

Control and Evaluation of Series Elastic Actuators with Nonlinear Rubber Springs

Jessica Austin¹, Alexander Schepelmann¹, and Hartmut Geyer¹

Abstract—Series elastic actuators primarily use linear springs in their drivetrains, which introduces a design tradeoff: soft springs provide higher torque resolution at the cost of system bandwidth, whereas stiff springs provide a fast response but lower torque resolution. Nonlinear springs (NLSs) potentially incorporate the benefits of both soft and stiff springs, but such springs are often large. An NLS design was recently proposed that combines a variable radius cam with a rubber elastic element, enabling a compact spring design. However, the rubber introduces hysteresis, which can lead to poor torque tracking if not accounted for in the controller. To overcome this limitation, we here propose a state observer that captures hysteretic effects exhibited by the rubber to provide an accurate estimate of actuator torque. We perform torque-control experiments with this observer on an actuator testbed and compare the performance of the NLS to both soft and stiff linear metal springs. Experiments show that the NLS exhibits improved output impedance compared to both linear springs, and comparable bandwidth to the stiff linear spring up to 1.5 Hz. However, the hysteresis in the urethane rubber introduces instability in higher-frequency conditions, suggesting that future NLS designs can be improved by use of a different rubber as the elastic element.

I. INTRODUCTION

Series elastic actuators (SEAs) provide many benefits that make them attractive for legged robotics and powered prostheses, including greater shock tolerance, low output impedance, passive energy storage, and more accurate force control [1] [2] [3]. Traditionally, SEAs use linear springs—typically metal—as their elastic element, as these are cheap, widely available, and follow Hooke’s Law for force estimates. However, the use of linear springs introduces a design tradeoff: in a system with set encoder resolution, soft springs provide higher torque resolution at the cost of system bandwidth, whereas stiff springs provide a fast response but lower torque resolution [4].

Variable stiffness actuators attempt to overcome this tradeoff by achieving a range of stiffnesses via actively tuning a passive mechanical element with a secondary motor [5] [6] [7]. However, these systems are often bulky or complicated, hampering their application in small, lightweight robots, or for retrofit in existing robots. Passive nonlinear springs (NLSs) reduce complexity by omitting active tuning of the mechanical element, and encode a single nonlinear torque profile [8] [9]. To further reduce weight and volume, rubber is an attractive alternative to metal for these springs’

elastic element because it is compact, can tolerate large deflections, and can be easily molded to a custom shape and size. For these reasons, rubber has been incorporated into actuator designs where weight and volume is a concern [10]. Compared to metal springs, viscoelastic materials like rubber have the disadvantage of exhibiting hysteresis due to viscous effects, though others have overcome this challenge by using state observers to account for hysteresis [11].

To improve bandwidth and torque resolution in an SEA while maintaining a lightweight, compact design, we previously developed an NLS design that combines a variable radius cam with rubber springs (Fig. 2) [12]. This design allows for an arbitrary, user-defined torque-deflection profile and strives to offer the improved torque tracking of soft springs at low amplitudes and the fast response of stiff springs at high amplitudes. However, the same rubber that allows for a compact design also exhibits a nonlinear stress-strain profile and hysteretic effects (Fig. 1). If unaccounted for, these behaviors degrade the overall stability and bandwidth of closed-loop torque control, and because they are nonlinear and time-dependent, simple approaches such as Hooke’s Law or a lookup table are infeasible. Therefore, in addition to choosing a rubber that minimizes undesirable behaviors, the designer must also develop a more sophisticated model to provide an estimate of rubber state in the controller.

In this work, we focus on development of a rubber model and corresponding state observer for a single type of urethane rubber; this observer captures nonlinear and hysteretic effects exhibited by the rubber in order to provide an accurate estimate of actuator torque. We perform hardware experiments on an actuator testbed to evaluate both the state observer and the performance of the overall system. Our rubber model has an average percent relative error of 13.6% in experiment, which is a significant improvement over the error in a Hooke’s Law fit of 25.2%. The resulting observer is stable and gives estimates as accurate as its internal rubber model. Placing this observer in closed-loop with a torque controller, we compare the performance of our NLS to both soft and stiff linear springs. Experiments with the NLS show improved output impedance compared to both linear springs, and comparable bandwidth to the stiff linear spring up to 1.5 Hz. However, hysteresis in the rubber introduces instability in higher-frequency conditions; therefore, careful selection of an alternative rubber is recommended for future designs.

II. NONLINEAR SPRING SEA DESIGN

The nonlinear spring design described in [12] is a two-part assembly, consisting of an elastic element and a rotary cam

*This work is supported by the Eunice Kennedy Shriver National Institute of Child Health & Human Development under award no. 1R01HD075492.

