http://www.ucs.louisiana.edu/~jev9637/

Learning Control Strategies and Design Parameters for Flexible-legged Locomotive Systems

Andrew Albright, Joshua Vaughan

andrew.albright1@louisiana.edu

Flexible Robotics

Figure 1: Boston Dynamics Atlas Robot [1]

Legged systems have many advantages when compared to their wheeled counterparts. For example, if designed properly, they can more easily navigate uneven and unstable terrain. However, there are disadvantages as well, including dramatically less energy efficient locomotion ~~that can causes them to be inconsiderable as mobile robotic solutions.~~

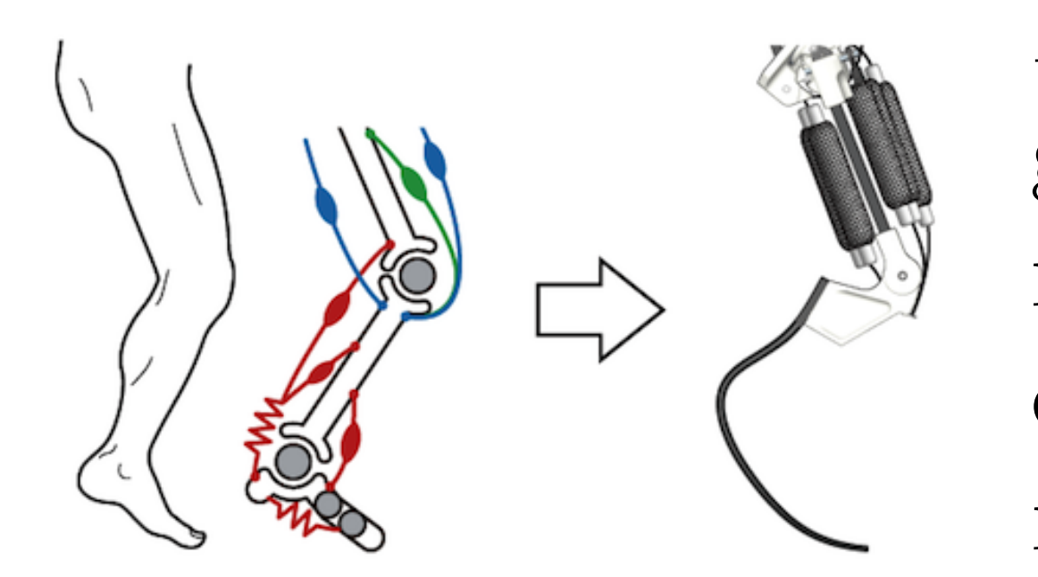
Figure 2: Athlete Robot [2]

In an effort to mitigate this disadvantage, research has been conducted that shows using flexible components in legged locomotive systems not only increases their efficiency but also their performance [3]. However, nonlinear models and controllers for flexible systems are difficult to develop using traditional methods.

Reinforcement Learning

A solutions to combat the nonlinear difficulties is the use of Reinforcement Learning which uses neural networks to represent the nonlinear aspects of the control architecture. Using an RL approach is attractive because it does not require ~~extreme levels of domain knowledge~~ closed-form models of the system dynamics to develop a controller. Rather, the RL approach lets a Machine Learning based agent interact with an environment to collect samples of rewards and actions that get mapped to system states. With this map, the agent ~~will have~~ define~~d~~s inputs for the system based in its state. Previous work has shown that this method can be successful at defining both mechanical parameters and control strategies for rigid systems [4].

