pound weight at the end of the link.

However, in the development process, we discovered that manually setting an unrealistically low inertia value, such as 0.01 $kg \cdot m^2$, significantly improved control performance. Manually setting an unrealistically high inertia value, such as 1 $kg \cdot m^2$, significantly decreased control performance. Figure 3 shows control performance when $I = 0.01$ $kg \cdot m^2$ (red line), $I = 1$ $kg \cdot m^2$ (light green line), and when I is estimated using RLS (blue line). This indicates that, at least for this system, performance-based adaptive control would be more effective than a system identification strategy. This may be particularly

true if our selected parameterization of the physical model is limited in what it can represent. Note that, in Figure 3, the Grub is at 0 deg when it is pointed straight up, so momentum and gravity drive the link to extreme overshoot when $q_{goal} =$ 0 deg, especially in the case using a low value for I.

IV. Model Reference Adaptive Control

We identified Model Reference Adaptive Control (MRAC) as a viable option for a performance-based adaptive control method and used Lavretsky's development for MRAC (see [32]) for our soft robot system. However, we also made adjustments so that we could continue to use MPC as the main control method.

For this implementation of MRAC/MPC, Eq 6 is modified to

$$\mathbf{x}_{k+1,i} = \mathbf{A}_{mpc,i}\mathbf{x}_{k,i} + \mathbf{B}_{mpc,i}\mathbf{u}_{k,i} + \mathbf{d}_i - \Omega_i \tag{7}$$

The adaptive laws used to find Ω_i are

$$\dot{q}_{ref,i} = \alpha_i(q_{goal,i} - q_{ref,i}) \tag{8}$$

$$q_{err,i} = q_{ref,i} - q_i \tag{9}$$

$$\dot{\theta}_{i,j} = \begin{cases} \Gamma_{i,j}\Phi_{i,j}q_{err,i} - \sigma_{i,j}|q_{err,i}|\theta_{i,j} & \forall q_{err,i} \geq 1\ deg \\ 0 & \forall q_{err,i} < 1\ deg \end{cases} \tag{10}$$

$$\Omega_i = \sum_j \theta_{i,j}\Phi_{i,j} \tag{11}$$

$$\forall j = 1 \dots J_i$$

where i designates a specific link, q_{goal} is the desired joint angle, q_{ref} is a desired reference trajectory, Φ is the regressor (defined in Eq 13 for the Grub and in Eq 14 for King Louie), J_i is the number of parameters to be adapted, and α, Γ and σ are gains. The purpose of q_{ref} is to smooth out the transition between step changes in q_{goal}. Otherwise, abrupt step changes can cause stability issues for MRAC. Larger Γ values result in more quickly changing values for the adaptive parameters θ, whereas σ acts as a spring, preventing θ from growing too large. In Eq 10, the breakpoint at $q_{err} = 1$ deg serves as a deadband so that adaptive parameters do not respond to small changes in q. The overall control scheme is summarized in Figure 4:

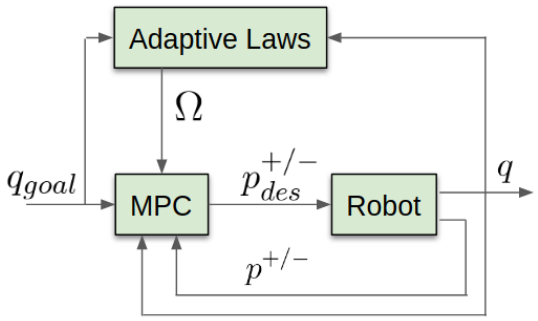

Fig. 4: MRAC with MPC diagram.

In contrast, Lavretsky's MRAC formulation has the following method for incorporating the adapted parameters:

$$u_{mrac,i} = u_i - \Omega_i \tag{12}$$

where u_i is the input determined by some controller, and $u_{mrac,i}$ is the adapted input. Lavretsky's method modifies the input directly, whereas our method passes Ω in to MPC as a model parameter. This directly affects the solutions that MPC finds for control, while still respecting the constraints of the inputs and states.