¹J. Austin, A. Schepelmann, and H. Geyer are with the Robotics Institute, Carnegie Mellon University, Pittsburgh, PA, 15213, USA. {jaustin, aschepelmann, hgeyer}@cmu.edu

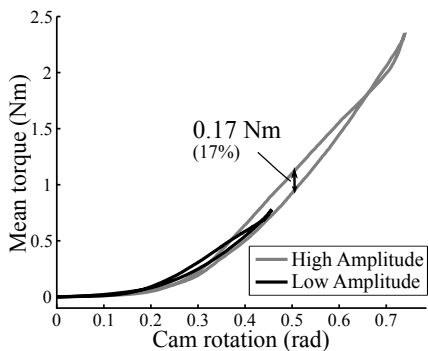


Fig. 1: Average torque profile for prototype NLS SEA. Averages are based on experimental data using open-loop position control, with velocities ranging from 0.1–9 Hz, and for low-amplitude ($\Delta\theta = 25^\circ$; $n_{cycles} = 354$) and high-amplitude ($\Delta\theta = 45^\circ$; $n_{cycles} = 222$) scenarios. A nonlinear torque profile results from both the nonlinear cam design and the rubber behavior under strain. Hysteresis on downstroke comes from using rubber as the spring.

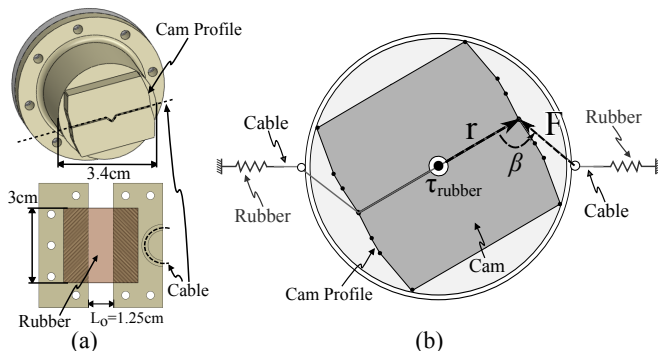


Fig. 2: NLS SEA cam prototype components (a) and schematic (b). A cable runs through the center of the cam and attaches to rubber springs at either end. As the cam rotates, the rubber stretches and the cable engages the cam at different points on the profile, effectively changing the lever arm r . The SEA torque, τ_{rubber} , is the cross product of the lever arm r and the rubber force vector \mathbf{F} .

whose profile is optimized to stretch the elastic element over a variable radius (Fig. 2). As the cam rotates, subsequent points on the cam profile are engaged, which changes the instantaneous lever arm and thus the torque generated by the spring. Through choice of the cam profile and choice of the rubber, the user can design an NLS with an arbitrary torque-deflection profile, within the constraints of manufacturing tolerances. Motivation for the NLS SEA comes from bandwidth and torque resolution limitations in existing SEAs in a robotic neuromuscular leg testbed [13] that was developed to investigate human neuromuscular controllers.

To evaluate the NLS SEA design, a prototype actuator was developed for incorporation into the robotic neuromuscular leg testbed [12]. Elastic elements are urethane rubber springs (PMC-770, Smooth-On Inc.) 3 cm wide, 3.4 mm thick and 1.25 cm long, that are held within a clamp and attached to the cam via a cable. The cam and rubber together encode an exponential torque profile with a maximum torque of 5 Nm at 53° maximum rotation. The system is designed for operating frequencies up to 9.8 Hz, which is the upper limit of human

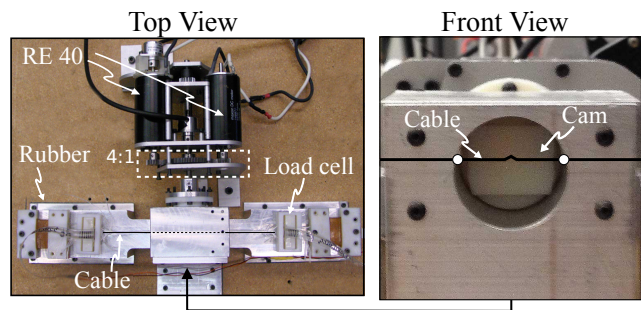


Fig. 3: Benchtop setup for rubber characterization and observer testing. To simplify experiments, the load side of the rubber is fixed. Load cells in-line with the rubber give rubber force measurements for testing, but will not be present in the final, compact design.

muscle bandwidth [14], dynamically scaled to match the robotic leg size and mass.