no caps

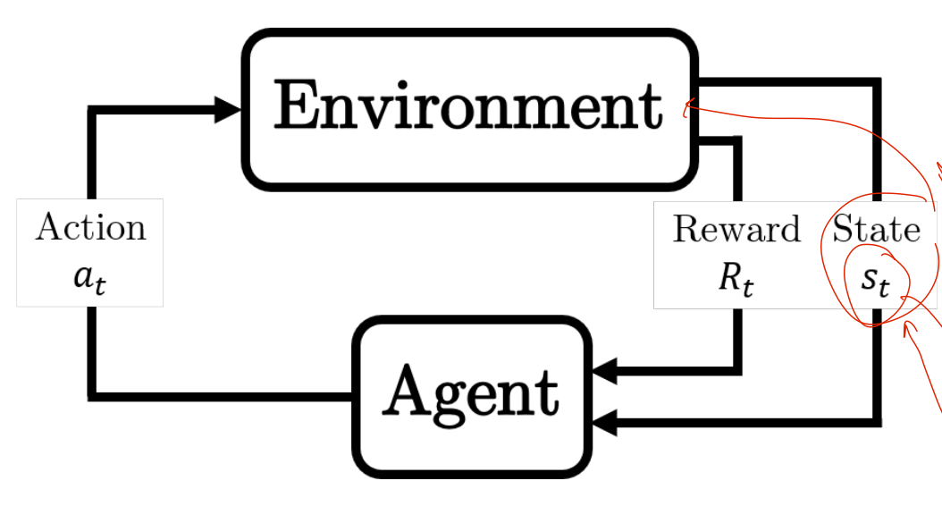

Figure 3: RL Block Diagram

make same text size

figure out how to get same text as LaTeX equations into whatever you use to generate this figure

in general, we want all text in figures and plots to be at least as large as the document body text

See Style Guide

Results Discussion

See Style Guide

Awkward

In this work it is discovered that using Reinforcement Learning for developing optimal control strategies leads to higher performance both in desired movement and energy efficiency. A pogo stick system, shown in Figure 4, is used to train an agent to determine the acceleration of the mass, m_a, along the rod, m_l, causing system to jump. The environment is defined such that the system can only move in the vertical direction so that keeping balanced is not part of the control input. The agent is tasked to learn a control strategy that jumps ~~the pogo stick~~ as high as possible in a single jump while also ~~maintaining~~ limiting power ~~conservation~~ consumption. Two training strategies are deployed to accomplish this task. In the first strategy, the agent is rewarded based on the height it achieves receiving a score directly correlated to the height it reaches. For the second strategy the agent is rewarded using a normalized positive score for jump~~ing~~ ~~high~~ height and punished a normalized negative score for using power.