A. Hardware Implementation on the Grub

Adaptive control was first implemented on the Grub (Figure 2). Adaptive gains (Γ and σ) were tuned with no extra mass attached, with Φ defined for the Grub as

$$\Phi_{Grub} = \begin{bmatrix} a_{1,1}\dot{q} & a_{1,2}q & a_{1,3}p^+ & a_{1,4}p^- & I^{-1}G & I \end{bmatrix} \tag{13}$$

where $a_{m,m}$ are the entries in the mth row and nth column of $\mathbf{A}$ from Eq 4, x_i is the ith element of $\mathbf{x}$ from Eq 2, I is inertia (equivalent to I from Section III) and G is gravity from Eq 1. These parameters were selected based on their impact on control performance during the tuning process.

For comparison to the nominal controller, a three-pound mass was added to the end of the link after the initial adaptive parameter tuning. The results are shown in Figure 5 and Table I. Given that some of the step input responses never attain steady-state, the q_{goal} Error metric is shown instead of Steady-State Error in Table I. This metric compares the q to q_{goal} at the final value, measured immediately before the next step input. Similarly, because some commands have significant q_{goal} Error, the 95% Rise Time metric shows rise time to the final value instead of rise time to q_{goal}. The nominal case is the same as in Figure 3 where I = 0.1 $kg \cdot m^2$. Figure 5 indicates some improvement with MRAC/MPC, especially with regards to the overshoot when $q_{goal} = 0$ deg. Table I shows some improvement in all three metrics.

In contrast, MRAC/MPC on King Louie shows significant performance improvement, especially when weight is added to the end effector. One explanation for this difference is that the success of MRAC/MPC on King Louie relies on adapting dynamic parameters that pertain to the motion of other links of the arm, whereas these dynamic couplings between joints do not exist with the Grub and therefore cannot easily be improved upon. This suggests that a different adaptive control approach may be more beneficial for improving performance of the Grub. However, we show in this paper that our approach is able to improve performance more significantly for a more complicated multiple degree of freedom system such as King Louie, so improving adaptive control further on the Grub was not pursued.

B. Hardware Implementation on King Louie

Adaptive control was implemented on the last three links of a King Louie arm (see Figure 1). In preliminary testing, it became clear that Joints 1 and 2 had strong coupling effects, due to causes unique to this platform. This could include

TABLE I: Grub Experiment Results

Controller	Mass (lb)	RMS Error (deg)	$\mathbf{q_{goal}}$ Error (deg)	95% Rise Time (s)
Nominal	3	17.22	2.94	6.29
MRAC/MPC	3	15.90	2.10	5.97

physical strain induced by pneumatic hoses that pass along the entire length of the arm, air flow limitations, or joint angle estimation. To focus on the contribution of this paper, we performed final experiments on the last three joints only. For this reason, equations and plots omit Link 1. Adaptive gains (Γ and σ) were tuned with no additional weight in the end effector, with Φ for King Louie defined as

$$\Phi_{KL} = \begin{bmatrix} a_{2,6}q_2 & a_{1,5}q_1 & a_{1,5}q_1 \\ g_2 & a_{1,9}p_1^+ & a_{1,9}p_1^+ \\ 0 & a_{3,7}q_3 & a_{3,7}q_3 \\ 0 & a_{3,11}p_3^+ & a_{3,11}p_3^+ \\ 0 & a_{3,15}p_3^- & a_{3,15}p_3^- \\ 0 & a_{4,8}q_4 & a_{4,8}q_4 \\ 0 & a_{4,12}p^+4 & a_{4,12}p^+4 \\ 0 & a_{4,16}p^-4 & a_{4,16}p^-4 \\ 0 & g_2 & g_2 \\ 0 & g_3 & g_3 \end{bmatrix} \quad (14)$$

where $a_{m,n}$ and $b_{m,n}$ refer to the element in row m and column n of $\mathbf{A}$ from Eq 4 and $\mathbf{B}$ from Eq 5, respectively, and g_i is the result of the product of $\mathbf{M}^{-1}(\mathbf{q})\mathbf{G}(\mathbf{q})$ (see Eq 1) that corresponds to Link i. The jth column of Φ contains dynamic parameters to be adapted for Link $j+1$. Note that the zeros in the first column indicate that fewer dynamic parameters were adapted for Link 2. Also note that Columns 2 and 3 are identical.

