2023 IEEE International Conference on Robotics and Automation (ICRA 2023)
May 29 - June 2, 2023. London, UK

Achieving Extensive Trajectory Variation in Impulsive Robotic Systems*

Luis Viornery[1], Chloe Goode[2], Gregory Sutton[2], and Sarah Bergbreiter[1]

***Abstract*— Robots that use impulsive mechanisms to achieve high-speed and high-powered motion are becoming more common and better understood, but control of these systems remains relatively rudimentary. Among robots that use spring actuation to generate motion, robot actuation and mechanisms are usually not controlled intentionally in order to achieve variation in the system's behavior, or they are controlled only roughly via adjustments made to the amount of energy stored in the mechanism. We describe the development, construction, and test of an impulsive catapult mechanism whose design is inspired by the grasshopper leg and for which extensive variation in the projectile trajectory is achieved by force control of the actuator that restrains the spring. As a step toward future controlled jumping robots, we give a detailed model of this system, validate this model experimentally, and explain how the actuator dynamics are critical to our ability to vary the system's trajectory using this approach. This work represents a novel approach to the control of spring actuated robots and illustrates how they can be controlled even under highly limiting actuator constraints.**

I. Introduction

Because robots are often limited by the capabilities of their actuators when high power is required, especially at small size scales, impulsive power-amplification mechanisms offer appealing advantages in size and mass. These mechanisms function by regulating the release of energy stored in elastic members, and are often inspired by biological systems where similar structures allow organisms to significantly exceed the power limits of their muscles [1], [2]. The utility of these systems in robots is limited by the difficulty with which they are controlled, which arises in part from the speed of actuation and magnitude of the forces involved.

The vast majority of small jumping robots have minimal ability to vary their trajectories [3]–[7]. In contrast, the degree of ballistic variation for a single grasshopper shown in Figure 1 demonstrates that minor changes to such systems' configurations and controls can produce large variations in observed behavior. In the grasshopper's case, this variation is largely due to changes in the positional and loading initial conditions [8]. However, as we show in this work, variation can also be accomplished via precise control of these systems without alteration to the mechanism's initial configuration.

*This material is based on work supported by the U.S. Army Research Laboratory and the U.S. Army Research Office under contract/grant no. W911NF-15-1-0358.

[1]Luis Viornery and Sarah Bergbreiter are with Department of Mechanical Engineering, Carnegie Mellon University, 5000 Forbes Ave, Pittsburgh, PA, USA, 15217 lviorner@andrew.cmu.edu, sbergb@andrew.cmu.edu

[2]Chloe Goode and Gregory Sutton are with Department of Life Sciences, University of Lincoln, Brayford Way, Brayford Pool, Lincoln, United Kingdom, LN6 7TS 16601563@students.lincoln.ac.uk, gsutton@lincoln.ac.uk

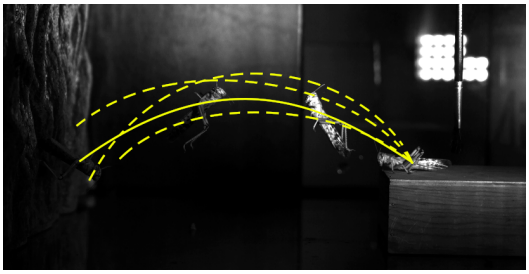

Fig. 1. Several jumping trajectories of a grasshopper starting from the same initial condition. The grasshopper that generated the trajectory represented by the solid line is shown at various points along the trajectory.

Only a few spring actuated systems have attempted to control the system output by controlling the latch – the element of the impulsive system that mediates potential to kinetic energy transfer [1]. Of the systems that have, the contact latch in the catapult system in [9] and the torque-reversal latch in the mantis shrimp robot in [10] used highly stiff actuators to move their latches so that the applied forces did not depend strongly on the actuator dynamics [11]. The same is true for power-amplification systems that are not purely driven by springs such as Salto [12]. The use of position control for these systems is only possible because the inertial and frictional forces present in these systems are substantially less than the maximum achievable actuator forces. If this were not the case, the high-bandwidth position control used by these systems would not be possible.

Conversely, some impulsive biological systems, including the grasshopper, operate in regimes where dissipative forces are of a similar magnitude to the muscle actuation forces [13]. The dynamic properties of the muscles which restrain the motion of these systems and mediate the transfer of stored potential energy to kinetic energy therefore play a major role in determining the evolution of their trajectories. These organisms also provide proof that it is possible for this class of mechanism to function well even when actuator dynamics play a significant role in the equations of motion, opening the door for the development of control strategies that will enable lighter-weight actuators to be used in impulsive applications.