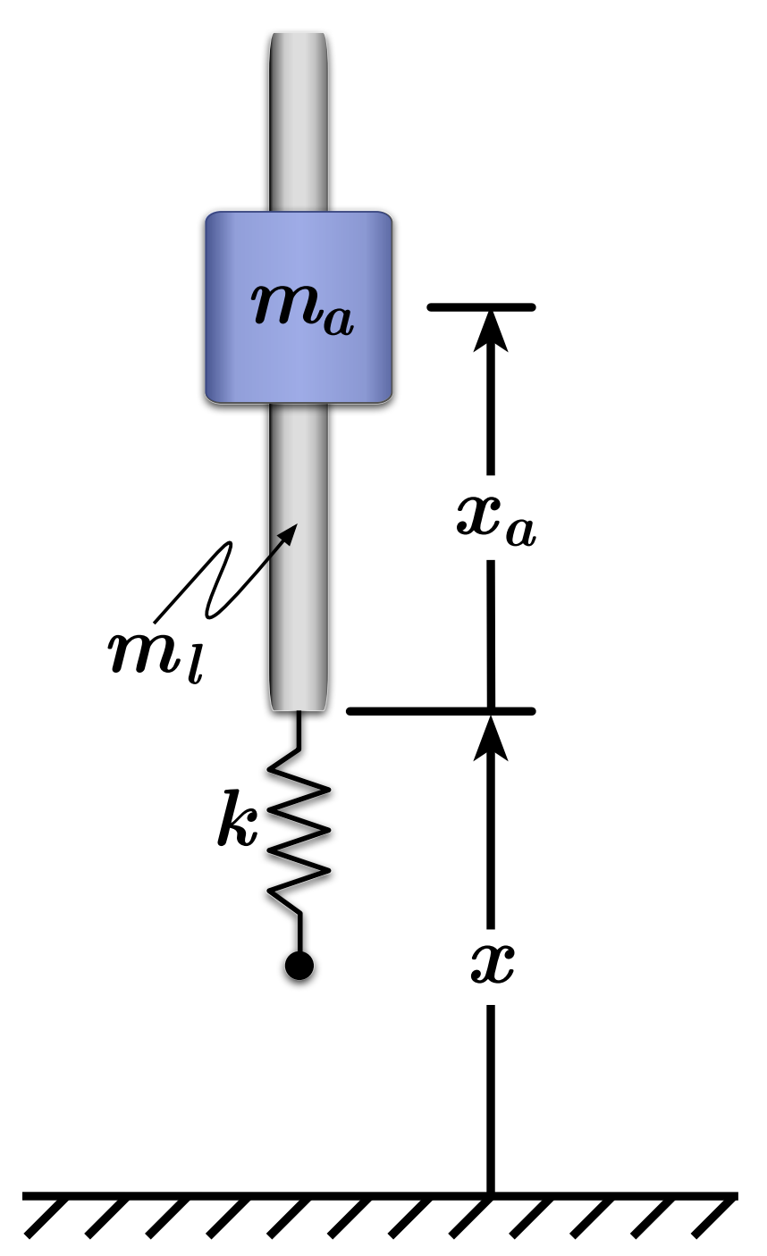

Figure 4: Pogostick

Jumping Analysis

Time series jumping results and power usage data are collected during an evaluation study after the agent is trained and ready to control the pogo sick. Figure 6 shows the jumping profile and power usage when using the first training strategy, where the agent is rewarded only based on jumping height. The maximum height achieved is 0.286 meters and the total power used to reach that height is $x.xx$ Watts. The jumping motion seen here is called a stutter jump and can be visualized in Figure 5. This kind of motion is expected as it has been shown previously in [5] that it leads to higher jumps.