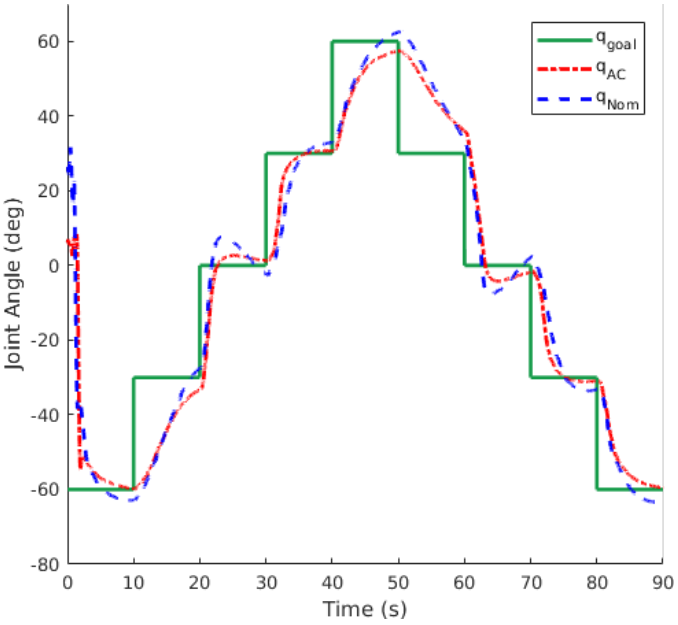

Fig. 5: Adaptive vs Nominal control for the Grub with a three-pound weight added to the end of the link.

Control was implemented while the arm moved for several minutes, allowing the adaptive parameters θ to settle. Values for θ were then saved so that future trials could start with the adaptive gains found from this process, and are referred to here as $\theta_{m=0}$. Figure 6 shows the control performance of an example trial as a result of this process, comparing nominal to adaptive control with no weight in the end effector.

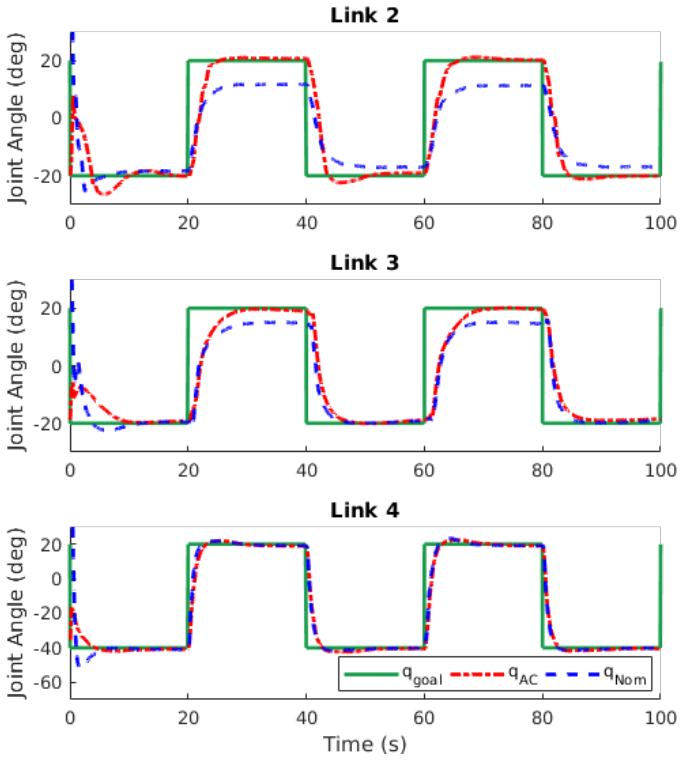

Fig. 6: Adaptive vs Nominal control for three links of King Louie's arm with no weight in the end effector.

Similar tests were run with one, two or three pounds grasped in the end effector (three pounds is approximately the max payload of each arm). These trials were also started with the $\theta_{m=0}$ values from the no-weight scenario. For the trials with extra weight attached, the controller was given no indication of the extra weight added. Figure 7 shows data from a trial with three pounds in the end effector. If this experiment was run for a few minutes, the adaptive laws would have time to adjust θ values and settle to ideal values for the three-pound scenario ($\theta_{m=3}$). This would improve performance further, but even without running it that long, performance is still significantly better than in the nominal case (see Table II). Note that the relatively poor performance of MRAC/MPC on Link 3 is due to actuator saturation where our commanded chamber