Robotic jumpers previously built using grasshopper leg type or similar mechanisms have typically been designed to demonstrate or test the efficiency of the conversion between elastic potential energy and kinetic energy rather than controllability [3]–[7]. However, the ballistic trajectory of a jumper depends on the evolution of ground reaction forces over time during takeoff, which are moderated by the joint

actuators and which act on the jumper to produce particular launch angles and kinetic energies. In many cases a more optimal launch angle will be more desirable than a perfectly efficient jump; the ability to control the forces developed in the joints (and subsequent ballistic trajectory) of such a jumper by changing the timing of spring recoil therefore promises to advance jumping robot mobility, even though it necessarily implies some loss of efficiency via the dissipation of kinetic energy.

With this goal in mind, the primary contribution of this work is a new strategy that can be used for extensive trajectory variation in impulsive systems. Specifically, we describe the design, development, and test of a grasshopper-leg-inspired mechanism whose output can be varied by shaping actuator effort. In this case, we model the grasshopper leg as a catapult to isolate the impulsive mechanisms for simpler experimental validation. We show that it is possible to leverage the passive dynamics of this type of system to achieve a greater degree of control authority over its behavior than would be possible with a massless and lossless actuator and show that this holds true for a physical system as well as in simulation.

II. Methods

A. Grasshopper-Inspired Mechanism

Mechanisms based on the grasshopper leg as shown in Figure 2 are well-understood [2], [13], [14] and have been incorporated into many jumping robot designs [3], [4], [15], [16]. These designs typically incorporate a strong and slow extensor actuator coupled to the knee joint via an elastic element and a fast and weak flexor actuator which connects directly to the knee, as well as an elastic power-storage element. This mechanism enables the low-power flexor and extensor actuators to be coupled to produce a high-power, high-acceleration output via the fast release of energy stored in the elastic element.

In order to execute a jump or kick, the leg is driven to a flexed position by contraction of the flexor, with the extensor loose and elastic element unloaded. In this position, the flexor enjoys a significant mechanical advantage over the extensor. The extensor then contracts and loads the elastic element. Actuation occurs when tension in the flexor drops below the threshold at which the joint can remain in static equilibrium. As the joint starts to move, the flexor loses mechanical advantage and the dynamic behavior is dominated by the release of tension in the elastic element. In this work, we based our mechanism on the grasshopper leg, flipping it upside down in order to mimic a kick rather than a jump.

B. Mathematical Models

We developed two distinct hybrid dynamical mathematical models of our mechanism. Although the only difference between the two models is how the force from the flexor actuator, $f_f(t)$, is modeled, this distinction has major implications for the mechanism's performance.

The first model includes no losses or inertial forces in the system's flexor actuator. This model can be thought of as the model of an "ideal" system in which the transmitted and commanded force in the flexor are identical. The second model incorporates frictional losses in this actuator and also considers the actuator rotor's inertia. It also enforces a constraint on the maximum force commanded from the flexor actuator ($|f_{fc}(t)| \leq 3$ N).

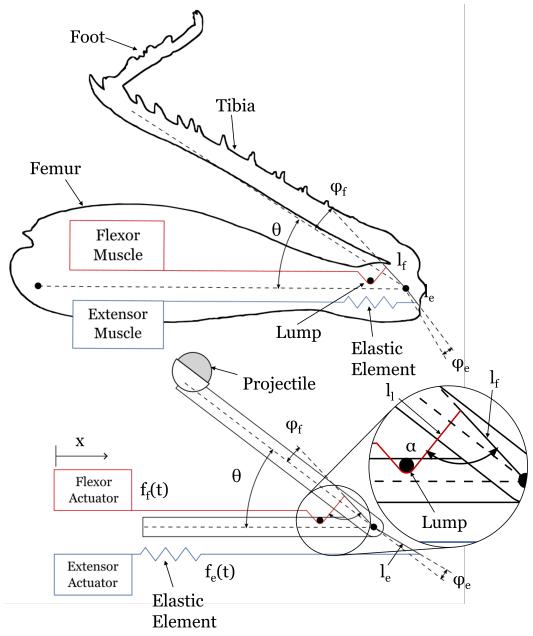

Fig. 2. Diagrams of grasshopper leg and catapult mechanisms, with flexor and extensor force paths shown in red and blue, respectively.

For both models, the element labeled "lump" is modeled as a frictionless pulley [2]; this element allows the flexor's mechanical advantage to change with the leg angle θ by changing the angle at which the force from this actuator is applied as described above.