need to reference before figure 6 is referenced

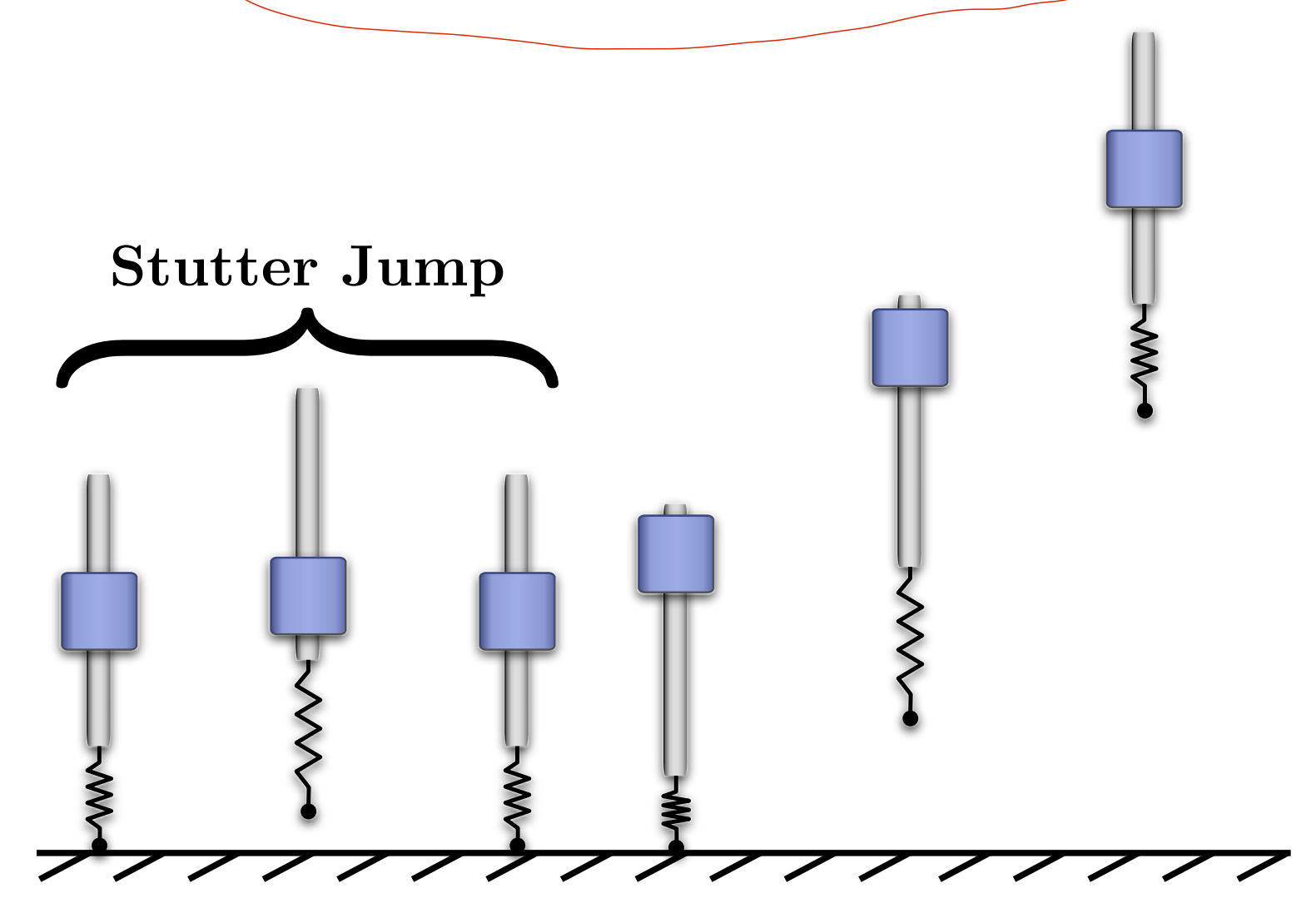

Figure 5: Rewarding Jump Height Data

When using the second training strategy, where the agent is rewarded for jumping and punished for using power, the max jump height achieved is $x.xx$ meters. This is not as high as before, however it achieved this height using X% less power as seen in Figure 7 where the total power usage adds to $x.xx$ Watts.

Evaluation Data

ask me if you want/need the LaTeX snippet

Can you use the "true" subfigure package here?