TABLE II: King Louie Experiment Results

Controller	Trajectory	Mass (lb)	RMS Error (deg)			q_{goal} Error (deg)			95% Rise Time (s)		
			Link 2	Link 3	Link 4	Link 2	Link 3	Link 4	Link 2	Link 3	Link 4
Nominal	Step	0	11.44	11.39	10.53	7.38	4.32	0.64	8.96	8.27	3.33
MRAC/MPC	Step	0	10.96	12.20	10.58	4.04	3.70	0.34	5.20	10.77	3.30
Nominal	Step	3	14.28	13.45	12.03	8.66	4.98	0.71	8.80	8.21	3.54
MRAC/MPC	Step	3	12.35	14.32	10.43	0.35	4.52	0.38	5.01	10.57	3.33
Nominal	Sin	3	25.28	14.57	11.47						
MRAC/MPC	Sin	3	20.88	16.78	10.82						

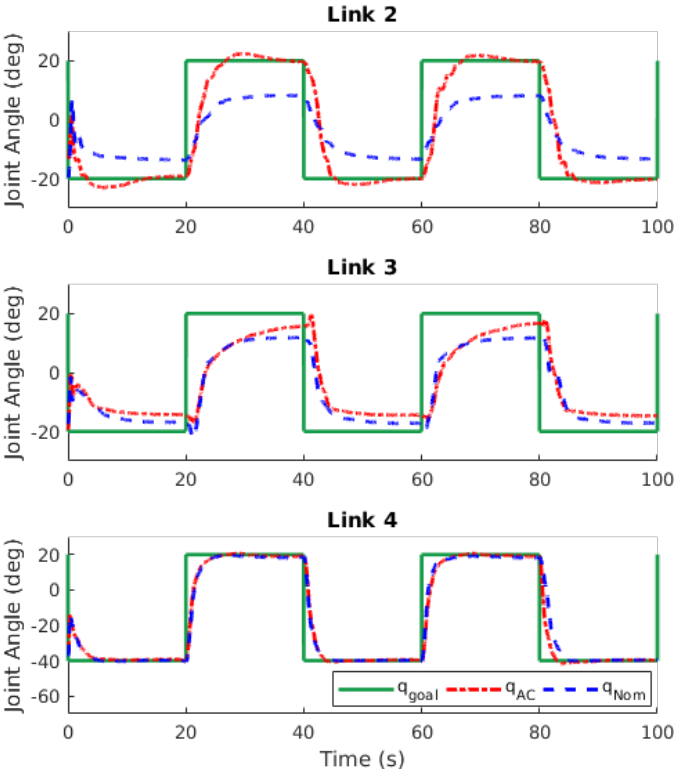

Fig. 7: Adaptive vs Nominal control for three links of King Louie's arm with a 3-pound weight (unknown to the controller) in the end effector.