Experiments are conducted in a testbed (Fig. 3) that incorporates piezoresistive pressure sensors (FlexiForce A201: Tekscan) to measure the tension in the rubber. These load cells assist in observer development and will not be present in the final, compact design. Previous work developed a high-fidelity simulation for actuators in the robotic leg; this simulation runs in MATLAB Simulink with SimMechanics and includes friction, noise, and encoder discretization [15]. The simulation is updated here for use with NLS SEAs, and is used for controller development and to simulate scenarios not possible in the hardware testbed.

Hardware evaluations with the NLS SEA prototype showed that the actuator matched the desired torque profile on upstroke at low speeds of 0.1 Hz. However, due to velocity- and time-dependent effects in the rubber, there was significant deviation from the desired profile during downstroke or at higher speeds.

III. APPROACH

A. Rubber Model Development

In order to account for nonlinearities and hysteresis in the rubber, the NLS SEA state-space observer design must incorporate a model of the rubber that accurately predicts these effects. Our goal is to model the rubber force; with this we can use known cam geometry to directly calculate spring torque. Our primary requirement for this model is the ability to represent behaviors seen in experiments, such as creep, recovery, and stress relaxation (Fig. 4). Creep is increasing deflection under a constant force, recovery is a non-instantaneous return to the rest length after the force is removed, and stress relaxation is decreasing force under a constant deflection [16]. Secondly, since we wish to use this model for a state-space observer to estimate force, the model must be linear in force and deflection. Finally, since we use this model on an actual system with noisy, discretized outputs, we exclude models with high-order derivatives on these states.

A set of rubber models exists that satisfy these criteria: linear viscoelastic models [17]. These models represent

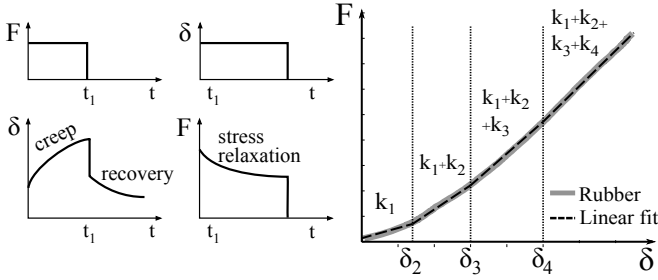


Fig. 4: Rubber behaviors that we seek to model. Left: Creep, which is deflection under a constant force, and stress relaxation, which is decreasing force under a constant deflection. Right: Strain-dependent stress, which can be approximated by linear stiffnesses.

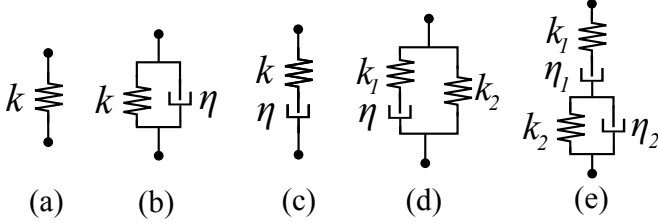


Fig. 5: Candidate viscoelastic models. Springs and dashpots represent elastic and viscous elements, respectively. (a) Hooke's Law. (b) Kelvin-Voigt model. (c) Maxwell model. (d) Standard Linear Solid model. (e) Burger's model.

rubber, a viscoelastic material, as a mechanical system composed of springs as the elastic elements and dashpots as the viscous elements (Fig. 5). These elements may be placed in series or parallel to encode various behaviors. In general, a model with a greater number of elements offers increased modeling accuracy, at the expense of mathematical complexity and higher order derivatives. For example, the Kelvin-Voigt model encodes creep but not stress relaxation, and the Maxwell model encodes stress relaxation but not creep; combining these models gives the Standard Linear Solid model and Burgers model, which encode both creep and stress relaxation [17]. However, the Burgers model has higher-order derivatives on force and deflection, resulting in amplified noise on the hardware system. Therefore we choose the Standard Linear Solid model as the simplest model that encodes our desired behaviors.

The constitutive equation for the Standard Linear Solid model is given by

$$F = k_1 \frac{A_0}{L_0} \delta + \eta \frac{k_1 + k_2}{k_2} \frac{A_0}{L_0} \dot{\delta} - \frac{\eta}{k_2} \ddot{F} \quad (1)$$

where k_1 and k_2 are stiffnesses, η is the viscosity, F is the rubber force, and δ is the change in length of the rubber. Note that rubber models are typically given in terms of stress, σ , and strain, ε ; in this paper, we give all equations in terms of F and δ . The conversion is made using $\sigma = \frac{F}{A_0}$ and $\varepsilon = \frac{\delta}{L_0}$, where A_0 is the cross-sectional area of the rubber and L_0 is the rest length. Each of the three terms of equation (1) contributes to the desired rubber behavior: the first term is essentially Hooke's Law and contributes to elasticity, the second term contributes to creep, and the third