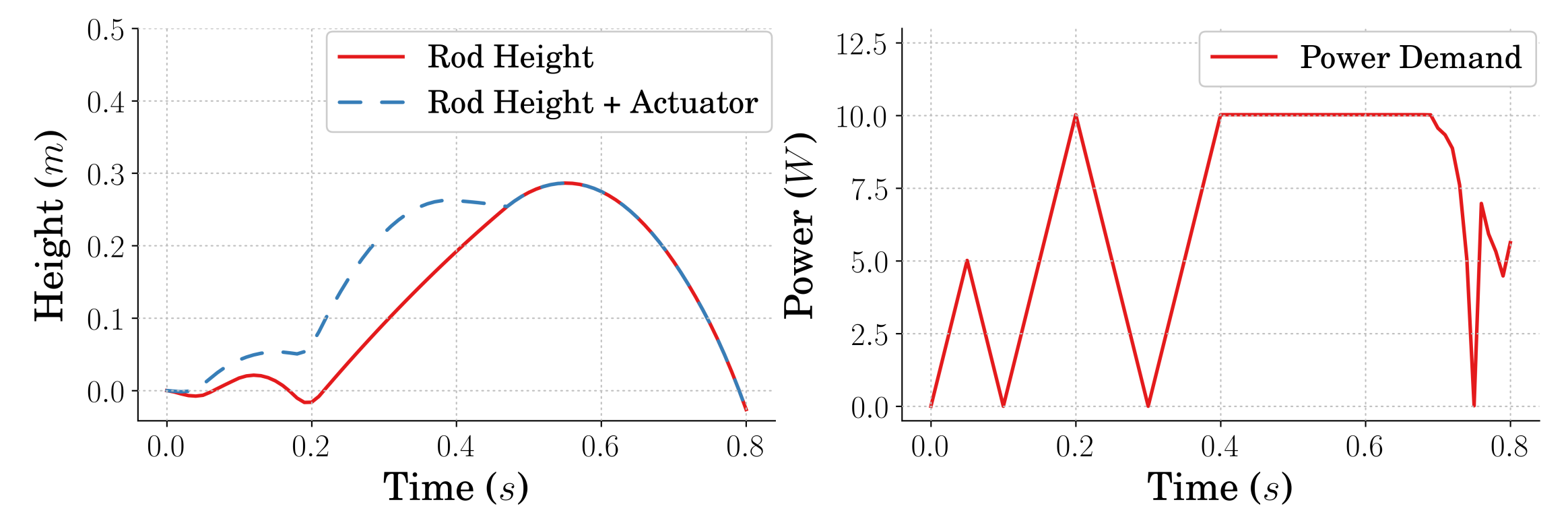

Figure 6: Rewarding Jump Height Data

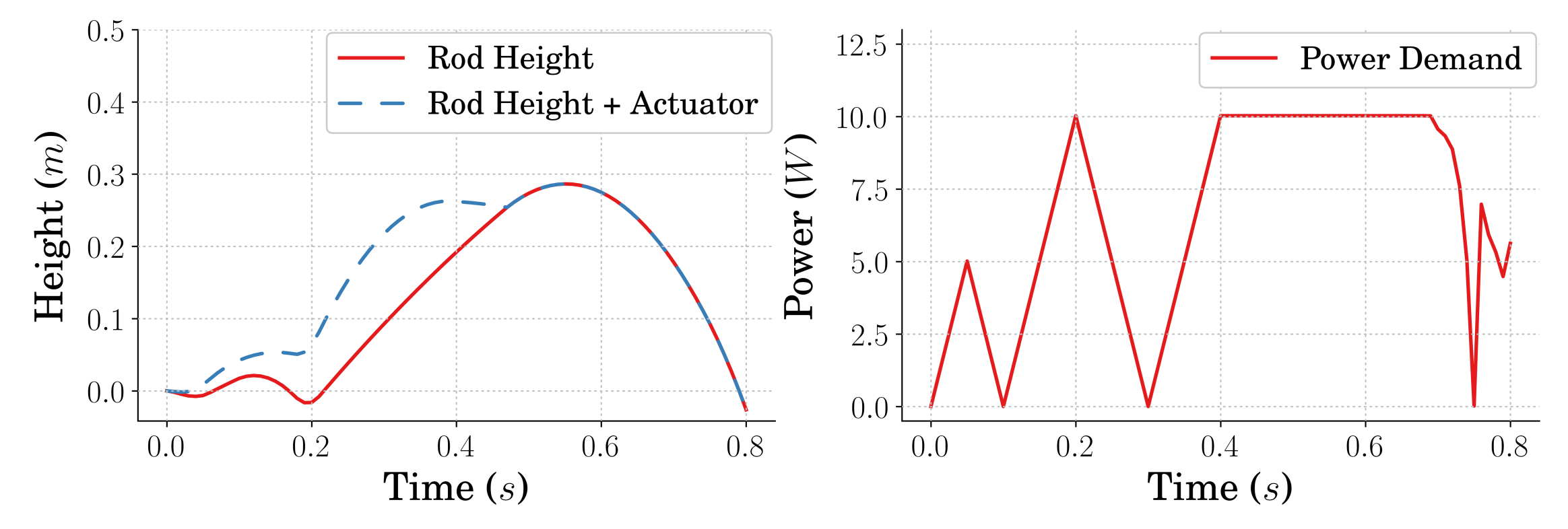

Figure 7: Rewarding Jump Height Punishing Power Usage Data

Seems this should come before results?