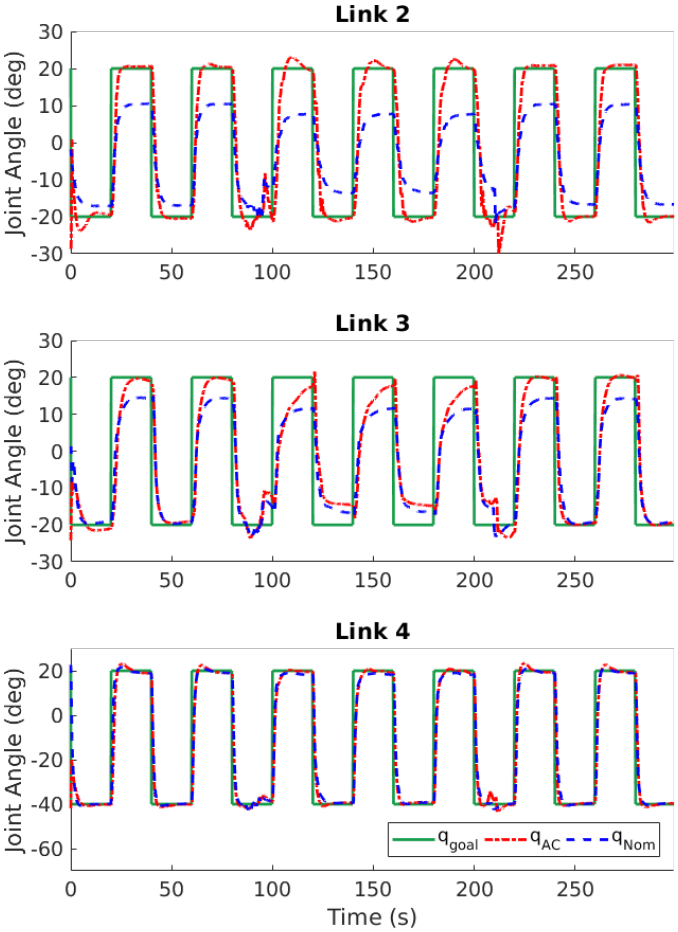

Fig. 8: Adaptive vs Nominal control for three links of King Louie's arm. A 3 lb mass was attached to the end effector at about 90s and removed at about 210s. This resulted in significant disturbances at those times, which can be seen in the plot.

pressures were equal to the source pressure or atmospheric pressure.

Finally, a trial was run starting with $\theta_{m=0}$ and no weight in the end effector. At Time $\approx$ 90 seconds, a 3 lb mass is attached to the end effector, and was removed at Time $\approx$ 210 seconds. This gives an idea of the control performance with a heavy object in a pick-and-place task. The results of this trial are shown in Figure 8. The process of attaching and removing the mass was done manually by the operator, and so resulted in significant joint angle disturbances, which can be seen in Figure 8 around 90 and 210 seconds.

A similar comparison is made where q_{goal} is a sine wave trajectory instead of step inputs (see Figure 9). Here, a 3 pound mass is loaded to the end effector for the entire run. Again, the trial was started with the $\theta_{m=0}$ parameters. Adaptive control gives significant improvement for Link 2, while adaptive control performance is slightly worse for Link 3 and nearly identical for Link 4. Poor performance for both controllers on Link 3 at the positive peaks of the wave is due to actuator saturation. The same is true for Link 4, but to a smaller extent. Note that q_{ref} is set equal to q_{goal} for MRAC/MPC with a sine wave trajectory.

Numerical results for these hardware experiments are shown in Table II. The Results in Table II show general improvement using MRAC/MPC over the nominal controller, with the

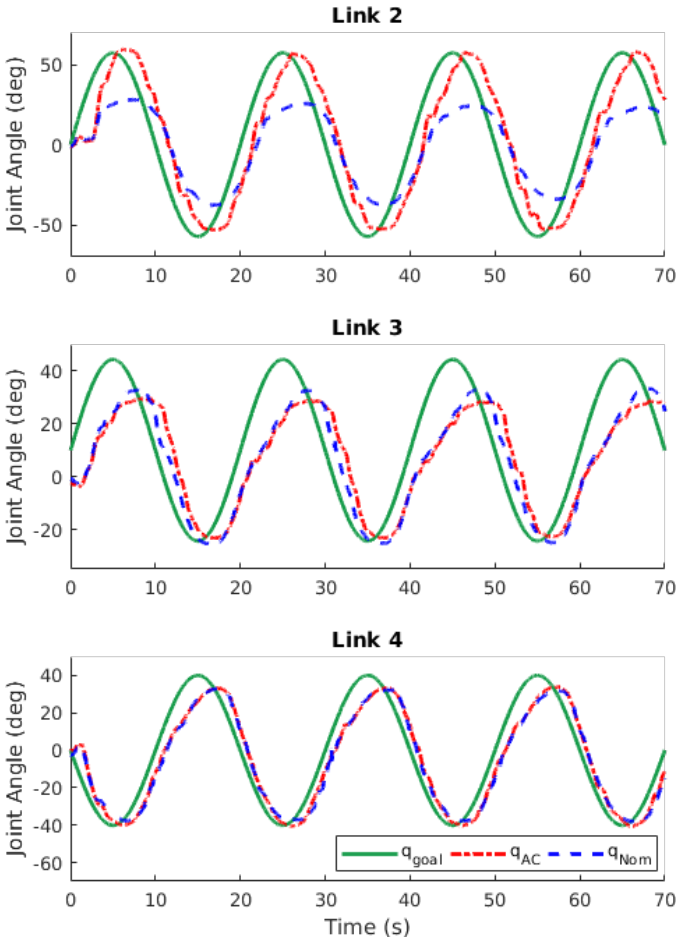

Fig. 9: Adaptive vs Nominal control for three links of King Louie's arm. A 3 lb mass was attached to the end effector for the entire run.

exception of RMS Error and Rise Time for Link 3. This is primarily due to the actuation chambers becoming saturated, which appears to impact MRAC/MPC control more than nominal control.