1) Simple Model: If we assume direct force transmission through the flexor as shown in Figure 2, and I the moment of inertia of the leg (with projectile), the equation of motion for this system is given by

$$I\ddot{\theta} = \tau_t(t,\theta) \tag{1}$$

With total torque τ_t given by

$$\tau_t(t,\theta) = \tau_e(\theta) - \tau_f(t,\theta) \tag{2}$$

With the extensor torque τ_e given by

$$\tau_e(\theta) = l_e f_e(\theta)\sin(\theta) \tag{3}$$

$$f_e(\theta) = f_{pl} + k\left(l_e\left[1+\cos\left(\theta-\phi_e\right)\right] - d_{k0}\right)$$

And the flexor torque τ_f

$$\tau_f(t,\theta) = l_f f_f(t)\sin\left(\alpha(\theta)\right) \tag{4}$$

$$\alpha(\theta) = \arccos\left(\frac{d^2(\theta)+l_f^2-l_l^2}{2d(\theta)l_f}\right)$$

$$d(\theta) = \sqrt{l_f^2+l_l^2-2l_f l_l\cos\left[\beta(\theta)\right]}$$

$$\beta(\theta) = \theta + \phi_f$$

With f_{pl} the preload force of the spring, k the spring constant, d_{k0} the offset required to set zero-force angular position θ_k, and all other parameters defined in Figure 2.

In the simple model, the force commanded in the flexor actuator $f_{fc}(t)$ is transmitted directly through the tendon, so $f_f(t) = f_{fc}(t)$. Launch is assumed to occur when the angular acceleration of the arm drops to zero or below, in which case separation between the arm and projectile occur. Spring recoil only contributes to actuation until the spring has returned to its zero-force position. We define an actuator constraint at 0 N, as the actuator cannot push the leg in the direction of motion. The system is thus limited to four hybrid states. The simplest two are pre-motion, where the system is not moving, and post-launch, where projectile separates from the mechanism and the simulation immediately terminates. In addition, there are two pre-launch states that take place while actuation is occurring. These pre-launch states are the spring-driven and free-flight states, which correspond to the conditions where the spring is still accelerating the system or has reached its preload length respectively.

For a future jumper, similar hybrid states would be required to analyze the dynamics of the system in stance and flight with both spring and motor actuation. The constraints that define these states are relatively simple to formulate and incorporate into both this model and the rotordynamics model described below.

2) Rotordynamics Model: Generating flexor forces $f_f(t)$ following a given profile using a physical actuator as described above is a nontrivial problem in hardware, as the inertia and friction of the actuator strongly affect the applied force. While it is possible in principle to use feedback control to attenuate these effects, incorporating this control loop imposes additional limits on both the bandwidth and the range of achievable flexor forces of the system. For these reasons, we modeled the rotordynamics of the flexor actuator as a component of the overall system in order to assess the variation in trajectory we could achieve without idealizing away the system's rotordynamics.

We formulate the dynamics of the full system using a constrained Lagrangian method, applying the holonomic constraint

$$x = \sqrt{l_f^2 + l_l^2 - 2 l_f l_l \cos(\theta - \phi_e)} + b$$

Where x is the displacement of the flexor actuator rotor and b is an arbitrary constant. The resulting Lagrange multiplier captures the tension in the flexor, and the equation of motion of the actuator is given by

$$m\ddot{x} = f_f(t, \theta, \dot{\theta}) + f_{fc}(t) - f_\mu - c\dot{x} \tag{5}$$

With f_μ the force of friction, c the damping coefficient of the motor, and m the motor mass, although $f_f(t, \theta, \dot{\theta})$ is treated as a force of constraint and omitted in the Lagrangian formulation. This yields the system of equations

$$\begin{bmatrix} M \end{bmatrix} \begin{bmatrix} \ddot{\theta} & \ddot{x} & \lambda \end{bmatrix}^T = \begin{bmatrix} F \end{bmatrix} \tag{6}$$

With

$$\begin{bmatrix} M \end{bmatrix} = \begin{bmatrix} I & 0 & \frac{l_f l_l \sin(\theta - \phi_e)}{\sqrt{l_f^2 + l_d^2 - 2 l_f l_l \cos(\theta - \phi_e)}} \\ 0 & m & 1 \\ \frac{l_f l_l \sin(\theta - \phi_e)}{\sqrt{l_f^2 + l_d^2 - 2 l_f l_l \cos(\theta - \phi_e)}} & 1 & 0 \end{bmatrix}$$

$$\begin{bmatrix} F \end{bmatrix} = \begin{bmatrix} \tau_e(\theta) \\ f_c(t) - f_\mu - c\dot{x} \\ \frac{l_f l_l \dot{\theta}^2 \cos(\theta - \phi_e) \left[l_f^2 + l_l^2 - 2 l_f l_l \cos(\theta - \phi_e) \right]}{\left(\sqrt{l_f^2 + l_d^2 - 2 l_f l_l \cos(\theta - \phi_e)} \right)^3} \end{bmatrix}$$

Launch is once again assumed to occur when the angular acceleration of the arm drops to zero or below.