Defining the Reward

Using the weights ω_x and ω_p, the reward function (1) can be tuned to guide the agents learning process towards maximizing power efficiency.

$$R = \frac{\frac{x_t - x_{min}}{x_{max} - x_{min}}\omega_x + m_a\frac{a_t v_t - a_{min} v_{min}}{a_{max} v_{max} - a_{min} v_{min}}\omega_p - \omega_p}{\omega_x - \omega_p} \quad (1)$$

leave some bottom margin

where ~~Note~~: x_t, a_t and v_t are ~~system state parameters~~ position, velocity, and acceleration, respectively. ~~which are updated every time step (0.01 seconds).~~

Implications of Results

Nunc tempus venenatis facilisis. Curabitur suscipit consequat eros non porttitor. Sed a massa dolor, id ornare enim. Nunc tempus venenatis facilisis. Curabitur suscipit consequat eros non porttitor. Nunc tempus venenatis facilisis. Curabitur suscipit consequat eros non porttitor. Sed a massa dolor, id ornare enim. Nunc tempus venenatis facilisis. Curabitur suscipit consequat eros non porttitor.

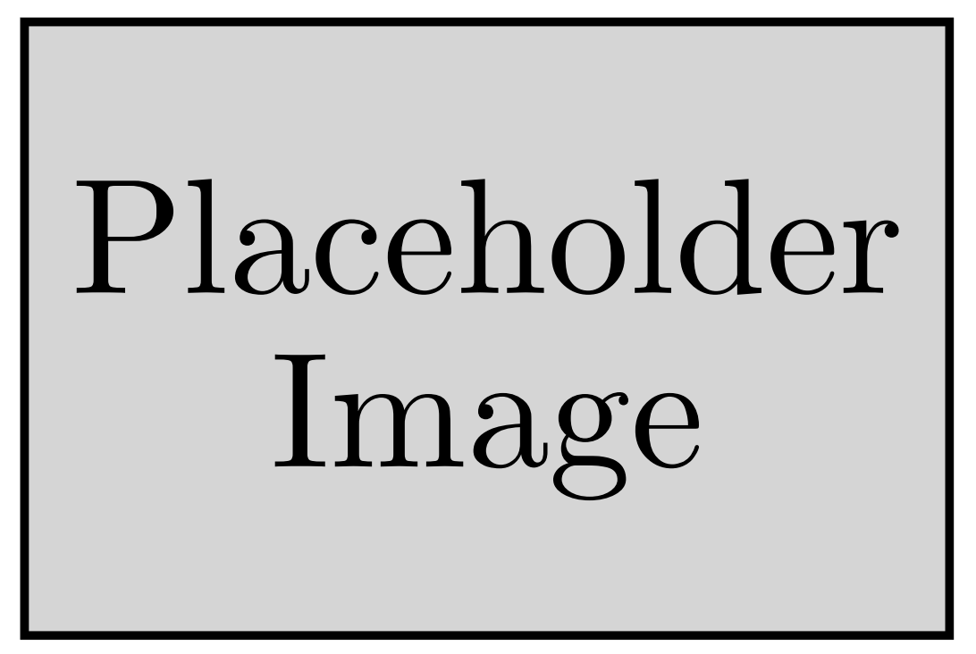

Figure: Figure Caption

Conclusion

Nunc tempus venenatis facilisis. **Curabitur suscipit** consequat eros non porttitor. Sed a massa dolor, id ornare enim. Fusce quis massa dictum tortor **tincidunt mattis**. Donec quam est, lobortis quis pretium at, laoreet scelerisque lacus. Nam quis odio enim, in molestie libero. Vivamus cursus mi at *nulla elementum sollicitudin*.

References

[1] Boston Dynamics.
Atlas® boston dynamics, 2021.

[2] designboom.
ryuma niiyama athlete robot, 2010.

[3] Yuuta Sugiyama and Shinichi Hirai.
Crawling and jumping of deformable soft robot.
2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 4(c):3276–3281, 2004.

[4] David Ha.
Reinforcement learning for improving agent design.
Artificial Life, 25(4):352–365, 2019.

[5] Joshua Vaughan.
Jumping Commands For Flexible-Legged Robots.
2013.

Don't thank me or MCHE (unless they do something out of ordinary). That's like thanking ourselves. You should thank Crawfish Board, since they provide your funding

Acknowledgements

Special thanks to Dr. Vaughan, UL's Mechanical Engineering Department and the CRAWLAB for providing the resources to perform this research.