The performance results demonstrate some strengths and weaknesses of this approach. Considerable improvements in joint angle control performance can be seen, especially in the case when significant weight (about half of the weight of the arm) is grasped in the end effector. We can also see from these results that there is room for improvement. Adaptive parameter evolution is not shown here, but we observed that the adaptive parameters require several minutes to settle at ideal values. However, it is desirable to have adaptive control formulated such that parameters are adapted much more quickly. Additionally, it would be beneficial to decrease rise time generally for all experiments with step inputs. However, it is not clear at this point whether this is more of a current hardware limitation or a deficiency in the control schemes or adaptations used.

V. Conclusion

Future work includes developing a method for faster parameter adaption. This might include reformulating the adaptive laws discussed in Section IV, or perhaps modifying a different adaptive strategy altogether, such as the methods discussed in Section I-A. The lack of improvement when MRAC/MPC was applied to the Grub provides evidence that utilizing another adaptation method could be beneficial. For the adaptive scheme presented in this paper, control performance improvement could be attained from a more rigorous method of identifying which dynamic parameters most need adaptation. Additionally, determining if improvements can be made to following a sine wave trajectory would also be beneficial.

However, we have shown that by adapting dynamic parameters for soft robots online, we can significantly improve performance for model-based control. We demonstrated that knowing parameter values more precisely does not necessary lead to better control performance for soft robots with a given dynamic model parameterization. Instead, for these inflatable systems, adapting parameters based on control performance can be more effective than adapting parameters based on model accuracy. We also developed a form of MRAC/MPC that builds upon Coupling-Torque based MPC, and demonstrated its usefulness on three links of an inflatable soft robot with no load and while carrying an unknown load.

References

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

[2] J. A. Jones, "Inflatable Robotics for Planetary Applications," *Jet Propulsion*, no. Figure 1, pp. 1–6, 2001.

[3] 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," *Smart Actuation and Sensing SystemsRecent Advances and Future Challenges*, no. February, pp. 567–608, 2014.

[4] A. Bicchi and G. Tonietti, "Fast and "soft-arm" tactics," *IEEE Robotics and Automation Magazine*, vol. 11, no. 2, pp. 22–33, 2004.

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

[6] S. Sanan, M. H. Ornstein, and C. G. Atkeson, "Physical human interaction for an inflatable manipulator," in *Proceedings of the Annual International Conference of the IEEE Engineering in Medicine and Biology Society, EMBS*, 2011, pp. 7401–7404.

[7] M. T. Gillespie, C. M. Best, and M. D. Killpack, "Simultaneous position and stiffness control for an inflatable soft robot," *Proceedings - IEEE International Conference on Robotics and Automation*, vol. 2016-June, pp. 1095–1101, 2016.

[8] J. S. Terry, L. Rupert, and M. D. Killpack, "Comparison of Linearized Dynamic Robot Manipulator Models for Model Predictive Control," in *IEEE Humanoids Conference*, 2017.

[9] A. D. Marchese, K. Komorowski, C. D. Onal, and D. Rus, "Design and control of a soft and continuously deformable 2D robotic manipulation system," *Proceedings - IEEE International Conference on Robotics and Automation*, pp. 2189–2196, 2014.

[10] A. D. Marchese, R. K. Katzschmann, and D. Rus, "Whole arm planning for a soft and highly compliant 2D robotic manipulator," *IEEE International Conference on Intelligent Robots and Systems*, no. Iros, pp. 554–560, 2014.

[11] A. D. Marchese, R. Tedrake, and D. Rus, "Dynamics and trajectory optimization for a soft spatial fluidic elastomer manipulator," *International Journal of Robotics Research*, vol. 35, no. 8, pp. 1000–1019, 2016.

[12] S. Sanan, P. S. Lynn, and S. T. Griffith, "Pneumatic Torsional Actuators for Inflatable Robots," *Journal of Mechanisms and Robotics*, vol. 6, no. 3, p. 031003, 2014.

[13] G. Chowdhary, M. Mühlegg, J. How, and F. Holzapfel, "Concurrent Learning Adaptive Model Predictive Control," in *Advances in Aerospace Guidance, Navigation and Control*, Q. Chu, B. Mulder, D. Choukroun, E.-J. Kampen, C. Visser, and G. Looye, Eds. Springer Berlin Heidelberg, 2013, ch. 3, pp. 29–47.