Moreover, the force actually applied through the flexor line to the arm is dependent on both the commanded force $f_{fc}(t)$ and the system state. Under some conditions, it is possible for the tension in the line between the actuator and mechanism to disappear and then later resume. For this reason, we apply actuator effort constraints and include dynamics equations for additional hybrid states where tension in the flexor line is lost in our code. The full set of hybrid states for this system are: pre-motion, spring-driven pre-launch with flexor force applied, spring-driven pre-launch without flexor force applied, pre-launch non-spring-driven with flexor force applied, pre-launch free flight, and post-launch. A similar model for a jumping robot could be constructed by considering the state of the robot rather than the projectile.

C. Commanded Flexor Force Variation

For both system models, we solved the equations of motion forward in time numerically in order to generate predictions of the system's behavior. To assess the variability of the system's trajectory, we varied the flexor effort curves $f_{fc}(t)$. Previous work has not controlled the rate of force decay [4], [5], [15], [16], whereas we chose to apply a time-varying force modeled as a polynomial curve parameterized by total decay time t_d and exponent n in order to assess variation across a range of possible applied force profiles. These curves had the form

$$f_{fc}(t) = \begin{cases} f_{fc}(0) \left[1 - \left(\frac{t}{t_d} \right)^n \right], & t < t_d \\ 0, & \text{otherwise} \end{cases} \tag{7}$$

With $f_{fc}(0)$ chosen to balance the system (for the simple model) or at the actuator constraint, $f_{fc}(0) = f_{fc_{max}}$ (for the rotordynamics model).

D. Flexor Force Generation for Trajectory Optimization

We also used a direct single shooting method to generate force profiles that would yield more varied system trajectories. In order to accomplish this, we parameterized the commanded force profile as a vector of forces applied at evenly spaced intervals of 5 ms, with the instantaneously applied force values given by interpolating linearly between these points. We parameterized the desired trajectories using the angular velocity at launch and the time of launch and used a nonlinear solver to obtain an optimal force profile.

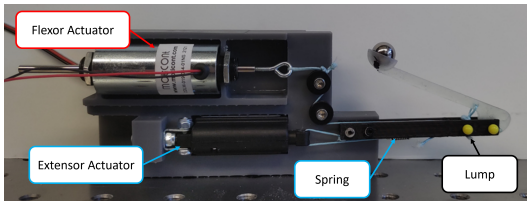

Fig. 3. Experimental setup

We chose maximum and minimum target values for these optimal control challenges based on the maximum and minimum speeds and launch times calculated for the simple and rotordynamics models in Section III-A.

E. Experimental Setup

In order to experimentally validate our computational results, we constructed the the physical catapult mechanism shown in Figure 3. We chose to implement this mechanism on a catapult platform rather than a jumper for two reasons: first, to isolate the dynamics in the leg from the dynamics of the full-scale system, and second, to use a large, heavy, high-bandwidth actuator for the flexor that we would not have to gear. These choices allowed us to explore a larger range of possible actuator inputs in our tests.

The catapult projectile is a 22 g steel sphere. The extensor actuator is a highly geared linear servomotor (PQ12-R, Actuonix Motion Devices) and the flexor actuator is a linear voice coil actuator (DDLM-019-044-01, Moticont) with maximum force constraint $|f_{fc}(t)| \leq 3$ N. We controlled the latter using PWM at a frequency of 160 kHz and with PWM values updated every 0.02 ms. We controlled both of these motors in an open-loop configuration using an Arduino Uno and a motor driver shield (DRV8835 Dual Motor Driver Shield, Pololu) and tracked the trajectory of the mechanism using a high-speed camera (FASTCAM NOVA S12, Photron) and TEMA motion tracking software (Image Systems) at 6000 fps. For each force profile tested, we captured data for 5 trials and tabulated the maximum trajectory tracking error across all five trials.