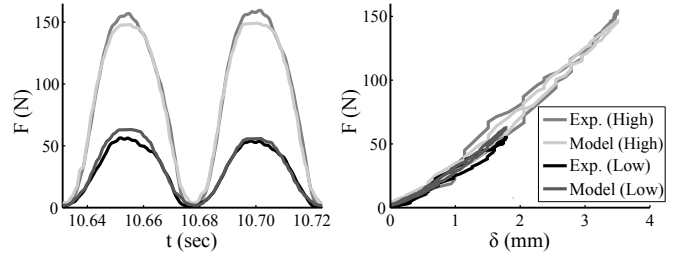


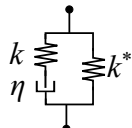
Fig. 6: Sample fit from Std. Lin. SDS model characterization, which incorporates creep, stress relaxation, and strain-dependent stiffnesses. Experimental data are from a single cycle at 4 Hz, for both low and high amplitudes. Left: Force vs time. Right: Force vs change in length of the rubber. The jagged lines are a result of finite encoder resolution.

term contributes to stress relaxation [17].

Experimental data also reveal that our rubber exhibits strain-dependent stiffness (Fig. 4), which the Standard Linear Solid model, with its constant stiffnesses k_1 and k_2 , is unable to model. However, we can approximate the nonlinear force-deflection curve with piecewise linear stiffnesses; the force equation for such a spring can be written as

$$F^* = \frac{A_0}{L_0} \sum_i^n k_i^* H_{\delta-\delta_i} (\delta - \delta_i) \quad (2)$$

where $H_{\delta-\delta_i}$ is the Heaviside step function centered at δ_i . The stiffness of k^* changes based on the amount of deflection in the rubber; it behaves as n springs in parallel, each of which ‘‘engages’’ once the rubber reaches a certain deflection. Replacing spring k_2 in the original model with this nonlinear spring and deriving the equations of motion gives



$$F = \frac{A_0}{L_0} \sum_i^n k_i^* H_{\delta-\delta_i} (\delta - \delta_i) + \frac{\eta}{k} \frac{A_0}{L_0} \left(k + \sum_i^n k_i^* \right) \dot{\delta} - \frac{\eta}{k} \ddot{F} \quad (3)$$

which we call the Standard Linear Solid model with Strain-Dependent Stiffness (Std. Lin. SDS). This model allows us to model strain-dependent stresses while keeping the model linear in δ for any particular δ , as will be shown in the following section.

To fit model parameters, we conduct characterization experiments, split into training and testing datasets, and run stochastic optimization [18] to minimize average percent relative error between the model and the actual data. In these experiments, we drive the SEA cam in sinusoids at amplitudes of 15° , 25° , and 45° , and at frequencies ranging from 0.1-9 Hz. Cam position, motor current, and force are measured. The value of δ is calculated based on the cam rotation θ along with known cam geometry, and F is measured directly using the tension load cells. The derivatives $\dot{\delta}$ and \ddot{F} are calculated using a bi-directional low pass filter.

The parameters for the Std. Lin SDS model are: $A_0 = 1 \text{ cm}^2$, $L_0 = 1.25 \text{ cm}$, $\eta = 411 \text{ Pa}\cdot\text{s}$, $k = 0.58 \text{ MPa}$, $\delta_1 = 0 \text{ mm}$,

$k_1^* = 0.92$ MPa, $\delta_2 = 0.20$ mm, $k_2^* = 0.58$ MPa, $\delta_3 = 1.06$ mm, $k_3^* = 0.74$ MPa, $\delta_4 = 2.65$ mm, $k_4^* = 0.77$ MPa, $\delta_5 = 9.04$ mm, $k_5^* = 0.47$ MPa, $\delta_6 = 9.05$ mm, and $k_6^* = 1.06$ MPa. A representative fit against one cycle of characterization data is given in figure 6. Against the full characterization data set, the model achieves an average percent relative error of 13.6%, which is a significant improvement over the error in a Hooke's Law fit of 25.2%. This fit corresponds to an RMSE of 0.10 Nm at the spring. To put this in context, we use the same SEA setup fitted with a linear spring of comparable stiffness to the NLS at small rotations (that is, higher torque resolution), and use Hooke's Law to estimate spring stiffness; this gives a fit with RMSE of 0.07 Nm, which comes from encoder discretization and noise in the load cells. Therefore, the developed rubber model approaches the limit of what can be achieved with imperfect sensing.