[14] H. Fukushima, T.-H. Kim, and T. Sugie, "Adaptive model predictive control for a class of constrained linear systems based on the comparison model," *Automatica*, vol. 43, no. 2, pp. 301–308, 2007.

[15] V. Adetola and M. Guay, "Robust adaptive MPC for systems with exogeneous disturbances," *IFAC Proceedings Volumes (IFAC-PapersOnline)*, vol. 7, no. PART 1, pp. 249–254, 2009.

[16] Y. Chen, K. L. Moore, J. Yu, and J.-X. Xu, "Iterative learning control and repetitive control in hard disk drive industry$–$A tutorial," *International Journal of Adaptive Control and Signal Processing*, vol. 22, no. 4, pp. 325–343, 2008.

[17] K. D. Nguyen and H. Dankowicz, "Adaptive control of underactuated robots with unmodeled dynamics," *Robotics and Autonomous Systems*, vol. 64, pp. 84–99, 2015.

[18] J.-J. E. Slotine and W. Li, "On the Adaptive Control of Robot Manipulators," *ASME Winter Meeting*, vol. 6, no. 1987, pp. 43–50, 1986.

[19] J. Taghia, A. Wilkening, and O. Ivlev, "Adaptive position control of fluidic soft-robots working with unknown loads," in *2013 IEEE/ASME International Conference on Advanced Intelligent Mechatronics:*, 2013, pp. 1121–1126.

[20] G. Tonietti, "Adaptive Simultaneous Position and Stiffness Control for a Soft Robot Arm," in *Intelligent Robots and Systems*, no. October, 2002, pp. 1992–1997.

[21] S. G. Khan, G. Herrmann, T. Pipe, C. Melhuish, and A. Spiers, "Safe adaptive compliance control of a humanoid robotic arm with anti-windup compensation and posture control," *International Journal of Social Robotics*, vol. 2, no. 3, pp. 305–319, 2010.

[22] G. A. . Medrano-Cerda, C. J. . Bowler, and D. G. . Caldwell, "Adaptive Position Control of Antagonistic Pneumatic Muscle Actuators," *Intelligent Robots and Systems*, vol. 1, pp. 378–383, 1995.

[23] S. J. Qin and T. A. Badgwell, "A survey of industrial model predictive control technology," *Control Engineering Practice*, vol. 11, no. 7, pp. 733–764, 2003.

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

[25] T. Erez, Y. Tassa, and E. Todorov, "Infinite-Horizon Model Predictive Control for Periodic Tasks with Contacts," *Robotics: Science and Systems VII*, 2011.

[26] L. Rupert, P. Hyatt, and M. D. Killpack, "Comparing Model Predictive Control and input shaping for improved response of low-impedance robots," *IEEE-RAS International Conference on Humanoid Robots*, vol. 2015-Decem, pp. 256–263, 2015.

[27] D. Shim, H. Kim, and S. Sastry, "Decentralized nonlinear model predictive control of multiple flying robots," *IEEE International Conference on Decision and Control*, vol. 4, no. December, pp. 3621–3626, 2003.

[28] C. Leung, S. Huang, N. Kwok, and G. Dissanayake, "Planning under uncertainty using model predictive control for information gathering," *Robotics and Autonomous Systems*, vol. 54, no. 11, pp. 898–910, 2006.

[29] A. S. K. Annamalai, R. Sutton, C. Yang, P. Culverhouse, and S. Sharma, "Robust Adaptive Control of an Uninhabited Surface Vehicle," *Journal of Intelligent & Robotic Systems*, vol. 78, no. 2, pp. 319–338, 2015.

[30] Y. Wang and S. Boyd, "Fast model predictive control using online optimization," *IEEE Transactions on Control Systems Technology*, vol. 18, no. 2, pp. 267–278, 2010.

[31] A. Domahidi and A. Zgraggen, "Efficient Interior Point Methods for Multistage Problems Arising in Receding Horizon Control," in *IEEE Conference on Decision and Control*, 2012.

[32] E. Lavretsky and K. A. Wise, *Robust and Adaptive Control With Aerospace Applications*. London: Springer, 2013.