III. Results and Discussion

A. Computational Results

1) Simple and Rotordynamics Model Trajectory Variation: The simple model described in Section II-B.1 showed substantial promise in its ability to produce varied output trajectories as t_d and n in equation (7) were changed. As Figure 4 (a) and (b) show, the launch angular velocity varied by as much as 20 rad/s over the set of allowed inputs, from 130 to 110 rad/s (corresponding to a maximum projectile range of over 6 m under ideal conditions). The launch time also varied substantially, up to 0.15 s. However, this variation was only obtained with values of $f_{fc}(0)$ significantly greater than those allowed by the actuator effort constraints imposed in our rotordynamics model; it is not possible to achieve this degree of variation in system output if this constraint is

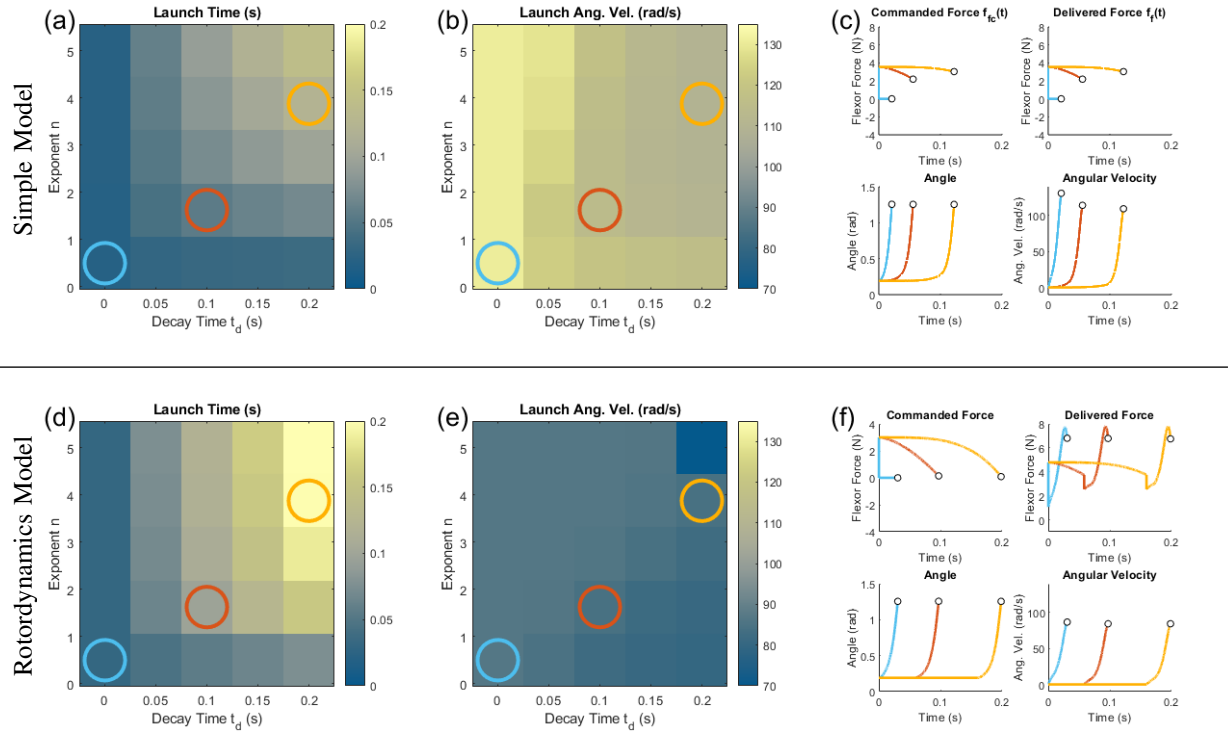

Fig. 4. Simple and rotordynamics model results. (a) and (b) show variation in launch time and launch angular speed over a range of commanded flexor force parameters t_d and n for the simple model, while (d) and (e) show this variation for the the rotordynamics model. (c) shows applied forces, delivered forces, and angle and angular velocity trajectories for three representative commanded force profiles using the simple model, and (f) shows the same plots for the same force profiles using the rotordynamics model. All plots terminate when launch occurs.

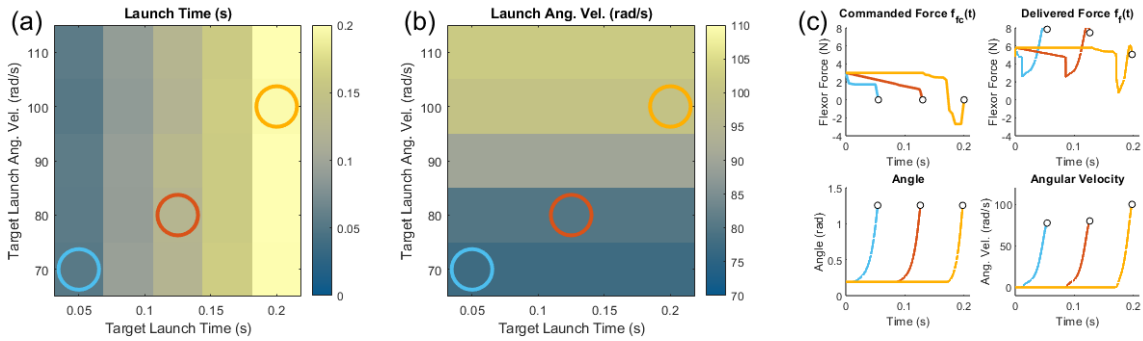

Fig. 5. Computational trajectory tracking results. (a) and (b) show variation in launch time and launch angular speed over a range of objective values. (c) shows applied forces, delivered forces, and angle and angular velocity trajectories for three representative commanded force profiles applied to the rotordynamics model.