B. State-Space Equations

Our rubber model (3) relates force, F , and change in length of the rubber, δ . In order to improve the estimate of F , we can also take advantage of the motor dynamics, which relate the known torque at the motor, τ_{motor} , to the unknown rubber torque, τ_{rubber} ,

$$J\ddot{\theta} = \tau_{motor} - \tau_{rubber} = nk_T I - rF \sin \beta, \quad (4)$$

where I is the motor current, k_T is the torque constant, and n is the gear reduction between the motor and the spring. Putting this together with the rubber model equation (3), and $\delta = \delta(\theta)$, gives state-space equations for the cam:

$$\begin{aligned} \dot{\mathbf{x}} &= \mathbf{A}(\theta)\mathbf{x} + \mathbf{B}\mathbf{u} + \mathbf{E}(\theta) \\ \mathbf{y} &= \mathbf{C}\mathbf{x} \end{aligned}$$

where

$$\mathbf{x} = \begin{bmatrix} F \\ \delta \\ \theta \\ \dot{\theta} \end{bmatrix}, \quad \mathbf{A}(\theta) = \begin{bmatrix} -\frac{k}{\eta} & a_{12} & 0 & 0 \\ 0 & 0 & 0 & \frac{d\delta}{d\theta}(\theta) \\ 0 & 0 & 0 & 1 \\ \frac{-r \sin(\beta(\theta))}{J} & 0 & 0 & 0 \end{bmatrix}$$

$$\mathbf{B} = \begin{bmatrix} \frac{A_0}{L_0} \left(k + \sum_{i=1}^6 k_i^* \right) & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & \frac{k_T n}{J} \end{bmatrix}, \quad \mathbf{u} = \begin{bmatrix} \dot{\delta} \\ I \end{bmatrix}$$

$$\mathbf{E}(\theta) = \begin{bmatrix} \frac{k}{\eta} \frac{A_0}{L_0} \sum_{i=1}^6 k_i^* H_{\delta(\theta) - \delta_i} \delta_i \\ 0 \\ 0 \\ 0 \end{bmatrix}, \quad \mathbf{C} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix}$$

and $a_{12} = \frac{k}{\eta} \frac{A_0}{L_0} \sum_{i=1}^6 k_i^* H_{\delta(\theta) - \delta_i}$. Note that $\mathbf{E}(\theta)$ is simply a constant offset matrix required to linearly interpolate between step changes in the force for the spring k^* .

Since β and δ change with the cam position, θ , the \mathbf{A} and \mathbf{E} matrices are not time-invariant. However, they can be calculated for all the possible values of θ , so in the observer preprocessing code a lookup table is created for $\mathbf{A}(\theta)$ and $\mathbf{E}(\theta)$ for $\theta \in (-\frac{\pi}{2}, \frac{\pi}{2})$.

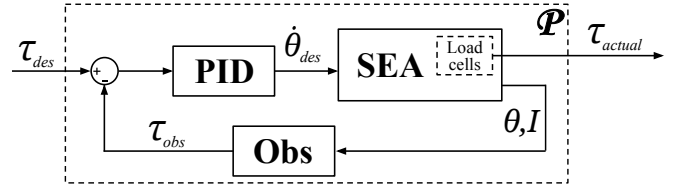


Fig. 7: Plant used for closed-loop system identification. The observer provides an estimate of spring torque, which is subtracted from a desired torque and fed into a PID controller. The controller converts this error to a velocity command that is sent to the motor controller, and sensors provide estimates of θ and motor current I .

C. Observer Design

In our system, we implement a Luenberger Observer [19],

$$\dot{\hat{\mathbf{x}}} = \mathbf{A}(\theta)\hat{\mathbf{x}} + \mathbf{B}\mathbf{u} + \mathbf{L}(\mathbf{y} - \mathbf{C}\hat{\mathbf{x}}) + \mathbf{E}(\theta) \quad (5)$$

where we select values for the observer gain matrix \mathbf{L} using pole placement; the poles are chosen via optimization to minimize the error between state estimates and actual data gathered during characterization.

To check for observability, we calculate the observability matrix $\mathbf{N} = [\mathbf{C} \ \mathbf{C}\mathbf{A} \ \mathbf{C}\mathbf{A}^2 \ \mathbf{C}\mathbf{A}^3]^T$ for each value of θ . In most cases, \mathbf{N} is full rank, but when $\beta = \pi$ it is rank 3 and thus the system is unobservable. Pole placement in unobservable conditions is impossible, and this occurs when the cam is passing through $\theta = 0$ and so is unavoidable during typical use. However, using Kalman decomposition [19] we can decompose the system into observable and unobservable sub-systems, $\mathbf{A}_{obs}, \mathbf{C}_{obs}$ and $\mathbf{A}_{unobs}, \mathbf{C}_{unobs}$. The observable sub-system is composed of the states F , δ , and $\dot{\theta}$. The unobservable sub-system is the state θ and has eigenvalue less than zero, so it is asymptotically stable. Since the unobservable sub-system is stable, we can determine \mathbf{L} using pole-placement with the following algorithm: when $\theta \neq 0$, choose \mathbf{L} to satisfy desired closed-loop poles for $\mathbf{A} - \mathbf{L}\mathbf{C}$, and when $\theta = 0$, perform pole placement for the observable sub-system $\mathbf{A}_{obs} - \mathbf{L}\mathbf{C}_{obs}$ while leaving the unobservable sub-system poles at zero.

Since our state-space matrices and observer gain matrix change based on the cam rotation θ , our design implements a form of gain scheduling. With the exception of some special cases, there is no method to prove global stability for a system with gain scheduling—instead the researcher must verify stability through experiment [20]. In our case, the developed observer is indeed stable, both in simulation and on the hardware testbed. To characterize the observer error, we run simulations and compare against characterization data; the observer achieves 13.7% average percent relative error and 0.10 Nm RMSE—the same performance as the rubber model.

IV. PERFORMANCE ASSESSMENT

The original goal of the NLS SEA was to provide the torque resolution of a soft linear spring at low amplitudes and the bandwidth of a stiff linear spring at high amplitudes, all in a lightweight and compact design. To evaluate whether the

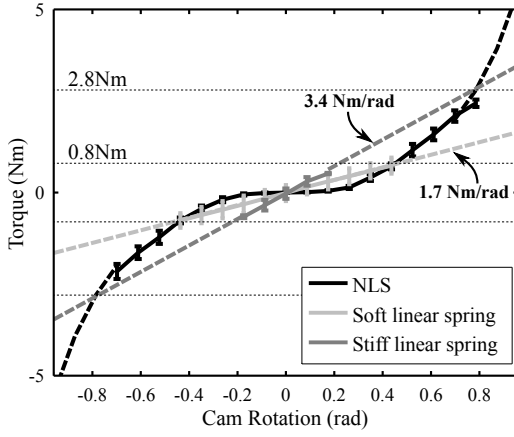


Fig. 8: Average experimental torque profiles for each spring, with standard deviations. Dashed lines indicate extrapolated data. For comparison purposes, linear spring stiffnesses are chosen for the low-amplitude ($\tau_{des} = 0.8$ Nm) and high-amplitude ($\tau_{des} = 2.8$ Nm) scenarios via linear fit: $k_{s,soft} = 1.7$ Nm/rad, $k_{s,stiff} = 3.4$ Nm/rad.

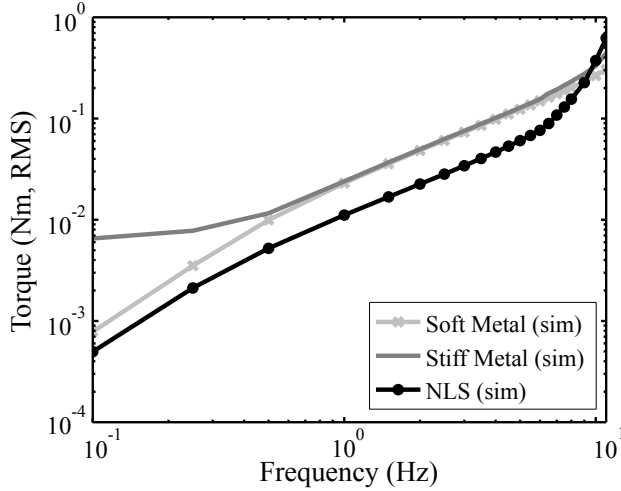


Fig. 9: Simulated output impedance (zero-torque tracking) response. The load is moved in sinusoids of varying frequency, while the controller attempts to maintain zero torque.

system with state observer that estimates rubber hysteresis achieves these goals, we run a series of experiments. To test torque resolution, we run output impedance (zero-torque tracking) experiments and compare performance versus the linear springs. A spring with higher torque resolution will have lower output impedance because the controller has finer-grained feedback with which to reduce error. We also generate and plot frequency response at low amplitude to verify that the system can track torques as well as the soft spring. To test bandwidth, we run high-amplitude frequency response experiments, and compare to the phase margin of the stiff linear spring.