applied to the simple model, as the constrained simple model system yields tightly clustered launch times and less variation in angular velocity. Therefore, once actuator constraints are introduced, it becomes impossible to both obey those constraints and to require the system to behave as though it contains an idealized force source through normal feedback control.

Although the model described in Section II-B.2 implemented a model of the actuator rotordynamics, it showed substantially less variation overall in output trajectories using inputs based on equation 7. Although it displayed slightly greater variation in launch time relative to the simple model, the launch angular velocity varied by less than 10 rad/s in the rotordynamics model. As Figure 4 shows, this behavior can be ascribed to homogenization of the delivered flexor force $f_f(t)$ when actuator rotordynamics are considered. In these plots, it is clear that the forces contributed by those passive dynamics dominated the contribution of the commanded forces over a large region of actuation.

Because the forces introduced in the rotordynamics model all performed negative work on the system, the launch angular velocities obtained for the rotordynamics model were also uniformly lower than those obtained for the simple model, with all results in the range of 80-90 rad/s, corresponding to a maximum projectile range of less than 3 m, and peak delivered flexor forces were substantially higher than for the simple model. This suggested that if the behavior of the simple model could be emulated, it might be possible to achieve much greater variation in the output trajectory through optimal control than is possible using either of the simple commanded force profiles that we used in our initial computational approaches.

2) Trajectory Optimization: In order to assess the range of achievable trajectories, we generated trajectories targeting launch times between 0 and 0.2 seconds and launch angular velocities between 70 and 110 rad/s for the system model with rotordynamics and actuator constraints. Full coverage of this trajectory space would demonstrate that rotordynamics-aware optimal control of this system would be able to outperform blind input variation of an "ideal" version of the system in achieving trajectory variation. As Figure 5 shows, this optimization was largely successful. Two particular features of these results are notable: first, we were unable to achieve launch angular velocities as high as in the ideal model case. This is largely due to the fact that it is not possible for separation between the arm and the flexor rotor to occur under spring loads as high as those developed in these simulations. The second is that high angular velocities are typically achieved by commanding negative forces $f_{fc}(t)$ in the flexor actuator, although even under these conditions the delivered flexor force $f_f(t)$ remains positive.

This strategy produced variations in launch angular velocity of 25 rad/s, between 77 and 102 rad/s, and variations in launch time of the full 0.2 s sought. We achieved this variation by leveraging the passive properties of the actuator to produce larger resistive forces than the "ideal" model would have been capable of, while still allowing actuator constraints and rotordynamics to affect the behavior of our system. These results demonstrate the utility of optimal control approaches for robotic platforms that use spring-actuated impulsive mechanisms, as our method proves that it is possible to achieve a great degree of control over the behavior of such a system even near the limits of actuator capability. Based on these results, we expect a similar approach to yield significant variation in the behavior of jumpers as well as projectile launchers like ours.

B. Experimental results

In our tests, we found that the the physical system tracked the model well in some cases and poorly in others. For target angular velocities of 80 rad/s or below, we observed maximum trajectory tracking errors as high as 1 radian and 70 rad/s, with no trajectories tracked well. Above this threshold, however, we observed maximum angular tracking errors as high as 0.5 rad and maximum angular velocity tracking errors as high as 31 rad/s interspersed with maximum errors as low as 0.05 rad and 8 rad/s. This irregularity above the threshold is due to the fact that deviations from expected behavior produced time shifts in the system behavior that caused large deviations between the expected and observed behaviors at any given time index for some force profiles, as can be seen in the red curve in Figure 6. These time shifts are generally