These experiments are conducted in the benchtop setup described in section II; the system plant is given in figure 7. The actuator remains the same for all experiments, and the spring is either a soft linear spring, a stiff linear spring, or

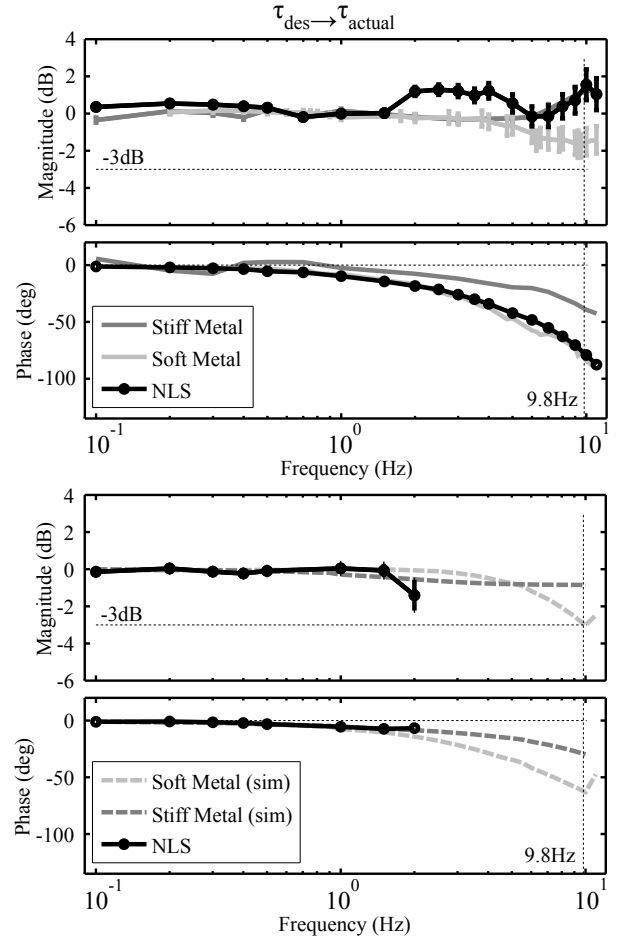


Fig. 10: Frequency response for NLS SEA. Top: Low-amplitude hardware experiments; $\tau_{des} = 0.8$ Nm. Bottom: High-amplitude hardware NLS experiments, with simulated linear spring performance; $\tau_{des} = 2.8$ Nm. RMSE is given for hardware torque tracking at each frequency.

the NLS. PID gains are tuned for each spring; these gains are first optimized in simulation and then fine-tuned in hardware. As with the characterization experiments, our testbed used load cells to acquire ground-truth force measurements, but these values are not fed back into the system; instead, the proposed state observer provides force estimates for the NLS, and Hooke's Law is used for the linear springs.

In choosing the stiffness of our linear springs, we aim to have as wide a range as possible, but are limited by available hardware that is compatible with our actuator: the soft linear spring has measured stiffnesses of 1.7 Nm/rad (A5Z26M0606: SDP/SI) and the stiff spring 3.4 Nm/rad (S50TLCM13H06H06: SDP/SI). Desired torque for low and high amplitude cases is chosen based on the intersection between these stiffnesses and the NLS torque profile (Fig 8).

In the output impedance experiments, the load end of the actuator is driven in a sinusoid at frequencies ranging from 0.1–10 Hz, and the controller is given a desired torque of zero; therefore low output impedance corresponds to torque RMS near zero. Since our benchtop setup fixed the load side of the NLS, output impedance experiments were limited to

run in the simulation environment described in section II. The results of this experiment are given in figure 9: the NLS achieves lower output impedance than both linear springs up to 9 Hz. We attribute this to two reasons: at low amplitudes, the stiffness of the NLS is comparable to the soft spring, which leads to higher torque resolution; in addition, the rubber has better shock tolerance to load movement due to its viscous properties.

In addition to output impedance experiments, we must also verify that the NLS can track desired torque sinusoids as well as the soft linear spring. Figure 10 gives frequency response for the NLS and both linear springs for frequencies from 0.1–11 Hz at $\tau_{des} = 0.8$ Nm. The phase of the NLS and soft spring are comparable, and so is the magnitude up to 1.5 Hz; however beyond 2 Hz the NLS does not track as well as the soft spring. In this region, the observer is not degrading in performance, rather the overshoot comes from integrator windup due to a deadzone around zero torque. This deadzone comes primarily from stress relaxation in the rubber—modeled with the $-\frac{\eta}{k}\dot{F}$ term in equation (3)—which results in a resistance to sudden force changes. A rubber with higher viscosity, η , will have greater stress relaxation, and since the term is keyed to velocity, this effect will also have greater influence at high frequency. Stress relaxation is apparent even in the open-loop characterization data (Fig. 1), where a higher amplitude at the same frequency results in faster deflection of the rubber and thus a wider deadzone around zero torque. Based on the output impedance and frequency response experiments at low amplitude, the NLS SEA displays comparable torque resolution to the soft linear spring up to 1.5 Hz, but degraded torque tracking at higher frequencies due to rubber hysteresis.

To test the bandwidth of the NLS SEA, we conduct frequency response experiments at high torque amplitude, $\tau_{des} = 2.8$ Nm, and compare the NLS phase margin to that of the stiff linear spring. For this scenario, there were no linear springs available that both matched our actuator design and had the necessary maximum torque rating, so in this case the linear spring experiments were simulated. Up to 1.5 Hz, the NLS has the same phase margin as the stiff linear spring, however beyond 2 Hz the system was unstable. Again, observer performance remains stable and instability results from stress relaxation due to high rubber viscosity, and is exacerbated by the higher linear velocities.

For both the low and high amplitude frequency response experiments, using a less aggressive integrator gain did result in stable performance up to 11 Hz, however the performance compared to the linear springs also degraded. Therefore, in order to achieve our original NLS goals, a rubber with lower hysteresis and comparable stiffness must be used.

V. CONCLUSIONS AND FUTURE WORK

In this paper, we presented a state observer design to estimate the torque generated by a rubber spring in an SEA with a nonlinear spring and compact design, and we evaluated both the observer and the overall system in hardware experiments. These experiments demonstrate that this

observer is stable, and its torque estimates are a significant improvement over Hooke’s Law. Hardware and simulation experiments also demonstrate that the NLS with an observer has improved output impedance over the linear springs, and comparable torque resolution and bandwidth up to 1.5 Hz. However, the hysteretic properties of the rubber lead to instability at higher frequencies, which suggests that state observation alone cannot eliminate control issues associated with nonlinear rubber behaviors; therefore, careful selection of an alternative rubber is recommended in future designs.

VI. ACKNOWLEDGEMENTS

We thank Lawrence Hayhurst for assistance in testbed fabrication, Carmel Majidi for comments on our rubber model, and Steve Collins for his comments on this manuscript.

REFERENCES

- [1] G.A. Pratt and M.M. Williamson. Series elastic actuators. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, volume 1, pages 399–406, 1995.
- [2] J.E. Pratt and B.T. Krupp. Series elastic actuators for legged robots. In *Defense and Security*, pages 135–144. International Society for Optics and Photonics, 2004.
- [3] S.K. Au and H.M. Herr. Powered ankle-foot prosthesis: the importance of series and parallel motor elasticity. *IEEE Robotics & Automation Magazine*, 15(3):52–59, 2008.
- [4] D.W. Robinson. *Design and analysis of series elasticity in closed-loop actuator force control*. PhD thesis, Massachusetts Institute of Technology, 2000.
- [5] S.S. Groothuis *et al.* The vsaUT-II: A novel rotational variable stiffness actuator. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 3355–3360, 2012.
- [6] A. Jafari *et al.* A novel actuator with adjustable stiffness (AwAS). In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 4201–4206, 2010.
- [7] G. Tonietti *et al.* Design and control of a variable stiffness actuator for safe and fast physical human/robot interaction. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 526–531, 2005.
- [8] S.A. Migliore *et al.* Novel nonlinear elastic actuators for passively controlling robotic joint compliance. *Journal of Mechanical Design*, 129(4):406–412, 2007.
- [9] N. Schmit and M. Okada. Design and realization of a non-circular cable spool to synthesize a nonlinear rotational spring. *Advanced Robotics*, 26(3-4):234–251, 2012.
- [10] D. Rollinson *et al.* Design and modeling of a series elastic element for snake robots. In *ASME Dynamic Systems and Control Conference*, pages V001T08A002–V001T08A002, 2013.
- [11] F. Parietti *et al.* Series viscoelastic actuators can match human force perception. *IEEE/ASME Transactions on Mechatronics*, 16(5):853–860, 2011.
- [12] A. Schepelmann *et al.* Compact nonlinear springs with user defined torque-deflection profiles for series elastic actuators. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 2014.
- [13] A. Schepelmann *et al.* Development of a testbed for robotic neuromuscular controllers. In *Robotics: Science and Systems*, 2012.
- [14] J.D. Bronzino. *Biomedical engineering handbook*, volume 2. CRC press, 1999.
- [15] A. Schepelmann *et al.* Evaluation of decentralized reactive swing-leg control on a powered robotic leg. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2015.
- [16] R.S. Lakes. *Viscoelastic solids*, volume 9. CRC press, 1998.
- [17] W.N. Findley and F.A. Davis. *Creep and relaxation of nonlinear viscoelastic materials*. Courier Dover Publications, 2013.
- [18] N. Hansen. CMA-ES MATLAB implementation (version 3.61) [Computer program]. Available at <https://www.lri.fr/~hansen/cmaes.m>.
- [19] P.J. Antsaklis and A.N. Michel. *Linear systems*. Springer, 2006.
- [20] J.S. Shamma and M. Athans. Gain scheduling: potential hazards and possible remedies. *IEEE Control Systems Magazine*, 12(3):101–107, 1992.