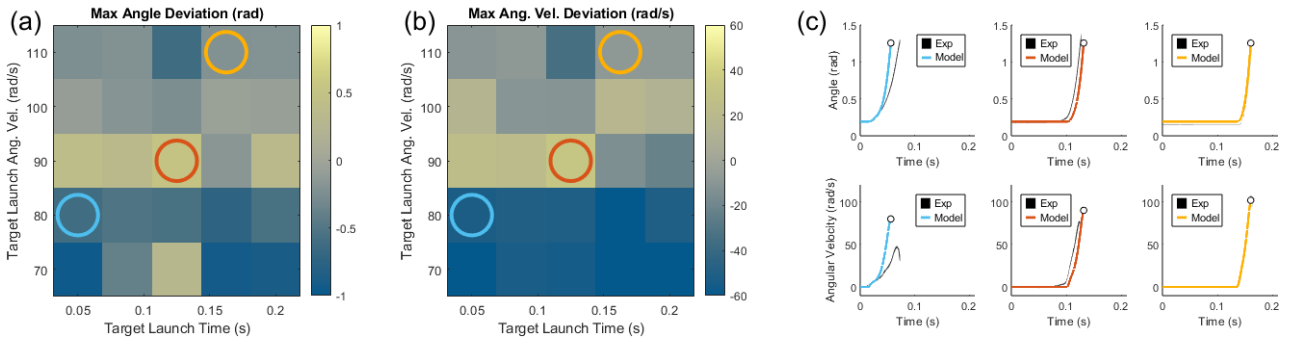

Fig. 6. Experimental results. (a) and (b) show the maximum angle and angular velocity tracking error for each generated trajectory respectively. (c) shows the angle and angular velocity for three representative experimental trials. In (c), the dark area marked "Exp" represents the envelope that all experimental data falls within and the colored line marked "Model" is the trajectory predicted by the rotordynamics model.

on the order of 10 ms, and if corrected for, the experimental performance of the system consistently matches the model.

Below the 80 rad/s threshold, however, we observed substantial divergence between modeled and observed behavior as shown in the blue curve in Figure 6. The observed launch speeds were well below the modeled launch speeds, and we were unable to account for these behaviors in our model. In this regime, the system actuated much more slowly than our model predicted, achieving launch velocities well below our targets. This behavior may indicate that the system was operating in a different frictional regime during these trials.

This result indicates that our model of the system is incomplete, and that alternative strategies must be pursued in order to achieve accurate low-speed actuation. On the other hand these speeds are lower than our model was able to achieve, suggesting that if we can develop an accurate model of the dynamics, the degree of trajectory variation we will be able to achieve in the physical system may increase substantially relative to our estimate from Section III-A.2.

Overall, the trajectories produced by our experimental system exhibit up to 0.2 s of variation in launch timing and up to 50 rad/s in launch angular velocity, although we were unable to accurately predict the system's behavior at the low end of this range. Some example trajectories are shown in Figure 7. With a standalone jumper, we expect to see this variation reflected in the possible ballistic trajectories, as we expect to see impacts on launch angle as well. Additionally, the behavior of the physical system was extremely consistent, as can be seen in the small spread of trajectories captured in the dark areas in Figure 6(c), with inter-trial variations in the observed system trajectories less than 0.15 radians in displacement and less than 5 rad/s in angular velocity. This consistency substantiates the suitability of this control strategy for robotic applications, as the behavior of the system is likely to be highly deterministic.

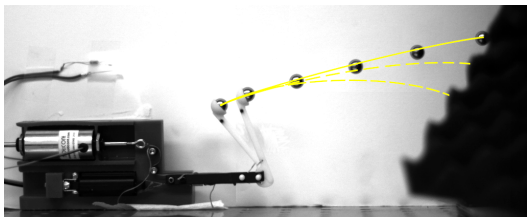

Fig. 7. Several launch trajectories of our mechanism starting from the same initial condition. The projectile that generated the trajectory represented by the solid line is shown above.

IV. Conclusions and Future Work

This paper demonstrates that it is possible to exploit the dynamic properties of a lossy, non-stiff actuator operating near the limits of its capability to produce extensive variation in the trajectory of an impulsive spring actuated mechanism. In addition, we experimentally showed that the space of feasible force inputs for such a system is actually larger than the space of feasible force inputs for an idealized lossless system, allowing us to produce a large range of trajectory outputs. However, this increased variation comes at the cost of sensitivity to the dynamics of the system; in slow dynamic regimes, these systems can become quite challenging to control, particularly in regimes where stick-slip-like dynamics may dominate the system's behavior. For this reason, it is important to establish the limitations of model validity when planning these trajectories.

Although impulsive spring-actuated systems are typically designed to release as much energy as possible, this method provides the possibility of designing these systems to enable trajectory variation in addition to power amplification. By proving the viability of a control pathway that does not require additional hardware, our results point to a means by which these systems such as jumpers, strikers, and launchers like ours can be made lighter, more responsive, and more efficient. In a sense, these systems provide control authority amplification alongside power amplification, as long as the models are of sufficient quality.

Work remains to be done to further increase the accuracy of the generated optimal controls at low speeds. The use of a neural network for computing inverse dynamics that correspond more closely to the observed system behavior presents one possible avenue by which deficits in the experimental performance might be addressed.

REFERENCES

[1] M. Ilton, M. S. Bhamla, X. Ma, S. M. Cox, L. L. Fitchett, Y. Kim, J. sung Koh, D. Krishnamurthy, C.-Y. Kuo, F. Z. Temel, A. J. Crosby, M. Prakash, G. P. Sutton, R. J. Wood, E. Azizi, S. Bergbreiter, and S. N. Patek, "The principles of cascading power limits in small, fast biological and engineered systems," *Science*, vol. 360, no. 6387, p. eaao1082, 2018.

[2] W. Heitler, "The locust jump - specialisations of the metathoracic femoral-tibial joint," *Journal of Comparative Physiology A: Neuroethology, Sensory, Neural, and Behavioral Physiology*, vol. 89, pp. 93–104, Mar. 1974.

[3] A. Konez Eroğlu, "Development and analysis of grasshopper-like jumping mechanism in biomimetic approach," Master's thesis, Middle East Technical University, Üniversiteler, Dumlupınar Blv. 1/6 D:133, 06800 Çankaya/Ankara, Turkey, 2007.

[4] V. Zaitsev, O. Gvirsman, U. B. Hanan, A. Weiss, A. Ayali, and G. Kosa, "A locust-inspired miniature jumping robot," *Bioinspiration & Biomimetics*, vol. 10, p. 066012, nov 2015.

[5] M. Noh, S.-W. Kim, S. An, J.-S. Koh, and K.-J. Cho, "Flea-inspired catapult mechanism for miniature jumping robots," *IEEE Transactions on Robotics*, vol. 28, no. 5, pp. 1007–1018, 2012.

[6] M. Kovač, M. Fuchs, A. Guignard, J. christophe Zufferey, and D. Floreano, "A miniature 7g jumping robot," in *IEEE International Conference on Robotics and Automation (ICRA)*, pp. 373–378, 2008.

[7] E. W. Hawkes, C. Xiao, R.-A. Peloquin, C. Keeley, M. R. Begley, M. T. Pope, and G. Niemeyer, "Engineered jumpers overcome biological limits via work multiplication," *Nature*, vol. 604, pp. 657–661, Apr 2022.

[8] G. P. Sutton and M. Burrows, "The mechanics of elevation control in locust jumping," *Journal of Comparative Physiology A*, vol. 194, pp. 557–563, Jun 2008.

[9] S. Divi, X. Ma, M. Ilton, R. St. Pierre, B. Eslami, S. N. Patek, and S. Bergbreiter, "Latch-based control of energy output in spring actuated systems," *Journal of The Royal Society Interface*, vol. 17, no. 168, p. 20200070, 2020.

[10] E. Steinhardt, N.-s. P. Hyun, J.-s. Koh, G. Freeburn, M. H. Rosen, F. Z. Temel, S. N. Patek, and R. J. Wood, "A physical model of mantis shrimp for exploring the dynamics of ultrafast systems," *Proceedings of the National Academy of Sciences*, vol. 118, no. 33, 2021.

[11] G. Ribak, "Insect-inspired jumping robots: challenges and solutions to jump stability," *Current Opinion in Insect Science*, vol. 42, pp. 32–38, 2020. Neuroscience * Biomechanics of Insect Flight and Bio-inspired engineering.

[12] D. W. Haldane, M. M. Plecnik, J. K. Yim, and R. S. Fearing, "Robotic vertical jumping agility via series-elastic power modulation," *Science Robotics*, vol. 1, no. 1, p. eaag2048, 2016.

[13] H. C. Bennet-Clark, "The energetics of the jump of the locust schistocerca gregaria.," *The Journal of experimental biology*, vol. 63 1, pp. 53–83, 1975.

[14] D. Cofer, G. Cymbalyuk, W. J. Heitler, and D. H. Edwards, "Neuromechanical simulation of the locust jump," *The Journal of Experimental Biology*, vol. 213, pp. 1060–1068, Apr. 2010.

[15] T. L. Khuong, "Design, implementation and analysis of 3d printed grasshopper robot for jumping mechanism," *Journal of Biomimetics, Biomaterials and Biomedical Engineering*, vol. 28, pp. 1–13, 2016.

[16] Q.-V. Nguyen and H. C. Park, "Design and demonstration of a locust-like jumping mechanism for small-scale robots," *Journal of Bionic Engineering*, vol. 9, no. 3, pp. 271–281, 2012.